### Contents

**Figures** ..................................... xiii

**Tables** ...................................... xv

**About this document** ................................. xvii

Who should read this document ............................ xvii

How this document is organized ............................. xvii

How to use this document ................................ xviii

   - Determining whether a publication is current .................... xviii
   - How to contact IBM service ................................ xviii

Conventions and terminology that are used in this document ................ xix

How to read a syntax diagram ............................... xx

Prerequisite and related information ........................... xxiv

**Summary of changes** ................................ xxix

Changes made in z/OS Version 2 Release 1, as updated February 2015........ xxix

Changes made in z/OS Version 2 Release 1, as updated December 2013 ........ xxix

Summary of changes for z/OS Version 2 Release 1 ........................... xxix

**Chapter 1. Introduction to VTAM resource definition** .................. 1

**Major nodes** .................................... 2

   - Adjacent control point major node ............................ 2
   - Application program major node .............................. 2
   - Channel-attachment major node ............................... 2
   - Cross-domain resource major node ............................ 2
   - Cross-domain resource manager major node ................... 2
   - External communication adapter major node .................... 2
   - Local non-SNA major node ................................. 3
   - Local SNA major node ................................... 3
   - LU group major node .................................... 3
   - Model major node ........................................ 3
   - NCP major node ......................................... 3
   - Switched major node ..................................... 3
   - Transport resource list major node .......................... 3

**Routing functions and dynamic reconfiguration** ...................... 3

   - Adjacent SSCP tables ................................... 3
   - Path definition statements ................................ 4
   - Dynamic reconfiguration and dynamic change ................ 4
   - Transmission group profile ............................... 5
   - Network node server list ................................. 5

**User-defined tables and session awareness data filter** .............. 5

   - Associated LU table .................................. 5
   - Class of Service ....................................... 6
   - CMIP services directory definition file ...................... 6
   - Interpret table ....................................... 6
   - Logon mode table .................................... 7
   - Message-flooding prevention table .......................... 7
   - Model name table ..................................... 7
   - Session awareness (SAW) data filter ........................ 7
   - Unformatted system services (USS) table .................... 7

**Coding VTAM definitions** ................................ 8

   - Format of definition statements .......................... 8
   - Format of macroinstructions ............................. 10
   - Rules for coding start option lists and configuration lists ... 11

© Copyright IBM Corp. 2000, 2015
Adjacent cluster routing definitions coding notes ........................................... 714
Adjacent cluster routing definitions full syntax ........................................... 715
Adjacent cluster routing definitions operand descriptions .................................. 716
APPN transmission group profile ................................................................. 720
  TGP definition statement ................................................................. 720
  IBM-supplied transmission group profiles ............................................. 720
  APPN transmission group profile full syntax ......................................... 721
  APPN transmission group profile operand descriptions ........................... 721
APPN-to-subarea Class of Service mapping table ........................................... 730
  APPN-to-subarea Class of Service mapping table coding notes .................. 732
  APPN-to-subarea Class of Service mapping table full syntax .................... 732
  APPN-to-subarea Class of Service mapping table operand descriptions ........ 733
Subarea-to-APPN Class of Service mapping table ........................................... 735
  Subarea-to-APPN Class of Service mapping table coding notes .................. 736
  Subarea-to-APPN Class of Service mapping table full syntax .................... 736
  Subarea-to-APPN Class of Service mapping table operand descriptions ....... 737
Border node CoS mapping definitions ......................................................... 739
  Border node CoS mapping definitions coding notes ............................... 740
  Border node CoS mapping definitions full syntax ................................... 741
  Border node CoS mapping definitions operand descriptions ..................... 741
Network node server list .............................................................................. 743
  Network node server list coding notes ................................................... 744
  Network node server list full syntax ..................................................... 744
  Network node server list operand descriptions ...................................... 745
Generic resources preferences table ............................................................. 748
  Generic resources preferences table coding notes .................................... 749
  Generic resources preferences table full syntax ..................................... 749
  Generic resources preferences table operand descriptions ....................... 750

Chapter 4. VTAM start options ........................................................................... 755
  Coding VTAM start options ........................................................................ 755
    Using MVS System Symbols in VTAM Start Options ................................. 756
    Start options syntax diagrams .............................................................. 756
  Summary of start options ........................................................................... 774
    SSCP or CP characteristics start options ................................................ 775
    VTAM initialization start options ............................................................ 776
    Storage usage start options ..................................................................... 776
    Session control start options .................................................................. 777
    Connectivity start options ...................................................................... 779
    Session security start options ............................................................... 780
    Messages start options .......................................................................... 780
    Recording and statistics start options ..................................................... 782
    Traces and dumps start options .............................................................. 782
    Buffer pools start options ....................................................................... 783
    Performance and tuning start options ..................................................... 783
    APPN start options ................................................................................. 786
    Functions from zappable constants .......................................................... 789
    HPR and Enterprise Extender start options .............................................. 791
    Network management start options ........................................................ 792
    Coupling facility and SYSPLEX start options .......................................... 793
  Descriptions of start options ...................................................................... 794
    AFFDELAY start option ........................................................................... 794
    ALSREQ start option .............................................................................. 794
    API64R start option ................................................................................ 794
    APPNCOSEX start option ....................................................................... 795
    ASIRMMSG start option .......................................................................... 796
    ASYE start option ................................................................................... 797
    AUTHLEN start option ............................................................................ 798
    AUTORETRY start option ......................................................................... 798
    AUTORTRY start option ........................................................................... 799
    AUTOTI start option ................................................................................ 799
    BN start option ....................................................................................... 800
BNORD start option ........................................ 801
BSCMDRS start option .............................. 802
BSCTMOUT start option ............................ 803
Buffer Pool start option ............................ 804
CACHETI start option ............................... 811
CDRDYN start option ............................... 811
CDRSTCI start option ............................. 812
CDSEVR start option ............................... 813
CDSPREFER start option .......................... 814
CINDXSIZ start option ............................. 814
CMPMIPS start option .............................. 815
CMVPVTAM start option ........................... 815
CNMTAB start option ............................... 816
CNNRTMSG start option ............................ 816
COLD and WARM start option ....................... 817
CONFIG start option ............................... 817
CONNTYPE start option .......................... 818
CPCDRSC start option .............................. 819
CPCP start option ................................. 820
CSALIMIT start option ............................ 820
CSA24 start option ............................... 822
CSDUMP start option ............................... 823
DATEFORM start option ........................... 827
DIALRTRY start option ........................... 828
DIRSIZE start option .............................. 829
DIRTIME start option ............................. 829
DISCNTIM start option ........................... 830
DLRORDER start option ........................... 831
DLRTCB start option .............................. 832
DLURSAW start option ............................ 833
DSIRFMSG start option ............................ 834
DSPLYDEF start option ............................ 835
DSPLYMAX start option .......................... 836
DSPLYWLD start option ........................... 837
DUPDEFS start option ............................. 838
DYNADJCP start option ........................... 839
DYNASSCP start option ........................... 840
DYNDLGMD start option ........................... 841
DYNHPPFX start option ............................ 842
DYNLU start option ............................... 842
DYNMODTB start option .......................... 843
DYNPUPFX start option ............................ 844
DYNVNPFX start option ............................ 844
EEHPRANR start option ........................... 845
EEPORTEK start option ........................... 845
EEVERIFY start option ............................ 846
ENCRPREF start option ........................... 846
ENCRYPTN start option ........................... 847
ENHADDR start option ............................. 847
ENSEMBLE start option ........................... 849
ESIRFMSG start option ............................ 849
EXPFITLRM start option ........................... 850
FLDTAB start option ............................... 851
FLRADDR start option ............................. 852
FSIRFMSG start option ............................ 853
GVBDL start option ............................... 854
GWSSCP start option .............................. 854
HNTSIZE start option ............................. 855
HOSTNAME start option ........................... 856
HOSTPU start option .............................. 858
HOSTSA start option ............................... 858
<table>
<thead>
<tr>
<th>Start option</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMTUD</td>
<td>911</td>
</tr>
<tr>
<td>PPOLOG</td>
<td>912</td>
</tr>
<tr>
<td>PROMPT and NOPROMPT</td>
<td>913</td>
</tr>
<tr>
<td>PSRETRY</td>
<td>914</td>
</tr>
<tr>
<td>PSSTRACE</td>
<td>916</td>
</tr>
<tr>
<td>PSWEIGHT</td>
<td>917</td>
</tr>
<tr>
<td>QDIOSTG</td>
<td>918</td>
</tr>
<tr>
<td>RESUSAGE</td>
<td>919</td>
</tr>
<tr>
<td>ROUTERES</td>
<td>920</td>
</tr>
<tr>
<td>RSIRFMSG</td>
<td>920</td>
</tr>
<tr>
<td>SACONNS</td>
<td>921</td>
</tr>
<tr>
<td>SAVERSCV</td>
<td>922</td>
</tr>
<tr>
<td>SAWMAXDS</td>
<td>923</td>
</tr>
<tr>
<td>SAWMAXQPK</td>
<td>924</td>
</tr>
<tr>
<td>SDLCMDRS</td>
<td>924</td>
</tr>
<tr>
<td>SECLVLCP</td>
<td>925</td>
</tr>
<tr>
<td>SIRFMSG</td>
<td>926</td>
</tr>
<tr>
<td>SLOWVAL</td>
<td>927</td>
</tr>
<tr>
<td>SLUALMSG</td>
<td>928</td>
</tr>
<tr>
<td>SMEAUTH</td>
<td>929</td>
</tr>
<tr>
<td>SNAMGMT</td>
<td>929</td>
</tr>
<tr>
<td>SNAPREQ</td>
<td>930</td>
</tr>
<tr>
<td>SNVC</td>
<td>931</td>
</tr>
<tr>
<td>SONLIM</td>
<td>931</td>
</tr>
<tr>
<td>SORDER</td>
<td>932</td>
</tr>
<tr>
<td>SRCHRED</td>
<td>934</td>
</tr>
<tr>
<td>SRCOUNT</td>
<td>934</td>
</tr>
<tr>
<td>SRTIMER</td>
<td>934</td>
</tr>
<tr>
<td>SSCPDYN</td>
<td>935</td>
</tr>
<tr>
<td>SSCPNAME</td>
<td>936</td>
</tr>
<tr>
<td>SSCPORD</td>
<td>937</td>
</tr>
<tr>
<td>SSDTMOUT</td>
<td>937</td>
</tr>
<tr>
<td>SSEARCH</td>
<td>938</td>
</tr>
<tr>
<td>STRGR</td>
<td>938</td>
</tr>
<tr>
<td>STRMNPS</td>
<td>939</td>
</tr>
<tr>
<td>SUPP</td>
<td>940</td>
</tr>
<tr>
<td>SWNORDER</td>
<td>942</td>
</tr>
<tr>
<td>TCPNAME</td>
<td>943</td>
</tr>
<tr>
<td>TDUDIAG</td>
<td>944</td>
</tr>
<tr>
<td>TINSTAT</td>
<td>946</td>
</tr>
<tr>
<td>TOPOTIME</td>
<td>946</td>
</tr>
<tr>
<td>TRACE</td>
<td>947</td>
</tr>
<tr>
<td>VARYWLD</td>
<td>955</td>
</tr>
<tr>
<td>VERIFYCP</td>
<td>965</td>
</tr>
<tr>
<td>UNRCHTIM</td>
<td>966</td>
</tr>
<tr>
<td>UPDDELAY</td>
<td>967</td>
</tr>
<tr>
<td>USSSTAB</td>
<td>968</td>
</tr>
<tr>
<td>VARYWLD</td>
<td>969</td>
</tr>
<tr>
<td>VERIFYCP</td>
<td>970</td>
</tr>
<tr>
<td>VFYRED</td>
<td>971</td>
</tr>
<tr>
<td>VFYREDXT</td>
<td>972</td>
</tr>
<tr>
<td>VOSDEACT</td>
<td>973</td>
</tr>
<tr>
<td>VRTG</td>
<td>974</td>
</tr>
<tr>
<td>VRTGCPCP</td>
<td>974</td>
</tr>
<tr>
<td>VTAMEAS</td>
<td>975</td>
</tr>
<tr>
<td>XCFGRPID</td>
<td>976</td>
</tr>
<tr>
<td>XCFINIT</td>
<td>976</td>
</tr>
<tr>
<td>XNETALS</td>
<td>978</td>
</tr>
</tbody>
</table>
Chapter 5. User-defined tables and data filter

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Associated LU table</td>
<td>981</td>
</tr>
<tr>
<td>Installing an associated LU table</td>
<td>982</td>
</tr>
<tr>
<td>Associated LU table full syntax</td>
<td>982</td>
</tr>
<tr>
<td>Associated LU table operand descriptions</td>
<td>983</td>
</tr>
<tr>
<td>Class of service</td>
<td>985</td>
</tr>
<tr>
<td>Subarea class-of-service table</td>
<td>985</td>
</tr>
<tr>
<td>Subarea class-of-service table coding notes</td>
<td>986</td>
</tr>
<tr>
<td>Installing a subarea CoS table</td>
<td>987</td>
</tr>
<tr>
<td>Subarea class-of-service table full syntax</td>
<td>987</td>
</tr>
<tr>
<td>Subarea class-of-service table operand descriptions</td>
<td>988</td>
</tr>
<tr>
<td>APPN Class of Service definitions</td>
<td>990</td>
</tr>
<tr>
<td>APPN Class of Service coding notes</td>
<td>991</td>
</tr>
<tr>
<td>Implementing APPN Classes of Service</td>
<td>991</td>
</tr>
<tr>
<td>APPN Class of Service full syntax</td>
<td>991</td>
</tr>
<tr>
<td>APPN Class of Service operand descriptions</td>
<td>992</td>
</tr>
<tr>
<td>Default values for APPN Class of Service definitions</td>
<td>1005</td>
</tr>
<tr>
<td>IBM-supplied APPN Classes of Service</td>
<td>1007</td>
</tr>
<tr>
<td>CMIP services directory definition file</td>
<td>1008</td>
</tr>
<tr>
<td>Example of a simple directory definition file</td>
<td>1008</td>
</tr>
<tr>
<td>Explanation of the example directory definition file</td>
<td>1008</td>
</tr>
<tr>
<td>Specifying different types of associations</td>
<td>1009</td>
</tr>
<tr>
<td>Coding the directory definition file</td>
<td>1010</td>
</tr>
<tr>
<td>How CMIP services uses the directory definition file</td>
<td>1015</td>
</tr>
<tr>
<td>Troubleshooting the directory definition file</td>
<td>1016</td>
</tr>
<tr>
<td>Interpret table</td>
<td>1017</td>
</tr>
<tr>
<td>Interpret table coding notes</td>
<td>1017</td>
</tr>
<tr>
<td>Coding logon-interpret routines</td>
<td>1018</td>
</tr>
<tr>
<td>Installing and changing interpret tables</td>
<td>1020</td>
</tr>
<tr>
<td>Interpret table full syntax</td>
<td>1021</td>
</tr>
<tr>
<td>Interpret table operand descriptions</td>
<td>1022</td>
</tr>
<tr>
<td>Logon mode table</td>
<td>1026</td>
</tr>
<tr>
<td>Logon mode table coding notes</td>
<td>1027</td>
</tr>
<tr>
<td>Installing a logon mode table</td>
<td>1028</td>
</tr>
<tr>
<td>Logon mode table full syntax</td>
<td>1028</td>
</tr>
<tr>
<td>Logon mode table operand descriptions</td>
<td>1030</td>
</tr>
<tr>
<td>Message-flooding prevention table</td>
<td>1046</td>
</tr>
<tr>
<td>Message-flooding prevention table coding notes</td>
<td>1047</td>
</tr>
<tr>
<td>Installing a message-flooding prevention table</td>
<td>1047</td>
</tr>
<tr>
<td>Considerations for message-flooding prevention</td>
<td>1048</td>
</tr>
<tr>
<td>Example of message suppression</td>
<td>1048</td>
</tr>
<tr>
<td>Message-flooding prevention table full syntax</td>
<td>1049</td>
</tr>
<tr>
<td>Message-flooding prevention table operand descriptions</td>
<td>1050</td>
</tr>
<tr>
<td>Model name table</td>
<td>1054</td>
</tr>
<tr>
<td>Model name table coding notes</td>
<td>1054</td>
</tr>
<tr>
<td>Installing a model name table</td>
<td>1055</td>
</tr>
<tr>
<td>Model name table full syntax</td>
<td>1055</td>
</tr>
<tr>
<td>Model name table operand descriptions</td>
<td>1056</td>
</tr>
<tr>
<td>Subarea mapping table</td>
<td>1057</td>
</tr>
<tr>
<td>Subarea mapping table coding notes</td>
<td>1058</td>
</tr>
<tr>
<td>Subarea mapping table full syntax</td>
<td>1058</td>
</tr>
<tr>
<td>Subarea mapping table operand descriptions</td>
<td>1059</td>
</tr>
<tr>
<td>Session awareness data filter</td>
<td>1061</td>
</tr>
<tr>
<td>Session awareness data filter coding notes</td>
<td>1062</td>
</tr>
<tr>
<td>Session awareness data filter full syntax</td>
<td>1062</td>
</tr>
<tr>
<td>Operand descriptions</td>
<td>1063</td>
</tr>
<tr>
<td>SAW sense filter</td>
<td>1066</td>
</tr>
<tr>
<td>Coding notes</td>
<td>1067</td>
</tr>
<tr>
<td>Full syntax</td>
<td>1067</td>
</tr>
<tr>
<td>Operand descriptions</td>
<td>1068</td>
</tr>
</tbody>
</table>
Unformatted system services tables ................................................. 1069
  Customizing USS tables ......................................................... 1070
  Order of USS table use .......................................................... 1071
  Installing a user-defined USS table ........................................... 1073
  Coding a user-defined USS table ............................................. 1073
  Unformatted system services tables coding notes .......................... 1074
  Unformatted system services (USS) tables full syntax ......................... 1075
  Operand descriptions .............................................................. 1078
  Customizing messages .............................................................. 1098
  Defining VTAM operator commands ........................................... 1101
  Defining terminal operator commands .................................... 1102
  LOGON command ...................................................................... 1103
  Full syntax .............................................................................. 1104
  Operand descriptions .............................................................. 1104
  LOGOFF command ..................................................................... 1108
  Full syntax .............................................................................. 1108
  Operand descriptions .............................................................. 1109
  IBMTEST command .................................................................... 1113
  Full syntax .............................................................................. 1113
  Operand descriptions .............................................................. 1114

Appendix A. IBM-supplied tables ..................................................... 1117
  Default logon mode table (ISTINCLM) ........................................... 1117
  Session-level USS table (ISTINCDT) ............................................. 1132
  Operation-level USS table (ISTINCNO) .......................................... 1133
  APPN Class of Service (CoS) definitions (COSAPPN) .......................... 1313
    COSAPPN ........................................................................ 1313
    ISTACST2 ..................................................................... 1325
    ISTACST3 ..................................................................... 1340
  APPN transmission group (TG) profile definitions (IBMTGPS) ................. 1356
    Ethernet .......................................................................... 1356
    Token ring ........................................................................ 1356
    ISDN non-switched ................................................................ 1356
    ISDN switched ................................................................... 1356
    SDLC non-switched ................................................................ 1356
    SDLC switched .................................................................... 1356
    X.25 .............................................................................. 1356
    16M token ring ................................................................... 1356
    Channel ............................................................................ 1356
    ESCON channel ................................................................... 1356
    XCF .............................................................................. 1357
    ATM TGs over PVCs [campus (private) ATM best effort] .................. 1357
    ATM TGs over PVCs (campus ATM reserved bandwidth) ................ 1357
    ATM TGs over PVCs [WAN (public) ATM best effort] ................... 1357
    ATM TGs over PVCs (WAN ATM reserved bandwidth) ................... 1357
    ATM TGs over SVCs (campus ATM best effort) ............................ 1357
    ATM TGs over SVCs (campus ATM reserved bandwidth) ................ 1357
    ATM TGs over SVCs (WAN ATM best effort) .............................. 1357
    ATM TGs over SVCs (WAN ATM reserved bandwidth) ................... 1357
    Enterprise Extender, campus ................................................... 1357
    Enterprise Extender, wide area network ................................... 1357
    Fast Ethernet ...................................................................... 1357
    Gigabit Ethernet ................................................................... 1357
    Gigabit Ethernet, 10G ........................................................... 1358
    FICON ............................................................................. 1358
    FICON Express .................................................................... 1358
    HiperSockets ...................................................................... 1358
    Message flooding table (ISTMSFLD) ........................................... 1358
    CMIP services directory definition file (ACYDDF) ......................... 1367
<table>
<thead>
<tr>
<th>Appendix B. Examples of USS command conversion</th>
<th>1369</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appendix C. VTAMTOPO filtering option reporting</td>
<td>1373</td>
</tr>
<tr>
<td>Appendix D. Related protocol specifications</td>
<td>1377</td>
</tr>
<tr>
<td>Appendix E. Architectural specifications</td>
<td>1401</td>
</tr>
<tr>
<td>Appendix F. Accessibility</td>
<td>1403</td>
</tr>
<tr>
<td>Notices</td>
<td>1407</td>
</tr>
<tr>
<td>Notices</td>
<td>1407</td>
</tr>
<tr>
<td>Notices Policy for unsupported hardware</td>
<td>1415</td>
</tr>
<tr>
<td>Notices Trademarks</td>
<td>1415</td>
</tr>
<tr>
<td>Bibliography</td>
<td>1417</td>
</tr>
<tr>
<td>Index</td>
<td>1421</td>
</tr>
<tr>
<td>Communicating your comments to IBM</td>
<td>1441</td>
</tr>
</tbody>
</table>
## Figures

1. Adjacent cluster routing definitions ................................................. 714
2. APPN-to-subarea CoS mapping table example ................................ 731
3. APPN-to-subarea CoS mapping table used to disable the APPNTOSA COS mapping function example 732
4. Subarea-to-APPN CoS mapping table example .................................. 736
5. CoS mapping table example .......................................................... 740
6. GRPREFS table definition ............................................................. 749
7. Subarea mapping table example ..................................................... 1058
8. USS message layout in storage ....................................................... 1078
## Tables

<table>
<thead>
<tr>
<th>Table Number</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Adjacent control point major node definition statements and operands</td>
<td>19</td>
</tr>
<tr>
<td>2</td>
<td>DYNLU combinations</td>
<td>24</td>
</tr>
<tr>
<td>3</td>
<td>Application program major node (Non-LU 6.2) definition statements and operands</td>
<td>33</td>
</tr>
<tr>
<td>4</td>
<td>Application program major node (LU 6.2) definition statements and operands</td>
<td>34</td>
</tr>
<tr>
<td>5</td>
<td>Channel-attachment (CTCA) major node definition statements and operands</td>
<td>85</td>
</tr>
<tr>
<td>6</td>
<td>Channel-attachment (MPC) major node definition statements and operands</td>
<td>86</td>
</tr>
<tr>
<td>7</td>
<td>Channel-attachment (NCP) major node definition statements and operands</td>
<td>88</td>
</tr>
<tr>
<td>8</td>
<td>Rules for multiple definition of resources</td>
<td>106</td>
</tr>
<tr>
<td>9</td>
<td>Examples of how LUALIAS values are merged</td>
<td>108</td>
</tr>
<tr>
<td>10</td>
<td>Resource types for NQNMODE merges</td>
<td>110</td>
</tr>
<tr>
<td>11</td>
<td>Matrix for NQNMODE merges, part 1</td>
<td>111</td>
</tr>
<tr>
<td>12</td>
<td>Matrix for NQNMODE merges, part 2</td>
<td>112</td>
</tr>
<tr>
<td>13</td>
<td>Notes for NQNMODE merges tables</td>
<td>112</td>
</tr>
<tr>
<td>14</td>
<td>Cross-Domain resource major node definition statements and operands</td>
<td>114</td>
</tr>
<tr>
<td>15</td>
<td>How NQNMODE relates to CDRSCs</td>
<td>137</td>
</tr>
<tr>
<td>16</td>
<td>Cross-Domain resource manager major node definition statement and operands</td>
<td>146</td>
</tr>
<tr>
<td>17</td>
<td>External communication adapter major node—LAN or ATM LAN emulation connections, subarea node</td>
<td>174</td>
</tr>
<tr>
<td>18</td>
<td>External communication adapter major node—LAN, ATM LAN emulation, or Enterprise Extender connections, peripheral node</td>
<td>176</td>
</tr>
<tr>
<td>19</td>
<td>External communication adapter major node—ATM native connection, port</td>
<td>180</td>
</tr>
<tr>
<td>20</td>
<td>External communication adapter major node—ATM native connection, TGs over PVCs</td>
<td>181</td>
</tr>
<tr>
<td>21</td>
<td>External communication adapter major node—ATM native connection, TGs over SVCs</td>
<td>183</td>
</tr>
<tr>
<td>22</td>
<td>External communication adapter major node—ATM native connection, TGs to connection networks</td>
<td>184</td>
</tr>
<tr>
<td>23</td>
<td>XCA AUTOGEN without user-supplied seeds</td>
<td>207</td>
</tr>
<tr>
<td>24</td>
<td>XCA AUTOGEN with single-character user-supplied seeds</td>
<td>208</td>
</tr>
<tr>
<td>25</td>
<td>XCA AUTOGEN with 3-character user-supplied seeds</td>
<td>208</td>
</tr>
<tr>
<td>26</td>
<td>XCA AUTOGEN with number of statements greater than 4096</td>
<td>208</td>
</tr>
<tr>
<td>27</td>
<td>Local non-SNA major node definition statements and operands</td>
<td>274</td>
</tr>
<tr>
<td>28</td>
<td>LU group major node definition statements and operands</td>
<td>288</td>
</tr>
<tr>
<td>29</td>
<td>Model major node definition statements and operands</td>
<td>340</td>
</tr>
<tr>
<td>30</td>
<td>XCF model PU operands used with TRLE</td>
<td>357</td>
</tr>
<tr>
<td>31</td>
<td>NCP major node statements and operands</td>
<td>415</td>
</tr>
<tr>
<td>32</td>
<td>NCP major node statements and operands</td>
<td>429</td>
</tr>
<tr>
<td>33</td>
<td>NCP line definition dependency for coding VTAMTOPO keyword</td>
<td>548</td>
</tr>
<tr>
<td>34</td>
<td>Switched major node definition statements and operands</td>
<td>552</td>
</tr>
<tr>
<td>35</td>
<td>Transport resource list major node statements and operands</td>
<td>659</td>
</tr>
<tr>
<td>36</td>
<td>Definition statement for routes</td>
<td>677</td>
</tr>
<tr>
<td>37</td>
<td>Definition statements for dynamic update NCP paths</td>
<td>682</td>
</tr>
<tr>
<td>38</td>
<td>Definition statements for dynamic update VTAM paths</td>
<td>688</td>
</tr>
<tr>
<td>39</td>
<td>Definition statements for dynamic reconfiguration with VARY DRDS (ADD operation)</td>
<td>695</td>
</tr>
<tr>
<td>40</td>
<td>Definition statements for dynamic reconfiguration with VARY DRDS (DELETE operation)</td>
<td>696</td>
</tr>
<tr>
<td>41</td>
<td>Definition statements for dynamic reconfiguration with VARY DRDS (MOVE operation)</td>
<td>697</td>
</tr>
<tr>
<td>42</td>
<td>Definition statements for ADJSSCP table</td>
<td>703</td>
</tr>
<tr>
<td>43</td>
<td>Definition statements for ADJCLUST table</td>
<td>714</td>
</tr>
<tr>
<td>44</td>
<td>Definition statement for transmission group profile (TGP)</td>
<td>720</td>
</tr>
<tr>
<td>45</td>
<td>CAPACITY values when displayed</td>
<td>722</td>
</tr>
<tr>
<td>46</td>
<td>Definition statements for APPINTOSA table</td>
<td>732</td>
</tr>
<tr>
<td>47</td>
<td>Definition statements for SATOAPPN table</td>
<td>736</td>
</tr>
<tr>
<td>48</td>
<td>Definition statements for COSMAP table</td>
<td>740</td>
</tr>
<tr>
<td>49</td>
<td>Definition statement for network node server list</td>
<td>743</td>
</tr>
<tr>
<td>50</td>
<td>Definition statements for GRPREFS table</td>
<td>749</td>
</tr>
<tr>
<td>51</td>
<td>Start options that define SSCP or CP characteristics</td>
<td>775</td>
</tr>
<tr>
<td>52</td>
<td>Start options that affect VTAM initialization</td>
<td>776</td>
</tr>
<tr>
<td>53</td>
<td>Start options that define storage usage</td>
<td>776</td>
</tr>
<tr>
<td>54</td>
<td>Start options that affect session control</td>
<td>777</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 2000, 2015
<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>55</td>
<td>Start options that affect connectivity</td>
<td>779</td>
</tr>
<tr>
<td>56</td>
<td>Start options that affect session security</td>
<td>780</td>
</tr>
<tr>
<td>57</td>
<td>Start options that affect messages</td>
<td>780</td>
</tr>
<tr>
<td>58</td>
<td>Start options that affect recording and statistics</td>
<td>782</td>
</tr>
<tr>
<td>59</td>
<td>Start options that affect traces and dumps</td>
<td>782</td>
</tr>
<tr>
<td>60</td>
<td>Start options that define buffer pools</td>
<td>783</td>
</tr>
<tr>
<td>61</td>
<td>Start options that affect performance and tuning</td>
<td>783</td>
</tr>
<tr>
<td>62</td>
<td>Start options that define APPN characteristics</td>
<td>786</td>
</tr>
<tr>
<td>63</td>
<td>Start options that were zappable constants</td>
<td>789</td>
</tr>
<tr>
<td>64</td>
<td>Start options that define HPR and Enterprise Extender (EE) characteristics</td>
<td>791</td>
</tr>
<tr>
<td>65</td>
<td>Start options that define network management characteristics</td>
<td>792</td>
</tr>
<tr>
<td>66</td>
<td>Start options that define coupling facility and SYSPLEX characteristics</td>
<td>793</td>
</tr>
<tr>
<td>67</td>
<td>VTAM buffer pools</td>
<td>805</td>
</tr>
<tr>
<td>68</td>
<td>IBM-supplied VTAM buffer pool values</td>
<td>809</td>
</tr>
<tr>
<td>69</td>
<td>IOBUF buffer size and number of buffers per page</td>
<td>810</td>
</tr>
<tr>
<td>70</td>
<td>Node type functional summary</td>
<td>793</td>
</tr>
<tr>
<td>71</td>
<td>XNETALS matrix</td>
<td>978</td>
</tr>
<tr>
<td>72</td>
<td>Associated LU table macroinstructions and operands</td>
<td>982</td>
</tr>
<tr>
<td>73</td>
<td>Subarea CoS macroinstructions and operands</td>
<td>986</td>
</tr>
<tr>
<td>74</td>
<td>APPN CoS definition statements and operands</td>
<td>990</td>
</tr>
<tr>
<td>75</td>
<td>Default values for LINEROW CAPACITY operand</td>
<td>993</td>
</tr>
<tr>
<td>76</td>
<td>Default values for LINEROW COSTBYTE operand</td>
<td>995</td>
</tr>
<tr>
<td>77</td>
<td>Default values for LINEROW COSTTIME operand</td>
<td>996</td>
</tr>
<tr>
<td>78</td>
<td>Default values for LINEROW PDELAY operand</td>
<td>1000</td>
</tr>
<tr>
<td>79</td>
<td>Default values for LINEROW WEIGHT operand</td>
<td>1004</td>
</tr>
<tr>
<td>80</td>
<td>Interpret table macroinstructions and operands</td>
<td>1017</td>
</tr>
<tr>
<td>81</td>
<td>Logon mode table macroinstructions and operands</td>
<td>1026</td>
</tr>
<tr>
<td>82</td>
<td>Exponent and mantissa values for each RU size</td>
<td>1043</td>
</tr>
<tr>
<td>83</td>
<td>Message-flooding prevention table macroinstructions and operands</td>
<td>1046</td>
</tr>
<tr>
<td>84</td>
<td>Model name table macroinstructions and operands</td>
<td>1054</td>
</tr>
<tr>
<td>85</td>
<td>Definition statements for SAMAP table</td>
<td>1058</td>
</tr>
<tr>
<td>86</td>
<td>Session awareness data filter macroinstructions and operands</td>
<td>1061</td>
</tr>
<tr>
<td>87</td>
<td>SAW sense filter definition statements and operands</td>
<td>1066</td>
</tr>
<tr>
<td>88</td>
<td>Description of the IBM-supplied USS tables</td>
<td>1070</td>
</tr>
<tr>
<td>89</td>
<td>VTAM messages search order for VTAM messages</td>
<td>1071</td>
</tr>
<tr>
<td>90</td>
<td>VTAM messages search order for a program operator</td>
<td>1072</td>
</tr>
<tr>
<td>91</td>
<td>VTAM commands search order for VTAM commands</td>
<td>1072</td>
</tr>
<tr>
<td>92</td>
<td>Input translation tables search order</td>
<td>1073</td>
</tr>
<tr>
<td>93</td>
<td>USS table macroinstructions and operands</td>
<td>1073</td>
</tr>
<tr>
<td>94</td>
<td>Valid character strings for message definition</td>
<td>1079</td>
</tr>
<tr>
<td>95</td>
<td>USS messages and VTAM message equivalents</td>
<td>1099</td>
</tr>
<tr>
<td>96</td>
<td>LOGON command and operands</td>
<td>1103</td>
</tr>
<tr>
<td>97</td>
<td>LOGOFF command and operands</td>
<td>1108</td>
</tr>
<tr>
<td>98</td>
<td>IBMTEST command and operands</td>
<td>1113</td>
</tr>
<tr>
<td>99</td>
<td>Connected switched PU report</td>
<td>1373</td>
</tr>
</tbody>
</table>
About this document

This document presents the meaning and coding syntax of Virtual Telecommunications Access Method (VTAM®) resource definition statements and operands, NCP resource definition statements and operands that affect VTAM, start options, and macroinstructions and operands for user-defined tables.

To plan for the tasks of installing VTAM and defining resources to it, see [z/OS Communications Server: New Function Summary](#).

When you are ready to install VTAM and define resources to it, you can find the broader, more conceptual information about installation and definition in the [z/OS Communications Server: SNA Network Implementation Guide](#).

This document contains the specifics of resource definition.

The information in this document includes descriptions of support for both IPv4 and IPv6 networking protocols. Unless explicitly noted, descriptions of IP protocol support concern IPv4. IPv6 support is qualified within the text.

Who should read this document

This document is intended to help system programmers define resources to VTAM.

How this document is organized

This document consists of the following topics and appendixes:

- [Chapter 1, “Introduction to VTAM resource definition,” on page 1](#) describes the symbols and coding conventions used for the definition statements and macroinstructions described in this information. Descriptions of the sift-down effect and each type of major node are also included.

- [“Major nodes” on page 2](#) describes the VTAM definition statements used to define major nodes. Descriptions of NCP definition statements and their operands are given only when they affect VTAM.

- [Chapter 3, “Routing and dynamic reconfiguration,” on page 677](#) describes the VTAM definition statements used, in general, to define routes and to dynamically reconfigure and dynamically change resources.

- [Chapter 4, “VTAM start options,” on page 755](#) describes VTAM start options and how you can code them to set various conditions that are to exist while VTAM is running.

- [Chapter 5, “User-defined tables and data filter,” on page 981](#) describes how to create or modify the associated LU table, Class-of-Service table, CMIP services directory definition file, interpret table, logon mode table, message-flooding prevention table, model name table, and unformatted system services (USS) table. It also describes how to code the session awareness (SAW) data filter.


- [Appendix B, “Examples of USS command conversion,” on page 1369](#) contains examples of how to code USS tables to convert commands.
• Appendix C, “VTAMTOPO filtering option reporting,” on page 1373 summarizes the results of using the VTAMTOPO filtering option for reporting a switched PU under an NCP.
• Appendix E, “Architectural specifications,” on page 1401 lists documents that provide architectural specifications for the SNA Protocol.
• Appendix F, "Accessibility," on page 1403 describes accessibility features to help users with physical disabilities.
• “Notices” on page 1407 contains notices and trademarks used in this document.
• “Bibliography” on page 1417 contains descriptions of the documents in the z/OS® Communications Server library.

How to use this document

You can use this document as a reference for specific operator commands and as background reading to help understand how to operate VTAM.

To use this document, you should be familiar with the basic concepts of telecommunication, SNA, and VTAM.

Determining whether a publication is current

As needed, IBM updates its publications with new and changed information. For a given publication, updates to the hardcopy and associated BookManager® softcopy are usually available at the same time. Sometimes, however, the updates to hardcopy and softcopy are available at different times. The following information describes how to determine if you are looking at the most current copy of a publication:

• At the end of a publication's order number there is a dash followed by two digits, often referred to as the dash level. A publication with a higher dash level is more current than one with a lower dash level. For example, in the publication order number GC28-1747-07, the dash level 07 means that the publication is more current than previous levels, such as 05 or 04.

• If a hardcopy publication and a softcopy publication have the same dash level, it is possible that the softcopy publication is more current than the hardcopy publication. Check the dates shown in the Summary of Changes. The softcopy publication might have a more recently dated Summary of Changes than the hardcopy publication.

• To compare softcopy publications, you can check the last 2 characters of the publication’s file name (also called the book name). The higher the number, the more recent the publication. Also, next to the publication titles in the CD-ROM booklet and the readme files, there is an asterisk (*) that indicates whether a publication is new or changed.

How to contact IBM service

For immediate assistance, visit this website: http://www.software.ibm.com/network/commserver/support/

Most problems can be resolved at this website, where you can submit questions and problem reports electronically, and access a variety of diagnosis information.

For telephone assistance in problem diagnosis and resolution (in the United States or Puerto Rico), call the IBM Software Support Center anytime (1-800-IBM-SERV). You will receive a return call within 8 business hours (Monday – Friday, 8:00 a.m. – 5:00 p.m., local customer time).
Conventions and terminology that are used in this document

Commands in this book that can be used in both TSO and z/OS UNIX environments use the following conventions:

- When describing how to use the command in a TSO environment, the command is presented in uppercase (for example, NETSTAT).
- When describing how to use the command in a z/OS UNIX environment, the command is presented in bold lowercase (for example, netstat).
- When referring to the command in a general way in text, the command is presented with an initial capital letter (for example, Netstat).

All the exit routines described in this document are installation-wide exit routines. The installation-wide exit routines also called installation-wide exits, exit routines, and exits throughout this document.

The TPF logon manager, although included with VTAM, is an application program; therefore, the logon manager is documented separately from VTAM.

Samples used in this book might not be updated for each release. Evaluate a sample carefully before applying it to your system.

**Note:** In this information, you might see the following Shared Memory Communications over Remote Direct Memory Access (SMC-R) terminology:

- RDMA network interface card (RNIC), which is used to refer to the IBM® 10GbE RoCE Express® feature.
- Shared RoCE environment, which means that the 10GbE RoCE Express feature operates on an IBM z13™ (z13) or later system, and that the feature can be used concurrently, or shared, by multiple operating system instances. The RoCE Express feature is considered to operate in a shared RoCE environment even if you use it with a single operating system instance.

For definitions of the terms and abbreviations that are used in this document, you can view the latest IBM terminology at the IBM Terminology website.

Clarification of notes

Information traditionally qualified as Notes is further qualified as follows:

**Note** Supplemental detail

**Tip** Offers shortcuts or alternative ways of performing an action; a hint

**Guideline** Customary way to perform a procedure

**Rule** Something you must do; limitations on your actions

**Restriction** Indicates certain conditions are not supported; limitations on a product or facility
How to read a syntax diagram

This section describes how to read the syntax diagrams used in this book.

- Read the diagrams from left-to-right, top-to-bottom, following the main path line. Each diagram begins on the left with double arrowheads (⇒⇒) and ends on the right with two arrowheads facing each other (⇐⇐).

```
⇒⇒ Syntax Diagram ┌────────────────┐
```

- If a diagram is longer than one line, the first line ends with a single arrowhead (⇒) and the second line begins with a single arrowhead.

```
⇒⇒ First Line ┌─ OPERAND1 ─ OPERAND2 ─ OPERAND3 ─ OPERAND4 ─ OPERAND5 ┐
⇒⇒ Second Line └────────────────────────────────────────┘
```

- Required operands and values appear on the main path line.

```
⇒⇒ REQUIRED_OPERAND ┌────────────────┐
```

You must code required operands and values.

If there is more than one mutually exclusive required operand or value to choose from, they are stacked vertically in alphanumeric order.
• Optional operands and values appear below the main path line.

You can choose not to code optional operands and values. If there is more than one mutually exclusive optional operand or value to choose from, they are stacked vertically in alphanumeric order below the main path line.

• An arrow returning to the left above an operand or value on the main path line means that the operand or value can be repeated. The comma means that each operand or value must be separated from the next by a comma.

• An arrow returning to the left above a group of operands or values means more than one can be selected, or a single one can be repeated.
A word in all uppercase is an operand or value you must spell exactly as shown. In this example, you must code OPERAND.

**Note:** VTAM and IP commands are not case sensitive. You can code them in uppercase or lowercase. If the operand is shown in both uppercase and lowercase, the uppercase portion is the abbreviation (for example, OPERand).

If an operand or value can be abbreviated, the abbreviation is described in the text associated with the syntax diagram.

- If a diagram shows a character that is not alphanumeric (such as parentheses, periods, commas, and equal signs), you must code the character as part of the syntax. In this example, you must code OPERAND=(001,0.001).

- If a diagram shows a blank space, you must code the blank space as part of the syntax. In this example, you must code OPERAND=(001 FIXED).

- Default operands and values appear above the main path line. VTAM uses the default if you omit the operand entirely.
A word in all lowercase italics is a *variable*. Where you see a variable in the syntax, you must replace it with one of its allowable names or values, as defined in the text.

References to syntax notes appear as numbers enclosed in parentheses above the line. Do not code the parentheses or the number.

**Notes:**

1. An example of a syntax note.

Some diagrams contain *syntax fragments*, which serve to break up diagrams that are too long, too complex, or too repetitious. Syntax fragment names are in mixed case and are shown in the diagram and in the heading of the fragment. The fragment is placed below the main diagram.
Prerequisite and related information

z/OS Communications Server function is described in the z/OS Communications Server library. Descriptions of those documents are listed in "Bibliography" on page 1417 in the back of this document.

Required information

Before using this product, you should be familiar with TCP/IP, VTAM, MVS™, and UNIX System Services.

Softcopy information

Softcopy publications are available in the following collection.

<table>
<thead>
<tr>
<th>Titles</th>
<th>Order Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM System z Redbooks Collection</td>
<td>SK3T-7876</td>
<td>The IBM Redbooks® publications selected for this CD series are taken from the IBM Redbooks inventory of over 800 books. All the Redbooks publications that are of interest to the System z® platform professional are identified by their authors and are included in this collection. The System z subject areas range from e-business application development and enablement to hardware, networking, Linux, solutions, security, parallel sysplex, and many others. For more information about the Redbooks publications, see <a href="http://www-03.ibm.com/systems/z/os/zos/zfavorites/">http://www-03.ibm.com/systems/z/os/zos/zfavorites/</a></td>
</tr>
</tbody>
</table>

Other documents

This information explains how z/OS references information in other documents.

When possible, this information uses cross-document links that go directly to the topic in reference using shortened versions of the document title. For complete titles and order numbers of the documents for all products that are part of z/OS, see [z/OS Information Roadmap](SA23-2299). The Roadmap describes what level of documents are supplied with each release of z/OS Communications Server, and also describes each z/OS publication.

To find the complete z/OS library, visit the [z/OS library in IBM Knowledge Center](www.ibm.com/support/knowledgecenter/SSLTBW/welcome).

Relevant RFCs are listed in an appendix of the IP documents. Architectural specifications for the SNA protocol are listed in an appendix of the SNA documents.

The following table lists documents that might be helpful to readers.

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Number</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>SNA Formats</td>
<td>GA27-3136</td>
</tr>
<tr>
<td>TCP/IP Tutorial and Technical Overview</td>
<td>GG24-3376</td>
</tr>
<tr>
<td>Understanding LDAP</td>
<td>SG24-4986</td>
</tr>
<tr>
<td>Z/OS Cryptographic Services System SSL Programming</td>
<td>SC14-7495</td>
</tr>
<tr>
<td>Z/OS IBM Tivoli Directory Server Administration and Use for z/OS</td>
<td>SC23-6788</td>
</tr>
<tr>
<td>Z/OS JES2 Initialization and Tuning Guide</td>
<td>SA32-0991</td>
</tr>
<tr>
<td>Z/OS Problem Management</td>
<td>SC23-6844</td>
</tr>
<tr>
<td>Z/OS MVS Diagnosis: Reference</td>
<td>GA32-0904</td>
</tr>
<tr>
<td>Z/OS MVS Diagnosis: Tools and Service Aids</td>
<td>GA32-0905</td>
</tr>
<tr>
<td>Z/OS MVS Using the Subsystem Interface</td>
<td>SA38-0679</td>
</tr>
<tr>
<td>Z/OS V2R1 Program Directory</td>
<td>GI11-9848</td>
</tr>
<tr>
<td>Z/OS UNIX System Services Command Reference</td>
<td>SA23-2280</td>
</tr>
<tr>
<td>Z/OS UNIX System Services Planning</td>
<td>GA32-0884</td>
</tr>
<tr>
<td>Z/OS UNIX System Services Programming: Assembler Callable Services Reference</td>
<td>SA23-2281</td>
</tr>
<tr>
<td>Z/OS UNIX System Services User’s Guide</td>
<td>SA23-2279</td>
</tr>
<tr>
<td>Z/OS XL C/C++ Runtime Library Reference</td>
<td>SC14-7314</td>
</tr>
<tr>
<td>z/Enterprise System and System z10 OSA-Express Customer’s Guide and Reference</td>
<td>SA22-7935</td>
</tr>
</tbody>
</table>

**Redbooks publications**

The following Redbooks publications might help you as you implement z/OS Communications Server.

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM z/OS V2R1 Communications Server TCP/IP Implementation, Volume 1: Base Functions, Connectivity, and Routing</td>
<td>SG24-8096</td>
</tr>
<tr>
<td>IBM z/OS V2R1 Communications Server TCP/IP Implementation, Volume 2: Standard Applications</td>
<td>SG24-8097</td>
</tr>
<tr>
<td>IBM z/OS V2R1 Communications Server TCP/IP Implementation, Volume 3: High Availability, Scalability, and Performance</td>
<td>SG24-8098</td>
</tr>
<tr>
<td>IBM z/OS V2R1 Communications Server TCP/IP Implementation, Volume 4: Security and Policy-Based Networking</td>
<td>SG24-8099</td>
</tr>
<tr>
<td>IBM Communication Controller Migration Guide</td>
<td>SG24-6298</td>
</tr>
<tr>
<td>IP Network Design Guide</td>
<td>SG24-2580</td>
</tr>
<tr>
<td>Managing OS/390 TCP/IP with SNMP</td>
<td>SG24-5866</td>
</tr>
<tr>
<td>Migrating Subarea Networks to an IP Infrastructure Using Enterprise Extender</td>
<td>SG24-5957</td>
</tr>
</tbody>
</table>
Where to find related information on the Internet

**z/OS**

This site provides information about z/OS Communications Server release availability, migration information, downloads, and links to information about z/OS technology


**z/OS Internet Library**

Use this site to view and download z/OS Communications Server documentation


**IBM Communications Server product**

The primary home page for information about z/OS Communications Server


**IBM Communications Server product support**

Use this site to submit and track problems and search the z/OS Communications Server knowledge base for Technotes, FAQs, white papers, and other z/OS Communications Server information


**IBM Communications Server performance information**

This site contains links to the most recent Communications Server performance reports.


**IBM Systems Center publications**

Use this site to view and order Redbooks publications, Redpapers™, and Technotes


**IBM Systems Center flashes**

Search the Technical Sales Library for Techdocs (including Flashes, presentations, Technotes, FAQs, white papers, Customer Support Plans, and Skills Transfer information)


**Tivoli NetView for z/OS**

Use this site to view and download product documentation about Tivoli® NetView® for z/OS
RFCs

Search for and view Request for Comments documents in this section of the Internet Engineering Task Force website, with links to the RFC repository and the IETF Working Groups web page

http://www.ietf.org/rfc.html

Internet drafts

View Internet-Drafts, which are working documents of the Internet Engineering Task Force (IETF) and other groups, in this section of the Internet Engineering Task Force website

http://www.ietf.org/ID.html

Information about web addresses can also be found in information APAR III1334.

Note: Any pointers in this publication to websites are provided for convenience only and do not serve as an endorsement of these websites.

DNS websites

For more information about DNS, see the following USENET news groups and mailing addresses:

USENET news groups

comp.protocols.dns.bind

BIND mailing lists

https://lists.isc.org/mailman/listinfo

BIND Users

- Subscribe by sending mail to bind-users-request@isc.org.
- Submit questions or answers to this forum by sending mail to bind-users@isc.org.

BIND 9 Users (This list might not be maintained indefinitely.)

- Subscribe by sending mail to bind9-users-request@isc.org.
- Submit questions or answers to this forum by sending mail to bind9-users@isc.org.

The z/OS Basic Skills Information Center

The z/OS Basic Skills Information Center is a web-based information resource intended to help users learn the basic concepts of z/OS, the operating system that runs most of the IBM mainframe computers in use today. The Information Center is designed to introduce a new generation of Information Technology professionals to basic concepts and help them prepare for a career as a z/OS professional, such as a z/OS systems programmer.

Specifically, the z/OS Basic Skills Information Center is intended to achieve the following objectives:

- Provide basic education and information about z/OS without charge
- Shorten the time it takes for people to become productive on the mainframe
- Make it easier for new people to learn z/OS
To access the z/OS Basic Skills Information Center, open your web browser to the following website, which is available to all users (no login required):

Summary of changes

This document contains terminology, maintenance, and editorial changes, including changes to improve consistency and retrievability. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

Changes made in z/OS Version 2 Release 1, as updated February 2015

This document contains information previously presented in z/OS Communications Server: SNA Resource Definition Reference, SC27-3675-01, which supported z/OS Version 2 Release 1.

Changes made in z/OS Version 2 Release 1, as updated December 2013

This document contains information previously presented in z/OS Communications Server: SNA Resource Definition Reference, SC27-3675-00, which supported z/OS Version 2 Release 1.

Summary of changes for z/OS Version 2 Release 1

For specifics on the enhancements for z/OS Version 2, Release 1, see the following publications:

- z/OS Summary of Message and Interface Changes
- z/OS Introduction and Release Guide
- z/OS Planning for Installation
- z/OS Migration
Chapter 1. Introduction to VTAM resource definition

This topic gives an overview of VTAM resource definition and presents general information on:
- Major nodes
- Routing functions
- Tables
- Coding conventions, formats, and procedures

Defining a single-domain network to VTAM involves defining the following resources:
- Adjacent control point major nodes
- Application program major nodes
- Channel-attachment major nodes
- External communication adapter major nodes
- Local non-SNA major nodes
- Local SNA major nodes
- LU group major nodes
- Model major nodes
- NCP major nodes
- Switched major nodes
- Path statements 1
- Transmission group profiles
- Tables

Defining a multiple-domain network to VTAM also involves defining:
- Adjacent SSCP tables 1
- Cross-domain resource manager major nodes
- Cross-domain resource major nodes 2
- Network node server lists
- Transport resource list

When you use SNA network interconnection, you might also have to redefine existing major node definitions.

VTAM uses tables for routing, to begin and end sessions, and to define messages and certain operator commands and to suppress messages. You can create, replace, or modify these tables while VTAM is running.

VTAM also provides a filter for session awareness (SAW) data that is passed to communication network management (CNM) application programs, such as NetView. You can create, modify or replace the filter while VTAM is running.

1. These definition statements, along with dynamic reconfiguration and dynamic path update, are described in Chapter 3, "Routing and dynamic reconfiguration," on page 677.
2. The cross-domain major node is also used to predefine independent LUs in a boundary-attached subarea network.
After you define a network, you can use dynamic reconfiguration to add, move, or
delete resources attached to an NCP or to add or delete locally attached logical units. In addition, you can dynamically change the values of certain operands.


**Major nodes**

Resources in a VTAM network are classified under major and minor nodes. Each major node can contain one or more minor nodes. Brief descriptions of the major nodes follow.

**Adjacent control point major node**

An adjacent control point major node specifies the name and attributes of an adjacent APPN node with which a node can establish type 2.1 connections.

**Application program major node**

An application program major node contains application programs as minor nodes. There can be one or more application programs in each application program major node.

**Channel-attachment major node**

A channel-attachment major node contains definition statements for one of several types of channel (link) groups. The possible types of link groups are:
- Channel-to-channel support
- Multipath channel support
- Channel-attached NCP support

**Cross-domain resource major node**

A cross-domain resource major node is a logical set of cross-domain resources (CDRSCs). You can define one or more cross-domain resources, each of which is a minor node, in each major node. You can also use the cross-domain resource major node to predefine independent LUs in a boundary-attached subarea network.

**Cross-domain resource manager major node**

A cross-domain resource manager major node is a logical set of cross-domain resource managers (CDRMs). You can define one or more cross-domain resource managers, each of which is a minor node, in each major node.

**External communication adapter major node**

An external communication adapter major node defines a connection (port) between VTAM and a token-bus network, a token-ring network, a CSMA/CD 802.3 LAN, or an FDDI LAN attached through the IBM 3172 Nways interconnect controller or an IBM Open Systems Adapter. The minor nodes are the line groups, lines, and physical units attached to the port.

The XCA major node is also used to define a connection (port) between VTAM and an asynchronous transfer mode (ATM) network attached through an IBM S/390® Open Systems Adapter. In addition, the XCA major node is used to define a connection (port) between VTAM and an Enterprise Extender (EE) network.
Local non-SNA major node
A local non-SNA major node consists of a logical group of channel-attached non-SNA terminals. Each terminal in the group is a minor node. The terminals in a group do not have to all be connected to the same cluster controller.

Local SNA major node
A local SNA major node consists of a logical group of channel-attached SNA terminals. The minor nodes are the physical units and their logical units that are contained in the terminals.

LU group major node
An LU group major node defines one or more model LU groups.

Model major node
A model major node defines dynamic switched definitions for PUs and LUs.

NCP major node
An NCP major node consists of the resources attached to an NCP. The attached resources (lines, physical units, and logical units) are minor nodes.

Switched major node
The switched major node contains definitions for potential switched connections to either a subarea node or a peripheral node.

Two types of resources make up a switched connection:
• A switched line within a switched line group
  Define the line group within an NCP major node, a channel-attachment major node, or an external communication adapter major node.
• A remote physical unit
  The physical unit and its logical units are the minor nodes that you define within a switched major node.

Transport resource list major node
The transport resource list major node describes the connectivity characteristics of a multipath channel connecting two APPN hosts. The characteristics of each multipath channel are defined by a TRLE definition statement. When an adjacent link station is activated, the PU definition statement identifies which TRLE definition statement VTAM uses to route data over the channel.

Routing functions and dynamic reconfiguration
Brief descriptions of routing functions and dynamic reconfiguration and change follow.

Adjacent SSCP tables
The adjacent SSCP table contains one or more lists of SSCPs that can be in session with the VTAM SSCP and can be used to reach destination SSCPs in the same or other networks. You can activate one or more adjacent SSCP tables in each VTAM host.
Because of the dynamic adjacent SSCP table function (see the start option “DYNASSCP start option” on page 840), you do not need to code adjacent SSCP tables to establish cross-domain and cross-network sessions. VTAM dynamically routes session establishment requests to all active adjacent SSCPs until it finds the correct SSCP. However, if you code adjacent SSCP tables, VTAM uses them.

Information on coding the statements that define adjacent SSCP tables is in “Adjacent SSCP table” on page 703.

Path definition statements

Path definition statements are used for two different purposes:

- To define VTAM paths
  
  PATH definition statements represent the routes VTAM takes to communicate with other subarea nodes. One or more PATH definition statements are filed in the VTAM definition library. The name assigned to a path definition set is used to activate that set of paths. More than one path definition set can be filed and activated.
  
  Information on coding the definition statements that define VTAM paths is in “VTAM routes” on page 677.

- For dynamic path update
  
  Dynamic path update allows the VTAM operator to add NCP path definitions or VTAM path definitions dynamically without regenerating and reloading the NCP. If the explicit route (ER) is not currently operative, NCP path definitions can also be replaced dynamically without regenerating and reloading the NCP. Also, a delete function for inoperative ERs is provided for both VTAM and NCP path definitions. You can also use dynamic path update to change path definition parameters such as virtual route (VR) pacing window size for inoperative routes.

  - For VTAM:
    
    The dynamic path specification set for VTAM contains VPATH and PATH definition statements.
    
    The VPATH definition statement identifies the VTAM subarea and the network to which the path update specification set applies in a dynamic path update member. The PATH definition statement defines the explicit routes, virtual routes, and virtual route pacing window size for the path.
    
    Information on coding the definition statements that define VTAM dynamic path update is in “Dynamic path update (VTAM)” on page 688.

  - For NCP:
    
    The dynamic path specification set for an NCP contains NCPPATH and PATH definition statements.
    
    The NCPPATH definition statement identifies the NCP and the network to which the path update specification set applies in a dynamic path update member. The PATH definition statement defines the explicit routes, virtual routes, and virtual route pacing window size for the path.
    
    Information on coding the definition statements that define NCP dynamic path update is in “Dynamic path update (NCP)” on page 682.

Dynamic reconfiguration and dynamic change

Dynamic reconfiguration (DR) allows you to change a network configuration without having to deactivate the affected major node and without having to regenerate the NCP. Dynamic change allows you to change certain operands in
resource definition statements without having to deactivate the major node. With dynamic reconfiguration and dynamic change, you can:

- Add and delete physical units or logical units
- Move a physical unit and its associated logical units
- Change the SDLC link station address for a physical unit
- Change an LU's LOCADDR value
- Modify operands

More information on dynamic reconfiguration and the definition statements that are coded for this function is found in "Dynamic reconfiguration and dynamic change" on page 694.

Transmission group profile

The transmission group profile definition statement defines a set of characteristics for a transmission group (TG). The TG profile is referenced by an adjacent link station definition on a PU in a switched or model major node or on a PU under a non-switched line.

The TG profile is not a resource, but describes the characteristics of a link. When an adjacent link station is activated, VTAM locates the TG profile. If the TG profile cannot be found, default values for a TG profile are used for the link characteristics. See "APPN transmission group profile" on page 720 for more details.

Network node server list

A network node server list indicates the network nodes that can act as servers for an end node. The network node server list is not required, but using it allows you to control which network node is selected by an end node.

An end node becomes aware of adjacent network nodes that can act as its network node server through normal, spontaneous network flows. By default, an end node establishes a CP-CP session with the first network node of which it becomes aware, and this network node acts as the end node's server. There is no predictability in this scheme because it depends on timing.

You might want to control which network node is selected by an end node. For example, you might want to exclude a particular network node from network node server responsibilities; acting as a server does involve some overhead, such as originating search requests or executing domain broadcasts. You might also want to isolate particular end nodes from certain network nodes for security reasons.

See "Network node server list" on page 743 for more details.

User-defined tables and session awareness data filter

Brief descriptions of the user-defined tables and filter follow.

Associated LU table

The associated LU table provides the subsystem with the associated printer information needed to dynamically define the SLU. During session initiation, VTAM passes the information to the primary logical unit (PLU) in the LOGON exit.
See “Associated LU table” on page 981 for information on the macroinstructions that define an associated LU table. See z/OS Communications Server: SNA Network Implementation Guide for information about how the table is used.

Class of Service

You can group routes based on characteristics such as security, transmission priority, and bandwidth of the routes. The characteristics of a particular group determine a Class of Service (CoS). VTAM provides classes of service for subarea and APPN routing.

See “Subarea class-of-service table” on page 985 for information on the macroinstructions that define a subarea CoS table and “APPN Class of Service definitions” on page 990 for information on the statements that define an APPN CoS. See z/OS Communications Server: SNA Network Implementation Guide for a discussion about how classes of service are used.

CMIP services directory definition file

The IBM-supplied directory definition file is required to implement Common Management Information Protocol (CMIP) services. The directory definition file for CMIP services controls the associations that CMIP services on a host has with other CMIP services and with itself. An association is a connection between CMIP services on a host and CMIP Services on another node or between CMIP services on the host and itself.

If you have specified the OSIMGMT=YES start option, a directory definition file is required. The IBM-supplied directory definition file in member ACYDDF of the SYS1.SISTCMIP data set allows unlimited associations with other instances of CMIP services and between CMIP services and itself. If you do not edit the IBM-supplied file, any CMIP application program in any network can have access to CMIP services on this host.

See “CMIP services directory definition file” on page 1008 for information on coding the directory definition file. See z/OS Communications Server: SNA Network Implementation Guide for more information about the directory definition file.

Interpret table

When VTAM processes a formatted Initiate or Terminate request (either received directly from a logical unit or formatted by unformatted system services (USS) from a character-coded logon or logoff), it uses an interpret table to determine which application program is to be notified.

The standard logon command definition should suffice for most networks. See “LOGON command” on page 1103 for coding details. You will have to write your own interpret table only under special circumstances, for instance, if the logon sequence you want to use does not follow the syntax for USS commands. IBM does not supply a default table.

See “Interpret table” on page 1017 for information on coding the macroinstructions that define an interpret table. See z/OS Communications Server: SNA Network Implementation Guide for information about how the table is used.
Logon mode table
A logon mode table contains entries specifying session parameters. VTAM contains an IBM-supplied default logon mode table, ISTINCLM. This table contains only a set of generally accepted session parameters for a basic list of IBM device types. If you need session parameters that are not included in the IBM-supplied default table, you can:
- Modify the IBM-supplied default table.
- Create supplementary tables, which you associate with device-type logical units or application programs by using the MODETAB operand in the statements defining them.

See “Logon mode table” on page 1026 for information on the macroinstructions that define a logon mode table. See z/OS Communications Server: SNA Network Implementation Guide for information about how the table is used.

Message-flooding prevention table
The message-flooding prevention table lists messages that are candidates for flooding suppression. The purpose is to prevent VTAM from issuing duplicate messages in rapid succession, which could flood the operator console without providing additional information.

For each candidate message, the table contains the criteria that must be met before VTAM suppresses duplicate messages. The suppression criteria include the amount of time between the original and subsequent messages, as well as an indication of which variable text fields are to be compared and whether messages are suppressed for the hardcopy log. If the message is reissued within the specified interval and the specified variable text fields contain the same information, VTAM suppresses the message.

See “Message-flooding prevention table” on page 1046 for information on coding the macroinstructions that define a message-flooding prevention table.

Model name table
The model name table provides a model name that is passed to the subsystem in the LOGON exit during session initiation. The subsystem uses the model name while dynamically defining the secondary logical unit (SLU).

See “Model name table” on page 1054 for information on the macroinstructions that define a model name table. See z/OS Communications Server: SNA Network Implementation Guide for information about how the table is used.

Session awareness (SAW) data filter
The SAW data filter reduces the amount of session awareness trace data passed over the communication network management (CNM) interface to a CNM application program. See “Session awareness data filter” on page 1061 for information on the macroinstructions that define a SAW data filter. See z/OS Communications Server: SNA Network Implementation Guide for information about how the filter is used.

Unformatted system services (USS) table
VTAM uses USS tables to define messages and certain operator commands. If you want to change the text or other characteristics of a message or change the syntax...
or default values for a command, you can do so by creating a supplementary USS

table. Only the commands and messages that you want to change need to be
defined in the supplementary USS table.

See “Unformatted system services tables” on page 1069 for information on the

macroinstructions that define a USS table. See z/OS Communications Server: SNA

Network Implementation Guide for information about how the table is used.

Coding VTAM definitions

This information explains the format of definition statements and

macroinstructions and the rules used to code them. For sample definitions to help

you implement VTAM functions in your network, see z/OS Communications

Server: SNA Resource Definition Samples.

The VTAMLST definition source validation is available in VTAM as part of the

VARY ACT command. It allows you to check the syntax of a definition file without

activating the resources in the file. For more information on this function, see z/OS

Communications Server: SNA Operation.

The rules detailed in this information are assembler language rules.

Format of definition statements

NCP and VTAM definition statements have the following format:

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Definition Statement</td>
<td>Required and optional operands</td>
</tr>
</tbody>
</table>

The name field identifies the definition statement or minor node. If you use the

name field, it must contain 1–8 characters of the following types:

First character Alphabetical (A–Z); the national characters @, #, or $; for APPL or

CDRSC definition statements, the wildcard character ?.

Second to eighth characters Alphabetical (A–Z); numerical (0–9); the national characters @, #, or

$; for APPL or CDRSC definition statements, the wildcard

characters * or ?.

Note: Any name fields coded or referenced in VTAM API macroinstructions must

be in uppercase.

You can code MVS system symbols in the name field of definition statements. See

“Using MVS system symbols in VTAM definition statements and

macroinstructions” on page 15 for information about using MVS system symbols in

VTAM macroinstructions.

If coded, the name field must begin in column 1 of the definition statement and

must be followed by one or more blanks. Together, the name field and definition

statement field must fit in columns 1–70 of a line. Although the name might be

indicated as optional, you should code it because VTAM uses it in operator

messages referring to the resource defined. Also, resources without names are not

saved in checkpoint data sets. When a resource is “checkpointed”, information

about the resource is saved in a data set and later read by VTAM when it is

restated.
You can use network-qualified names to ensure that resources that are LU-LU session-capable within an SNA network interconnection (SNI) have unique names. Network-qualified names are in the form netid.resource_name, where netid identifies the name of the network in which the resource resides. Two resources residing in different networks can have the same resource name, but because their network-qualified names have unique network identifiers, no name conflict exists.

The definition statement field, which must be preceded and followed by one or more blanks, identifies the definition statement. All definition statements can be in mixed case, even though they will be translated and processed by VTAM in uppercase.

The operands field contains operands, which can be coded in any order and must be separated by commas. The operand field ends with one or more blanks placed after the last operand. In most definition statements, operands are used in the operand field. Operands are followed by an equal sign (=) and the operand value. The operand value can be a single value or a list of values. If it is a list of values, the values must be separated by commas and the list must be enclosed in parentheses. If the operand fields are to be continued, the continuation character must be placed in column 72. If the value is specified as name, it must follow the preceding rules for name.

Note: You can code one or more MVS system symbols in VTAM definition statements in VTAMLST. If you do, the preceding rules for the name, definition statement, and operands fields still apply, even before the symbols are resolved. For example, as shown below, a name field containing MVS system symbols must begin in column 1 on the same line as the definition statement to which it applies. The name field must be followed by one or more blanks. Together, the name field and the definition statement field must not go beyond column 70.

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...&USER1.&USER2.&USER3.&USER4.&USER5.&USER6.&USER7.&USER8. APPL X AUTH=(ACQ,PASS),ENCR=REQD

After MVS substitutes the values for the symbols, the resolved name must be 1–8 characters in length. If the symbols used above are defined in MVS to have the following text:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;USER1</td>
<td>A</td>
</tr>
<tr>
<td>&amp;USER2</td>
<td>P</td>
</tr>
<tr>
<td>&amp;USER3</td>
<td>P</td>
</tr>
<tr>
<td>&amp;USER4</td>
<td>L</td>
</tr>
<tr>
<td>&amp;USER5</td>
<td>N</td>
</tr>
<tr>
<td>&amp;USER6</td>
<td>A</td>
</tr>
<tr>
<td>&amp;USER7</td>
<td>M</td>
</tr>
</tbody>
</table>
the name field resolves to APPLNAME after MVS substitutes the values in place of the symbols in VTAMLST.

See “Using MVS system symbols in VTAM definition statements and macroinstructions” on page 15 for information about using MVS system symbols in VTAM definition statements.

Format of macroinstructions

VTAM macroinstructions, which define tables and the SAW data filter, have the following format:

<table>
<thead>
<tr>
<th>Name</th>
<th>Macroinstruction</th>
<th>Operands</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Macroinstruction</td>
<td>Required and optional operands</td>
</tr>
</tbody>
</table>

The name field identifies the macroinstruction. If you use a name, it must contain 1–8 characters in the following format:

First character

- Alphabetical (A–Z) or the national characters @, #, or $

Second to eighth character

- Alphabetical (A–Z), numerical (0–9), or the national characters @, #, or $.

You can code MVS system symbols in the name field of the macroinstructions that define the following user-definable tables:

- APPN Class of Service (CoS) definitions
- APPN-to-subarea Class of Service mapping table
- Associated LU table
- Message-flooding prevention table
- Model name table
- Subarea-to-APPN Class of Service mapping table

See “Using MVS system symbols in VTAM definition statements and macroinstructions” on page 15 for information about using MVS system symbols in VTAM macroinstructions.

If coded, the name field must begin in column 1 of the macroinstruction, and must be followed by one or more blanks. Together, the name field and macroinstruction field must fit in columns 1–70 of a line. Although the name might be optional, you should code it, because VTAM uses it in operator messages referring to the resource defined.

The macroinstruction field, which must be preceded and followed by one or more blanks, identifies the macroinstruction.

The operands field contains operands, which can be coded in any order, and must be separated by commas. The operand field ends with one or more blanks placed after the last operand. In most macroinstructions, operands are used in the operand field. Operands are followed by an equal sign (=) and the operand value.
The operand value can be a single value or a list of values. If it is a list of values, the values must be separated by commas and the list must be enclosed in parentheses. If the operand fields are to be continued, the continuation character must be placed in column 72. If the value is specified as name, it must follow the rules given above for a name.

**Note:** You can code one or more MVS system symbols in VTAM macroinstructions in VTAMLST. If you do, the preceding rules for the name, macroinstruction, and operands fields still apply, even before the symbols are resolved. For example, as shown below, a name field containing MVS system symbols must begin in column 1 on the same line as the macroinstruction to which it applies. The name field must be followed by one or more blanks. Together, the name field and the macroinstruction field must not go beyond column 70, even when there are no operands to be coded on a macroinstruction, as in the following example:

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
&USER1.&USER2.&USER3.&USER4.&USER5.&USER6.&USER7.&USER8. MDLTAB
```

After MVS substitutes the values for the symbols, the resolved name must be 1–8 characters in length. If the symbols used above are defined in MVS to have the following text:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;USER1</td>
<td>M</td>
</tr>
<tr>
<td>&amp;USER2</td>
<td>O</td>
</tr>
<tr>
<td>&amp;USER3</td>
<td>D</td>
</tr>
<tr>
<td>&amp;USER4</td>
<td>L</td>
</tr>
<tr>
<td>&amp;USER5</td>
<td>N</td>
</tr>
<tr>
<td>&amp;USER6</td>
<td>A</td>
</tr>
<tr>
<td>&amp;USER7</td>
<td>M</td>
</tr>
<tr>
<td>&amp;USER8</td>
<td>E</td>
</tr>
</tbody>
</table>

the name field resolves to MODLNAME after MVS substitutes the values in place of the symbols in VTAMLST.

See "Using MVS system symbols in VTAM definition statements and macroinstructions" on page 15 for information about using MVS system symbols in VTAM macroinstructions.

**Rules for coding start option lists and configuration lists**

The coding of ATCCONyy (configuration list members) and ATCSTRyy (start list members) in the VTAM definition library are governed by the same syntax rules, which are different from the rules governing the coding of NCP and VTAM definition statements and VTAM macroinstructions.
Code start option lists and configuration lists as one or more 80-byte records in the following format:

- **position**
  - position
  - 1
  - [...]item[,item]...

- **item**
  - Represents a start option in a start option list or a resource in a configuration list. Separate items with commas. Use no intervening blanks between items.

- **c**
  - Reserved for the continuation indicator. The continuation indicator is optional, but is ignored. VTAM ignores positions 72–80.

- **(,)**
  - Commas are optional after the last item on a line, but required between items if coded on the same line.

The following information shows a sample start list, ATCSTRyy. The first line is a scale to show you column positions. Every tenth column position is indicated by a number.

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...

SSCPID=01
SSCPNAME=SSCP1A,
  DYNASSCP=YES
SSCPDYN=YES,HOSTSA=1
  * this is a full line comment
NETID=NETA
```

One start list cannot reference another, and only an operator can enter LIST=yy on the start command to use this start list. One configuration list cannot reference another, and either an operator can enter CONFIG=yy on the start command, or the start option CONFIG=yy can be coded in a start list to use this configuration list.

The following information shows a sample configuration list, ATCCONyy. The first line is a scale to show you column positions. Every tenth column position is indicated by a number.

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...

LOCSNA1,
LOCN1
NCI1,
APPL1,
  * this is a full line comment
CDRM1,CDRM2
```

**Rules:** Depending on the number of items, two or more records might be required for the start option list or configuration list. Observe the following rules if more than one record is required:

1. You can code data through position 71, even if you are in the middle of an item. The remainder of the item can be continued in the next record.
2. If you do not code data through position 71, end the data in one record with an item, and begin the next record with the new item.
3. To create a comment line, enter an asterisk(*) in position 1 of any record.
4. Blank lines are not allowed. There must be an (*) in column 1 for a record to be treated as a comment line.
Rules for coding continued lines

Code VTAM and NCP definition statements and VTAM macroinstructions—including the name, definition statement or macroinstruction, and operand fields—in columns 1–71 of a line.

**Note:** The name field together with the definition statement field or macroinstruction field must fit in columns 1–70.

You can continue a definition statement or macroinstruction that exceeds 71 columns on one or more additional lines. When you need to continue on another line, the following rules apply:

1. Code the definition statement or macroinstruction in one of the following ways:
   - Up through column 71
   - Through any completed operand, stopping after the comma that separates the operand from those that follow
2. If the definition statement field or the macroinstruction field ends in column 70, leave a blank in column 71.
3. Enter a nonblank continuation character in column 72. The continuation character is not considered part of the definition statement.
4. Begin coding or continue coding operands in column 16 of the next line. Columns 1–15 must be blank. A continuation line that begins in column 17 or beyond is ignored, with one exception: a quoted string that needs to be continued on the next line can have a blank in column 16 of the next line if a blank in the string coincides with that column.
   - A comment line cannot follow a continuation line.
5. If you must continue on another line, proceed with rule 1.
6. Definition statements and macroinstructions can be coded on as many lines as needed.
7. Comments can appear on every line of a continued definition statement or macroinstruction.
8. Columns 73–80 can be used to code identification characters, definition statement sequence characters, or both.

Rules for coding comments

You can write comments after the operand field, but they must be separated from the last operand of the operand field by one or more blanks. Comments can be continued to the next line, but they must start in column 17 or beyond. An entire line can be used for a comment by placing an asterisk (*) in the first column of the line. If you need several entire lines for comments, place an asterisk in the first column of each line and leave column 72 blank. A definition statement or macroinstruction that has no operands cannot have comments on the same line as the operation code.

Using a default

If an operand value has a default, letting the operand's value default or coding the operand's default has the same effect. For example, in the following definition statement for an application program with the name APPL01 the MAXPVT operand was not coded. VTAM will assign the MAXPVT value at 0 because that is the default.
VTAM will also use the default if you code a null value for the operand. In the following example, the operand was coded without a value.

```
APPL01  APPL  APPC=YES,  X
         CMPAPPL=4,  X
         CMPAPLO=1,  X
         AUTH=(ACQ,PASS)
```

**Using the sift-down effect in VTAM definition statements**

The sift-down effect allows you to code an operand on a higher-level node (for example, PU is a higher-level node than LU) so that you do not need to recode it on each lower-level node for which you want the same value. Many of the operands whose values sift to lower-level nodes do not affect the higher-level nodes on which they are coded but can be coded there to take advantage of sifting.

You can override the sifted values at any point in the hierarchy of definition statements. To override the sifted values for a specific lower-level node, use either of the following methods:

- Code the same operand with a different value on the lower-level node. The recoded value applies only to the specific node on which it is coded and all lower-level nodes. All other nodes at the same level will use the sifted value. For example, if you code a sifted value of 2 on an operand of a PU definition statement that has multiple subordinate LU definition statements and then code a value of 4 for the same operand on the third LU statement, the third LU will use the value of 4 and all other LUs subordinate to the PU will use the sifted value of 2.

- Code the same operand with no value, followed by a comma. This causes the default value to replace the sifted value. For example, if you want to automatically logon all but one LU under a PU to application APPL1, you could code LOGAPPL=APPL1 on the PU statement and LOGAPPL=, on the one LU you do not want automatically logged on. The sifted value from the PU definition statement would apply to all LU definition statements except the one on which LOGAPPL=, is coded, whether the other LU definition statements precede or follow it.

The operands to which sifting applies are identified in the definition statement and operand table in each topic, in the column labeled “Sift Effect”. For information about definition statement sequencing and the sift-down level for each NCP operand, see *NCP, SSP, and EP Resource Definition Reference*.

**Note:** For resources added through dynamic reconfiguration, sifting takes place within the hierarchy of minor nodes being added. Operand values coded in the original hierarchy do not sift down to those resources dynamically defined.

**Restrictions on names**

Specify unique names for all VTAM-related definitions that are active concurrently. For example, the name on a definition statement defining a minor node must not be the major node name. A definition is active when VTAM reads the major node in which it is contained (see *Planning for NetView, NCP, and VTAM* for NCP backup exceptions).

VTAM also creates internal resource definitions based on certain parameters in active VTAM definitions. For example, VTAM creates a direct attachment node...
(DAN) definition for each channel-attached non-SNA terminal in the VTAM domain when the device is activated. The name of the DAN is the character string coded on the CUADDR operand on the LOCAL macro that defines the terminal in a local non-SNA major node definition. Therefore, the name on a definition statement must not be the same as the CUADDR value specified on the LOCAL macro in a local non-SNA major node.

You should not use any name beginning with “IST”, “IUT”, or any of the following names for a major or minor node name:

- VTAM
- VTAMSEG
- VTAMSG2
- VTAMTERM

**Note:** VTAM may or may not police these names. Therefore, unpredictable results may occur. If, for example, the user codes IUTX0D00 as an application name, it will be accepted. If an attempt is made later to dynamically create TRLE IUTX0D00 for a TCP/IP CTC connection, the new definition will fail.

The names of major node members need to follow the rules for names detailed in “Format of definition statements” on page 8 and “Format of macroinstructions” on page 10, except that they cannot contain the MVS system symbol delimiters, ampersand (&) or a period (.)

### Using MVS system symbols in VTAM definition statements and macroinstructions

You can use MVS system symbols in the names you specify for VTAM definition statements and in the values you specify on the operands on those definition statements in all major nodes and in all definitions for routing and dynamic reconfiguration.

You can also use MVS system symbols in the names you specify for VTAM macroinstructions and in the values you specify on the operands on those macroinstructions for the following user-definable tables:

- Associated LU table
- APPN Class of Service (CoS) definitions
- APPN-to-subarea Class of Service mapping table
- Message-flooding prevention table
- Model name table

You can use the following symbols:

- **&SYSCONLINE**
  Represents a 1–2-character shorthand notation for the name of the system
- **&SYSNAME**
  Represents a 1–8-character system name
- **&SYSPLEX**
  Represents a 1–8-character sysplex name
- **installation-defined system symbol**
  Represents a 1–8-character identifier. You can define this symbol to be any identifier you want.
The substitution text for these system symbols is defined in MVS and is substituted by MVS in place of the symbols during VTAM startup, major node activation, resource activation, and table activation. See z/OS MVS Initialization and Tuning Guide and z/OS MVS Initialization and Tuning Reference for information about how to define the symbols in MVS.

MVS system symbols can be used to allow a single major node, routing or dynamic reconfiguration definition, or user-definable table to be used by multiple VTAMs in a multisystem environment. For information about how to use MVS system symbols in VTAMLST, see z/OS Communications Server: SNA Network Implementation Guide.

Restrictions on use of assembler features

The NCP and VTAM definition statements are coded in standard operating system macroinstruction format, as described in “Coding VTAM definitions” on page 8, with the following restrictions:

• You cannot use assembler program control instructions (such as ICTL and ISEQ) in groups of definition statements for major nodes.
• You can use assembler listing control statements (such as PRINT, SPACE, and EJECT) in the NCP generation definition statements but not in those for the other types of major nodes.

The following assembler features cannot be used in a group statement for a major node:

• Assembler macroinstructions that generate NCP definition statements
• Names generated by global variables (such as &SYSNDX and &SYSECT)
• Variable substitution at assembly time
• References to assembler attributes (such as length and type)
• Literals
• Quoted strings
• Null keywords; for example, two commas (,,) is treated as a null keyword and is not valid

You can use comments in groups of definition statements for all types of major nodes and PATH definition statements. However, do not use macroinstructions or operands, such as GENEND, especially in a continued line of comments.

Errors made in the groups of major node definition statements filed in the VTAM definition library result in messages to the system operator’s console during VTAM initialization or VARY ACT command processing.

Filing definition statements

Code the appropriate VTAM or NCP definition statements to define the major nodes. You can also file the definitions in the VTAM definition library or to update your definitions after filing them in the VTAM definition library.

Procedure

To define major nodes and file the definitions in the VTAM definition library or to update your definitions after filing them in the VTAM definition library, do the following steps:
1. Code the appropriate VTAM or NCP definition statements to define the major node. If you are defining an NCP, code the NCP-only definition statements and operands as described in the NCP resource definition documents.
   You must use domain-unique names in all VTAM-related definitions that are active at the same time. For example, the name on a definition statement defining a minor node must not be the major node name. A definition is active when VTAM reads the major node in which it is contained. VTAM reads a major node when it is referenced in the configuration list or when it is activated by a VARY ACT command.

2. Generate the NCP as described in the NCP, SSP, and EP Generation and Loading Guide. If you are updating VTAM-only definition statements or operands, update the NCP definitions in the VTAM file; no NCP generation is required. If you are updating NCP-only definition statements or operands, you can use dynamic reconfiguration to update the statements without regenerating the NCP. See “Dynamic reconfiguration and dynamic change” on page 694 for more information.

3. Store the network definition in the proper VTAM library. You can link SYS1.VTAMLST files, but one of the following conditions must be met:
   - The block size must be the same for all files.
   - The file with the largest block size must be the first file in the link.
   - DCB=BLKSIZE=[largest blocksize of concatenated files] must be coded on the first file in the link.
   You must run an operating system utility program to put the definitions into the definition library.

---

**Filing macroinstructions**

Use VTAM macroinstructions to create, replace, or modify VTAM’s interpret, logon mode, Class of Service, or USS tables. You can also use macroinstructions to install the associated LU, model name, or message-flooding prevention tables.

**Procedure**

To create, replace, or modify VTAM’s interpret, logon mode, Class of Service, or USS tables, follow these steps:

1. Code the appropriate VTAM macroinstructions.
2. Assemble the group of macroinstructions.
3. Link-edit the table as a nonexecutable module into a library associated with the VTAMLIB DD statement.

**Installing tables**

**Procedure**

No assembly or link-edit is required to install the associated LU, model name, or message-flooding prevention tables. The groups of macroinstructions are stored in source-language form. To install these tables, follow these steps:

1. Code the appropriate VTAM macroinstructions.
2. Store the group of macroinstructions as a member in SYS1.VTAMLST. The member name is used as the name of the table.
Chapter 2. Major nodes

This topic contains descriptions of the definition statements, operands, and syntax to define major node resources to the VTAM program.

The major nodes are:
- "Adjacent control point major node" on page 32
- "Application program major node" on page 32
- "Channel-attachment major node" on page 84
- "Cross-domain resource major node" on page 103
- "Cross-Domain resource manager (CDRM) major node" on page 145
- "External communication adapter (XCA) major node" on page 172
- "Local non-SNA major node" on page 274
- "Local SNA major node" on page 288
- "LU group major node" on page 340
- "Model major node" on page 357
- "Network control program major node" on page 422
- "Switched major node" on page 551
- "Transport resource list major node" on page 658

Adjacent control point major node

To define an adjacent control point (ADJCP) major node in an APPN network, code a VBUILD definition statement for the major node and an ADJCP definition statement for each adjacent control point (CP) minor node in the major node. There can be one or more minor nodes in the major node.

Note: If the DYNADJCP start option is defaulted or specified as YES, VTAM dynamically creates ADJCP definition statements in the dynamic adjacent control point major node (ISTADJCP) for each adjacent control point.

If the DYNADJCP start option is defaulted or specified as YES, and you code an ADJCP major node, adjacent control points not specified in the ADJCP major node are still dynamically defined in the ISTADJCP major node.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=ADJCP</td>
<td>31</td>
<td>R</td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>name</td>
<td>ADJCP</td>
<td>ALIASRCH</td>
<td>21</td>
<td></td>
<td>YES</td>
<td>BN=YES; APPN only</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AUTHNETS</td>
<td>22</td>
<td></td>
<td></td>
<td>BN=YES; APPN only</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DYNLU</td>
<td>23</td>
<td></td>
<td>DYNLU start option</td>
<td>APPN only</td>
<td></td>
</tr>
</tbody>
</table>
### Table 1. Adjacent control point major node definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>NATIVE</td>
<td>BN=YES; APPN only</td>
<td></td>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NETID</td>
<td>APPN only</td>
<td></td>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NN</td>
<td>APPN only; VN</td>
<td></td>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RTPONLY</td>
<td>NO</td>
<td></td>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td>BN=YES; APPN only</td>
</tr>
<tr>
<td>VN</td>
<td>NO</td>
<td></td>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td>APPN only; DYNLU=NO; NN</td>
</tr>
</tbody>
</table>

Legend:

- **R** Required operand.

### Adjacent control point major node coding notes

This information contains coding notes for the adjacent control point major node.

**VBUILD definition statement**

Code a VBUILD definition statement for each adjacent control point major node.

**ADJCP definition statement**

Code one ADJCP definition statement for each adjacent control point (CP) in the major node.

**NODETYPE start option**

The NODETYPE start option is required for this major node.

The NODETYPE start option allows APPN function. The combination of NODETYPE and the HOSTSA start options determines the various configurations of subarea node, interchange node, migration data host, network node, or end node.

### Adjacent control point major node full syntax

The full syntax for the adjacent control point major node follows. The syntax for each operand is repeated in "Adjacent control point major node operand descriptions" on page 21.

**VBUILD**
Adjacent control point major node operand descriptions

The following information describes the operands you can code for the adjacent control point major node.

ALIASRCH
Specifies whether the defining node (the node that is activating this ADJCP definition) is allowed to process inbound ALIAS searches from this adjacent control point (ADJCP) when the defining node is acting as an entry border node.

**ALIASRCH=YES**
- Specifies that inbound ALIAS searches from this adjacent control point (ADJCP) are allowed.

**ALIASRCH=NO**
- Specifies that inbound ALIAS searches from this adjacent control point (ADJCP) are halted at the defining node. Code ALIASRCH=NO to configure border nodes so that they prevent unwanted ALIAS searches received from one or more adjacent non-native nodes from entering the native APPN network.

**Tip:** You can dynamically change the ALIASRCH value for one or more adjacent CP definitions without restarting VTAM. If a definition for the adjacent CP to be changed has already been activated, then you must first deactivate the entire ADJCP major node containing that definition. (Deactivating a predefined ADJCP major node does not cause links or CP-CP sessions to stop. Rather, the adjacent CP definitions for nodes that currently have active connections to this node are moved to the dynamic ADJCP major node, ISTADJCP. During this process, the current ALIASRCH value in effect is preserved.) After you modify or create the necessary ADJCP minor node definitions, and you activate the ADJCP major node, VTAM updates these definitions immediately if the NETID operand is specified on the ADJCP minor node. If you do not specify the NETID operand on the ADJCP minor node, then you must stop (at the same time) all the links to this adjacent node and restart them for the updated definitions to be used.

**Rule:** The ALIASRCH operand can be specified only on ADJCP definitions that are activated on border nodes (BN=YES). The ALIASRCH operand is ignored and an error message is issued if the activating node is not a border node.

**AUTHNETS**
AUTHNETS—
- netid
- ( - netid - )

statement: ADJCP
dependencies: BN=YES;APPN only

Specifies the list of destination network identifiers that the adjacent node can search for using this node, when the adjacent node is a non-native network node and the defining node is acting as an entry border node. If an inbound search is received for a destination LU whose network ID is not in this list, the search is rejected.

If you specify the AUTHNETS operand but do not specify a netid value, inbound searches for all real network identifiers are rejected.

If you do not specify the AUTHNETS operand, inbound searches for all real network identifiers are allowed.

Tips:
- Specifying the AUTHNETS operand provides a way to configure border nodes so that they prevent externally originating searches for unauthorized network IDs from entering the APPN network without coding a directory services management exit.
- If you specify the AUTHNETS operand but do not specify a netid value, and you specify ALIASRCH=NO, then all inbound searches from the adjacent CP are rejected.
- You can dynamically change the AUTHNETS value for one or more adjacent CP definitions without restarting VTAM. If a definition for the adjacent CP to be changed has already been activated, then you first must deactivate the entire ADJCP major node containing that definition. (Deactivating a predefined ADJCP major node does not cause links or CP-CP sessions to be stopped. Rather, the adjacent CP definitions for nodes that currently have active connections to this node are moved to the dynamic ADJCP major node, ISTADJCP. During this process, the current AUTHNETS value in effect is preserved.) After you modify or create the necessary ADJCP minor node definitions, and you activate the ADJCP major node, VTAM updates these definitions immediately if the NETID operand is specified on the ADJCP minor node. If you do not specify the NETID operand on the ADJCP minor node, then you must stop all the links (at the same time) to this adjacent node and restart them for the updated definitions to be used.

Rule: The AUTHNETS operand can be specified only on ADJCP definitions that are for non-native network nodes.

DYNLU
Specifies whether cross-domain resources using adjacent link stations attached to this adjacent control point can be defined dynamically. The value coded here overrides the value coded on the DYNLU start option and also overrides the values coded on definition statements for resources attached to this adjacent CP.

**DYNLU=NO**
Specifies that dynamic definition of CDRSCs is not allowed for this adjacent CP. You must predefine resources that use adjacent link stations attached to this adjacent CP. If you do not define the resources, the session request fails.

**DYNLU=YES**
Specifies that dynamic definition of CDRSCs is allowed for this adjacent CP. You are not required to predefine resources that use adjacent link stations attached to this adjacent CP.

For subarea nodes (HOSTSA is specified), DYNLU=YES is meaningful only when the host CDRM is activated and CDRDYN=YES is specified on the CDRM definition statement.

For APPN-only nodes (HOSTSA is not specified), DYNLU=YES is meaningful even though a host CDRM cannot be activated.

**Guideline:** To ensure predictable results, code DYNLU on the ADJCP definition statement. If you do not, the DYNLU value coded on the first activated link is assigned to this adjacent CP and to all other links activated to that adjacent node, regardless of the DYNLU value specified for each link.

If DYNLU is not coded on the first activated link station, the value of the DYNLU start option, which defaults to NO, is assigned to this adjacent CP and to all links activated to this adjacent CP. The DYNLU value taken by an adjacent CP and all links attached to it depends on where you code the DYNLU operand. Table 2 shows the DYNLU coding combinations.

<table>
<thead>
<tr>
<th>DYNLU coded on:</th>
<th>DYNLU value assigned to adjacent CP and attached resources:</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJCP</td>
<td>DYNLU Coded</td>
</tr>
<tr>
<td></td>
<td>No effect</td>
</tr>
</tbody>
</table>
Table 2. DYNLU combinations (continued)

<table>
<thead>
<tr>
<th>ADJCP</th>
<th>DYNU coded on:</th>
<th>LINE</th>
<th>GROUP</th>
<th>Other PUs</th>
<th>Start option</th>
<th>DYNU value assigned to adjacent CP and attached resources:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not Coded</td>
<td>DYNU Not Coded</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>Value coded on first PU</td>
</tr>
<tr>
<td>Not Coded</td>
<td>DYNU Not Coded</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>Value coded on LINE (sifts to PU)</td>
</tr>
<tr>
<td>Not Coded</td>
<td>DYNU Not Coded</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>Value coded on GROUP (sifts to PU)</td>
</tr>
<tr>
<td></td>
<td>DYNU Not Coded</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>Value coded on DYNU start option, or default (DYNU=NO)</td>
</tr>
<tr>
<td></td>
<td>DYNU Not Coded</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>Value coded on DYNU start option</td>
</tr>
<tr>
<td></td>
<td>DYNU Not Coded</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>No effect</td>
<td>Default DYNU start option (DYNU=NO)</td>
</tr>
</tbody>
</table>

**Tip:** The DYNU value is set for an adjacent CP and its attached resources from the CDRSC keyword associated with the CDRM for that node when the first link that is activated to the adjacent node represents a virtual route transmission group (VRTG).

**name**
**On VBUILD:**
Specifies the optional name of this major node.

If a syntax or definition error is detected with this statement during activation of the major node, *name* appears in the message issued to identify the error.

The optional *name* you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the status of the node.

**On ADJCP:**
Specifies the required name of the adjacent CP to which connections are to be established.

**NATIVE**
Specifies whether the adjacent node is expected to be in the native subnetwork. If NATIVE is not specified on the ADJCP definition statement or on the PU definition statement, the two nodes negotiate their subnetwork affiliation during connection establishment.

- If the NETIDs match, the connection defaults to a native connection.
- If the NETIDs are different, the connection defaults to a nonnative connection.

No subnetwork negotiations occur if NATIVE has been specified, in this case the value between the two nodes must match for connection to occur.

If you code NATIVE on both the ADJCP and PU definition statements, the values must match.

For transmission groups (TGs) between nodes, the value specified or negotiated for NATIVE must match. If a TG is already active, subsequent TG activations must not have a specified or negotiated value for NATIVE that conflicts with the value of NATIVE for the active TG. If there is a conflict, the transmission group attempting activation is not allowed to become active. If the activation is for a virtual-route-based transmission group then the SSCP-SSCP session also fails.

**NETID**
The network identifier of the adjacent CP.

If NETID is specified, then the network identifier is validated when a connection is established to the adjacent node. If the NETIDs do not match, the session setup is based on the value of the DYNADJCP start option. If DYNADJCP=YES, the session is established dynamically; if DYNADJCP=NO, the session is not established.

If NETID is not specified, the network identifier of the adjacent node is acknowledged at connection establishment. When an ADJCP is defined without a NETID, any NETID with that ADJCP name is considered to be predefined, so the session is always established dynamically, even if DYNADJCP=NO is specified in the start options.

**Tip:** For ADJCP minor nodes that specify the NETID operand, you can dynamically change the ALIASRCH, AUTHNETS, and RTPONLY operands without stopping and restarting CP-CP sessions to these adjacent CPs. If you do not specify the NETID operand on the ADJCP minor node, then you must stop all the links (at the same time) to this adjacent node and restart them for the updated definitions to be used.

**NN**
NN = NO
Specifies that the adjacent node is expected to be an end node.

NN = YES
Specifies that the adjacent node is expected to be a network node.

Note: In a peripheral subnetwork boundary configuration between a border node and a network node without border node function, the border node appears to the non-border node as an end node. In this case, coding NN=YES on the definition statement that represents the border node, a connection failure will occur. Avoid the failure by not coding the NN operand or by coding NN=NO.

RTPONLY
Specifies whether the defining node (the node that is activating this ADJCP
definition) is allowed to perform the Automatic Network Routing (ANR) function
for RTP connections that would otherwise pass through this node and continue to
or through the adjacent node being defined. RTPONLY=YES means that ANR is
not allowed through this node for any RTPs that are established to, from, or
through the adjacent node being defined. Instead, such RTPs will be required to
have an RTP endpoint on this node.

Tips:
- Coding RTPONLY=YES provides a way to configure border nodes so that they
  maintain awareness of all sessions established over specific subnetwork
  boundaries (that is, to or from specific adjacent nonnative nodes). This is
  accomplished by forcing RTPs established over these subnetwork boundaries to
  terminate on the border node or (in some cases where composite network nodes
  are involved and HPRNCPBF=NO is in effect) by reverting to APPN
  intermediate session routing (ISR) instead of HPR.
- You can dynamically change the RTPONLY value for one or more adjacent CP
definitions without restarting VTAM. If a definition for the adjacent CP to be
  changed has already been activated, then you must first deactivate the entire
  ADJCP major node containing that definition. (Inactivating a predefined ADJCP
  major node does not cause links or CP-CP sessions to stop. Rather, the adjacent
  CP definitions for nodes that currently have active connections to this node are
  moved to the dynamic ADJCP major node, ISTADJCP. During this process, the
  current RTPONLY value in effect is preserved.) After you modify or create the
  necessary ADJCP minor node definitions, and you activate the ADJCP major
  node, VTAM updates these definitions immediately if the NETID operand is
  specified on the ADJCP minor node. If you do not specify the NETID operand
  on the ADJCP minor node, then you must stop all the links (at the same time) to
  this adjacent node and restart them for the updated definitions to be used.

Rules:
- RTPONLY can be specified only on ADJCP definitions that are activated on
  border nodes (BN=YES). The RTPONLY operand is ignored and an error
  message is issued if the activating node is not a border node.
- RTPONLY=YES requires that RTP be specified as the first operand of the HPR
  start option. RTPONLY=YES is ignored and an error message is issued if the
  activating node has not specified HPR=RTP or HPR=(RTP,RTP|ANR|NONE).
- RTPONLY=YES is meaningful only on ADJCP definitions that represent
  nonnative nodes (that is, network nodes or border nodes that are not part of the
  local APPN subnetwork.) As a result, RTPONLY=YES is not allowed with
  NATIVE=YES, NN=NO, or VN=YES. The RTPONLY operand is ignored for any
  ADJCP definition that represents a native network node or end node.

Restrictions:
• Use of RTPONLY=YES at any APPN subnetwork boundary on a session setup path will prevent the use of Global VRNs (GVRNs) for intersubnetwork connectivity, because using GVRNs could result in sessions being established across this subnetwork boundary without this border node maintaining awareness of these sessions.

• Use of RTPONLY=YES can result in an increase in network traffic in the form of additional Route_Setup flows used for RTP establishment. These additional Route_Setup flows will occur only during the establishment of sessions that cross subnetwork boundaries defined with RTPONLY=YES.

• Use of RTPONLY=YES can result in an increase in storage and CPU utilization because of VTAM maintaining awareness of these sessions and performing ISR instead of HPR/ANR for these sessions.

• An RTP that is established over a path with RTPONLY=NO specified will not be allowed to path switch to an alternate path that has RTPONLY=YES specified. Therefore, defining alternate paths to the same destination APPN subnetwork with differing values for RTPONLY might result in path switch processing failing to find a viable alternate path during an outage. This could happen during the initial deployment of this function while some border nodes have RTPONLY=YES enabled and others do not, or as a result of forgetting to enable this function at some of the subnetwork boundaries that provide alternate paths to the same nonnative subnetwork.

**TYPE**

```
   TYPE = ADJCP
```

*statement:* VBUILD
*dependencies:* APPN only

Specifies that the VBUILD statement defines an adjacent control point major node to VTAM.

**VN**
Specifies that a virtual node is to be predefined.

VN=NO
  Specifies that no virtual node is to be predefined.

VN=YES
  Specifies that a virtual node is to be predefined.

**Application program major node**

To define an application program major node, code a VBUILD definition statement for the major node, an optional GROUP definition statement to group application program minor nodes, and an APPL definition statement for each application program minor node in the major node. There might be one or more application programs in each application program major node.

**Registering application programs**

An APPN node by default provides some level of resource registration within the network node or end node, depending on its resource type. However, the level of registration can be specified on local resource definitions with the REGISTER operand.

Because VTAM application programs are likely to generate more LU-LU session traffic than other resources, and therefore more APPN locate searches, application programs should be registered on their network node server and optionally on the central directory server to reduce traffic. However, in certain cases, an application program should not be dynamically defined to the APPN network. Such application programs include:

- Programs without network-unique names, such as those that have been excluded from participating in cross-domain session setup
- Programs that participate only in LU-LU sessions within the local VTAM domain
- Programs that can participate in cross-domain LU-LU sessions but will never be the target of a cross-domain search request.

An example of an application program that should not be defined dynamically is TSO/VTAM. TSO usually runs on a host as several applications programs. One main task handles cross-domain LU-LU session setup, and several subtasks control LU-LU sessions after session setup so that TSO services can be accessed by users. See “Resource Registration in an APPN Network” in the **z/OS Communications Server: SNA Network Implementation Guide** for information about how TSO/VTAM applications are registered.
## Non-LU 6.2 application programs

Table 3. Application program major node (Non-LU 6.2) definition statements and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=APPL</td>
<td>79</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>GROUP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>See the APPL definition statement for operands.</td>
</tr>
<tr>
<td>name</td>
<td>APPL</td>
<td>ACBNAME</td>
<td>40</td>
<td></td>
<td></td>
<td>name value</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>APPC</td>
<td>41</td>
<td>G</td>
<td></td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASLENT</td>
<td>42</td>
<td>G</td>
<td></td>
<td>ASLTAB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASLTAB</td>
<td>43</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASRCVLNM</td>
<td>43</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AUTH</td>
<td>44</td>
<td>G</td>
<td></td>
<td>(NOACQ, NOASDP, NOCNM, NOPASS, NOPO, NOTSO, VPACE)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CERTIFY</td>
<td>47</td>
<td>G</td>
<td></td>
<td>NO ENCR</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CMPAPPLI</td>
<td>47</td>
<td>G</td>
<td></td>
<td>0 ENCR</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CMPAPPLO</td>
<td>48</td>
<td>G</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DLOGMOD</td>
<td>50</td>
<td>G</td>
<td></td>
<td>MODETAB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DSPLYWLD</td>
<td>53</td>
<td>G</td>
<td>YES</td>
<td>AUTH</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>EAS</td>
<td>53</td>
<td>G</td>
<td></td>
<td>509</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ENCR</td>
<td>54</td>
<td>G</td>
<td></td>
<td>OPT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ENCRTYPE</td>
<td>56</td>
<td>G</td>
<td></td>
<td>DES</td>
<td>TSO only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FASTPASS</td>
<td>56</td>
<td>G</td>
<td>YES</td>
<td>AUTH</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HAVAIL</td>
<td>57</td>
<td>G</td>
<td></td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOSTERM</td>
<td>59</td>
<td>G</td>
<td></td>
<td>NORMAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LUAPFX</td>
<td>59</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAC</td>
<td>60</td>
<td>G</td>
<td></td>
<td>NONE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MACLNTH</td>
<td>61</td>
<td>G</td>
<td></td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MACTYPE</td>
<td>62</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXPVT</td>
<td>62</td>
<td>G</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MDLEN</td>
<td>63</td>
<td>G</td>
<td></td>
<td>MDLETAB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MDLTAB</td>
<td>64</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODETAB</td>
<td>64</td>
<td>G</td>
<td></td>
<td>ISTINCLM</td>
<td>model only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODSRCH</td>
<td>65</td>
<td>G</td>
<td></td>
<td>NEVER</td>
<td>model only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARSSESS</td>
<td>67</td>
<td>G</td>
<td></td>
<td>NO APPC=NO</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PERSIST</td>
<td>68</td>
<td>G</td>
<td></td>
<td>SINGLE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>POAQLIM</td>
<td>69</td>
<td>G</td>
<td></td>
<td>5000</td>
<td>AUTH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRTCT</td>
<td>70</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RAMPUP</td>
<td>71</td>
<td>G</td>
<td></td>
<td>NO APPN only</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>REGISTER</td>
<td>71</td>
<td>G</td>
<td></td>
<td>CDSERVR</td>
<td>APPN only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SESSLIM</td>
<td>74</td>
<td>G</td>
<td></td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SIRFMSG</td>
<td>75</td>
<td>G</td>
<td></td>
<td>STARTOPT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SONSCIP</td>
<td>76</td>
<td>G</td>
<td></td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>76</td>
<td>G</td>
<td></td>
<td>NetView</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SRBEXIT</td>
<td>77</td>
<td>G</td>
<td></td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SSCPFM</td>
<td>77</td>
<td>G</td>
<td></td>
<td>USSPOI</td>
<td>AUTH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USSTAB</td>
<td>80</td>
<td>G</td>
<td></td>
<td>ISTINCNO</td>
<td>AUTH</td>
</tr>
</tbody>
</table>
Table 3. Application program major node (Non-LU 6.2) definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>VARYWLD</td>
<td></td>
<td></td>
<td>80</td>
<td>G</td>
<td>YES</td>
<td></td>
<td>AUTH</td>
</tr>
<tr>
<td>VCNS</td>
<td></td>
<td></td>
<td>81</td>
<td>G</td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VPACING</td>
<td></td>
<td></td>
<td>82</td>
<td>G</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VTAMFRR</td>
<td></td>
<td></td>
<td>83</td>
<td>G</td>
<td>NO</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R    Required operand.
G    Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.

LU 6.2 application programs

Table 4. Application program major node (LU 6.2) definition statements and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD TYPE=APPL</td>
<td></td>
<td>79</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>GROUP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>APPL</td>
<td>ACBNAME</td>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R    Required operand.
G    Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.

AUTOSES 46 G 0 APPC=YES
CMPAPPLI 47 G 0
CMPAPPLO 48 G 0
DDRAINL 49 G NALLOW APPC=YES
DLOGMOD 50 G MODETAB
DMINWNL 50 G 1 APPC=YES DMINWNR
DMINWNR 51 G 1 APPC=YES; DMINWNL; DSESLIM
DRESPL 51 G NALLOW APPC=YES
Table 4. Application program major node (LU 6.2) definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSESLIM</td>
<td></td>
<td>G</td>
<td>52</td>
<td></td>
<td></td>
<td>2</td>
<td>APPC=YES; DMINWNL; DMINWNR AUTH</td>
</tr>
<tr>
<td>DСПLYWLD</td>
<td></td>
<td>G</td>
<td>53</td>
<td></td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EAS</td>
<td></td>
<td>G</td>
<td>53</td>
<td></td>
<td>509</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENCR</td>
<td></td>
<td>G</td>
<td>54</td>
<td></td>
<td>OPT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENCRTYPE</td>
<td></td>
<td>G</td>
<td>56</td>
<td></td>
<td>DES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FASTPASS</td>
<td></td>
<td>G</td>
<td>56</td>
<td></td>
<td>YES</td>
<td></td>
<td>TSO only</td>
</tr>
<tr>
<td>HAVAIL</td>
<td></td>
<td>G</td>
<td>57</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LIMQSINT</td>
<td></td>
<td>G</td>
<td>58</td>
<td></td>
<td></td>
<td></td>
<td>APPC=YES</td>
</tr>
<tr>
<td>LMDENT</td>
<td></td>
<td>G</td>
<td>58</td>
<td></td>
<td>19</td>
<td></td>
<td>APPC=YES</td>
</tr>
<tr>
<td>LOSTERM</td>
<td></td>
<td>G</td>
<td>59</td>
<td></td>
<td>NORMAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXPVT</td>
<td></td>
<td>G</td>
<td>62</td>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAC</td>
<td></td>
<td>G</td>
<td>60</td>
<td></td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MACLNGTH</td>
<td></td>
<td>G</td>
<td>61</td>
<td></td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MACTYPE</td>
<td></td>
<td>G</td>
<td>62</td>
<td></td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDLENT</td>
<td></td>
<td>G</td>
<td>63</td>
<td></td>
<td></td>
<td></td>
<td>MDLTAB</td>
</tr>
<tr>
<td>MDLTAB</td>
<td></td>
<td>G</td>
<td>64</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODETAB</td>
<td></td>
<td>G</td>
<td>64</td>
<td></td>
<td>ISTINCLM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODSRCH</td>
<td></td>
<td>G</td>
<td>65</td>
<td></td>
<td>NEVER</td>
<td></td>
<td>model only</td>
</tr>
<tr>
<td>OPERCNOS</td>
<td></td>
<td>G</td>
<td>67</td>
<td></td>
<td>NALLOW</td>
<td></td>
<td>APPC=YES</td>
</tr>
<tr>
<td>PARSSESS</td>
<td></td>
<td>G</td>
<td>67</td>
<td></td>
<td>YES</td>
<td></td>
<td>APPC=YES</td>
</tr>
<tr>
<td>PERSIST</td>
<td></td>
<td>G</td>
<td>68</td>
<td></td>
<td>SINGLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>POAQLIM</td>
<td></td>
<td>G</td>
<td>69</td>
<td></td>
<td>5000</td>
<td></td>
<td>AUTH</td>
</tr>
<tr>
<td>PRTCT</td>
<td></td>
<td>G</td>
<td>70</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGISTER</td>
<td></td>
<td>G</td>
<td>71</td>
<td></td>
<td>CDSERVr</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>SECACPT</td>
<td></td>
<td>G</td>
<td>72</td>
<td></td>
<td>NONE</td>
<td></td>
<td>APPC=YES</td>
</tr>
<tr>
<td>SECLVL</td>
<td></td>
<td>G</td>
<td>73</td>
<td></td>
<td>ADAPT</td>
<td></td>
<td>APPC=YES</td>
</tr>
<tr>
<td>SESSLIM</td>
<td></td>
<td>G</td>
<td>74</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SIRFMSG</td>
<td></td>
<td>G</td>
<td>75</td>
<td></td>
<td>STARTOPT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SONSCIP</td>
<td></td>
<td>G</td>
<td>76</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPAN</td>
<td></td>
<td>G</td>
<td>76</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>SRBEXIT</td>
<td></td>
<td>G</td>
<td>77</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSCPFM</td>
<td></td>
<td>G</td>
<td>77</td>
<td></td>
<td>USSPOI</td>
<td></td>
<td>AUTH</td>
</tr>
<tr>
<td>SYNCVL</td>
<td></td>
<td>G</td>
<td>78</td>
<td></td>
<td>CONFIRM</td>
<td></td>
<td>APPC=YES</td>
</tr>
<tr>
<td>USSTAB</td>
<td></td>
<td>G</td>
<td>80</td>
<td></td>
<td>ISTINCNO</td>
<td></td>
<td>AUTH</td>
</tr>
<tr>
<td>VARYWLD</td>
<td></td>
<td>G</td>
<td>80</td>
<td></td>
<td>YES</td>
<td></td>
<td>AUTH</td>
</tr>
<tr>
<td>VCNS</td>
<td></td>
<td>G</td>
<td>81</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VERIFY</td>
<td></td>
<td>G</td>
<td>81</td>
<td></td>
<td>NONE</td>
<td></td>
<td>APPC=YES</td>
</tr>
<tr>
<td>VPACING</td>
<td></td>
<td>G</td>
<td>82</td>
<td></td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VTAMFRR</td>
<td></td>
<td>G</td>
<td>83</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R  Required operand.

G  Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.
Application program major node coding notes

This information contains coding notes for the application program major node.

**VBUILD definition statement**
Code a VBUILD definition statement for each application program major node.

**GROUP definition statement**
Code a GROUP definition statement to group application program minor nodes. All operands that are valid on the APPL definition statement can be coded instead on the GROUP statement for sifting, except for ACBNAME, which must be coded on the APPL definition statement.

Note: If APPC is coded on the GROUP definition statement, its value cannot be overridden by the APPL definition statement.

**APPL definition statement**
Code one APPL definition statement for each application program that is in the VTAM domain and include it in an application program major node.

If you use TSO/VTAM, the MVS logon manager, or the Tivoli NetView for z/OS (NetView) alias translation facility for alias name translation, there are specific coding requirements for the APPL definition statement. See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/SSLTBW_2.2.0/com.ibm.zos.v2r1.scdsos2r1.doc/info/sg247786.html) for specific information on these requirements.

**Dynamic change of operands for application program major node**
You can dynamically change all application program major node operands by editing the VTAMLST member and then issuing the VARY ACT,UPDATE=ALL command.

To dynamically change an operand on the APPL statement, the application must be inactive. You can dynamically change the operands associated with tables (ASLTAB, MDLTAB, and MODETAB) at any time without deactivating resources.

Restriction: Dynamic change of operands does not apply to clone applications.

**Application program major node full syntax**
The full syntax for the application program major node follows. The syntax for each operand is repeated in "Application program major node operand descriptions" on page 40.

**VBUILD**
Chapter 2. Major nodes
Application program major node operand descriptions

The following information describes the operands you can code for the application program major node.

ACBNAME
ACBNAME = value_of_name_operand
ACBNAME = acb_name

Specifies the minor node name assigned to this application program. This name must be unique within the domain in which the application program resides.

This operand is needed if application programs with the same APPLID are started in multiple hosts and the application programs are to set up a cross-domain session. If this name is not identical to the network name, it cannot be network qualified and used in any VTAM command or macroinstruction. The ACBNAME can be used by an LU (including an application program LU) only in a command issued in the same domain where the ACBNAME is defined.

If you do not code this operand, the network-unique name (the name of the APPL definition statement) is used as the ACBNAME.

For application programs used in extended recovery facility (XRF) sessions, the network name and the ACBNAME must be the same. Therefore, do not code the ACBNAME operand; use the default, which is the network name.

acb_name must follow the rules for naming application programs as described in the z/OS Communications Server: SNA Network Implementation Guide and can include the following wildcard characters:

**Asterisk (*)**
Represents 0 or more unspecified characters. An asterisk can be used as the second to eighth character.

**Question mark (?)**
Represents a single unspecified character. A question mark can be used as the first to eighth character.

For information about how to use wildcard characters to create model application program definitions, see z/OS Communications Server: SNA Network Implementation Guide.

APPC
Specifies whether the application program can issue APPCCMD macroinstructions. See the z/OS Communications Server: SNA Programmer's LU 6.2 Reference for a description of this macroinstruction.

**Note:** APPC, if specified as NO (or defaulted), on the GROUP definition statement, can be overridden on the APPL definition statement with APPC=YES. However, if APPC=YES on the GROUP definition statement, APPC=NO is ignored on the APPL definition statement and all application programs in the group will be APPC-capable.

**APPC=NO**
- Specifies that the application program cannot issue APPCCMD macroinstructions.

**APPC=YES**
- Specifies that the application program can issue APPCCMD macroinstructions.

**ASLENT**
ASLENT

associates_lu_table_entry_name

statements: GROUP, APPL
dependencies: ASLTAB

Specifies the name of the associated LU table entry to be used for this application program.

If you code the ASLENT operand, you must also code the ASLTAB operand. If you omit ASLENT and include ASLTAB, VTAM uses the first entry in the associated LU table by default.

ASLTAB

associates_lu_table_name

statements: GROUP, APPL
dependencies: none

Specifies the name of the associated LU table to be used for this application program. See "Associated LU table" on page 981 for a description of how to define an associated LU table.

If you omit the ASLTAB operand, VTAM will not provide the names of associated logical units to the application during LU-LU session initiation. However, you can provide the associated LU names in other ways. You can enter them from the terminal on the PRINTER1 and PRINTER2 operands of the LOGON command, or VTAM can send them to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the PRINTER1 and PRINTER2 operands of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace an associated LU table (ASLTAB). See z/OS Communications Server: SNA Operation for more information about this command.
ASRCVLM = amt_of_address_space

Specifies an amount of address space storage used by VTAM to calculate constrain and critical threshold values for the address space. High usage periods of address space storage should be considered when selecting this value. If you specify 0, the amount of data space storage used is 1,000,000 bytes.

Note: ASRCVLM is applicable only to the application receiving the data and not to the sender. VTAM will queue inbound data to the application's dataspace if the application does not have a RECEIVE RPL available to field the data. The ASRCVLM parameter is used by the receiving VTAM to adjust the window size in the pacing responses returned to the data sender. Eventually the ASRCVLM limits may be hit and VTAM will respond by lowering the session level.

ATNLOSS

ATNLOSS=ALL
- Specifies that the ATTN exit is to be scheduled for all session deactivations.

ATNLOSS=LAST
- Specifies that the ATTN exit is to be scheduled when the last session between an (LU-mode) pair is deactivated.

Code this operand for any LU 6.2 application.

AUTH
Determines whether this application program has the authority to use certain VTAM functions. You can code these operands in any order. The possible operands are listed below:

**AUTH=ACQ**
**AUTH=NOACQ**

Determines whether this application program can use either the OPNDST macroinstruction with the ACQUIRE option or the SIMLOGON macroinstruction. (These macroinstructions enable the application program to initiate a session with a particular logical unit.)

If you code APPC=YES, the default is ACQ. If you code APPC=YES and AUTH=NOACQ, VTAM supplies an override of ACQ and issues a warning message.

**AUTH=ASDP**
**AUTH=NOASDP**

Specifies whether an application can supply dial digits (telephone number) and other parameters for a dial-out switched connection during session initiation.

**AUTH=CNM**
**AUTH=NOCNM**

Determines whether this application program can use the communication network management (CNM) interface. This interface allows an application program to send and receive formatted request units from certain physical units in the application program’s domain. For more information on the CNM interface, see [z/OS Communications Server: SNA Programming](https://www.ibm.com/support/knowledgecenter/SANS58_7.2.0/com.ibm.zos.v2r1.cecsnperfil.htm).

You must code AUTH=CNM for all application programs for which entries exist in the IBM-supplied CNM routing table or for any user-written applications using the CNM interface. For a complete list of IBM-supplied CNM applications, see [z/OS Communications Server: SNA Customization](https://www.ibm.com/support/knowledgecenter/SANS58_7.2.0/com.ibm.zos.v2r1.cecsnconfig.htm).

**AUTH=PASS**
AUTH=NOPASS
Determines whether this application program can use the CLSDST macroinstruction with the PASS option to pass session-establishment requests to other application programs.

For more information on the CLSDST macroinstruction, see z/OS Communications Server: SNA Programming.

AUTH=PPO
AUTH=SPO
AUTH=NOPO
Specifies whether the application program can issue VTAM operator commands and receive responses and unsolicited messages. z/OS Communications Server: SNA Programming describes the program operator application.

If you code either the primary program operator (PPO) or the secondary program operator (SPO), the application program is authorized to issue SENDCMD and RCVCMD macroinstructions.

If you code PPO, the application program receives all unsolicited messages, that is, all informational and error messages that are not replies to operator commands. If no PPO-designated application program is active when these messages occur, they are directed to the system console.

AUTH=TSO
AUTH=NOTSO
Specifies whether the application program is a TSO/VTAM time-sharing program. If it is a TSO/VTAM time-sharing program, code TSO. If not, code NOTSO or take the default.

If you code AUTH=TSO, you cannot use wildcard values in the name field of this APPL definition statement.

AUTH=VPACE
AUTH=NVPACE
Determines whether this application program is subject to the VPACING specifications of SLUs with which the program is in session. Coding NVPACE is the same as coding VPACING=0 in the LU definition statements for all of the SLUs with which the application program is in session. NVPACE is ignored for same-domain local SNA logical units.

Note: It is recommended that you code AUTH=NVPACE for normal interactive TSO/VTAM sessions. However, if you have application-to-application TSO/VTAM sessions, it is recommended that you pace them by coding VPACING=20.

For more information on pacing, see the discussion in z/OS Communications Server: SNA Network Implementation Guide.

AUTOSES
**AUTOSES**

- **Statements:** GROUP, APPL
- **Dependencies:** APPC=YES
- **Range:** 0–32767

Tells VTAM the number of contention-winner sessions to start automatically, rather than by allocation requests from the application program. VTAM establishes the specified number of sessions when the first change number of sessions (CNOS) request is issued for a mode-name group of a given partner LU.

See z/OS Communications Server: SNA Programmer’s LU 6.2 Guide for more information about the role of the AUTOSES operand in session activation.

**CERTIFY**

- **Statements:** GROUP, APPL
- **Dependencies:** CERTIFY is ignored unless ENCR is also active. Before enabling the CERTIFY function, ensure that the device or software providing the partner's LU support also supports the CERTIFY function.

Specifies whether VTAM on behalf of this application program will verify authentication.

**CERTIFY=YES**

- Specifies SLU authentication (verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

**CERTIFY=NO**

- Specifies no SLU authentication (not verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.
Specifies the maximum compression levels for the application's input data, which is the data the primary logical unit (PLU) receives. The operand allows compression-level limits for a specific application to be set lower than the overall limit specified by the CMPVTAM start option.

Values of `input_data_compression_level` follow:

- 0    No compression
- 1    Run-length encoding (RLE)
- 2    Small adaptive compression table
- 3    Medium adaptive compression table
- 4    Large adaptive compression table

The limit specified by the operand is enforced only during sessions for which:
- The application acts as the PLU.
- The limit is lower than that specified by the CMPVTAM start option.

See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) for more information about compression levels.
Specifies the maximum compression levels for the application's output data, which is the data the primary logical unit (PLU) sends. The operand allows compression-level limits for a specific application to be set lower than the overall limit specified by the CMPVTAM start option.

Values of output_data_compression_level follow:

0  No compression
1  Run-length encoding (RLE)
2  Small adaptive compression table
3  Medium adaptive compression table
4  Large adaptive compression table

The limit specified by the operand is enforced only during sessions for which the application acts as the PLU, and the limit is lower than that specified by the CMPVTAM start option.

See the [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/docview.wss?uid=swg21414371) for more information about compression levels.
Specifies whether the application program can drain waiting allocation requests when the application is the target of a CNOS request that resets the session limit to zero.

**DDRAINL=ALLOW**

Allows the application program to drain its allocation requests. The drain-target value in the received CNOS request is not negotiated.

**DDRAINL=NALLOW**

Does not allow the application program to drain its allocation requests. The drain-target value in the received CNOS request is negotiated to no draining.

**DLOGMOD**

Specifies the name of the logon mode table entry used by default if one is not otherwise provided. If you do not code this operand and the name of a logon mode table entry is not otherwise provided, VTAM uses the first entry in the applicable logon mode table (specified on the MODETAB operand or used by default).

If you specify MODETAB, the entry must be in either the specified table or in ISTINCLM, an IBM-supplied logon mode table. If you do not specify MODETAB, the entry must be in ISTINCLM.

A logon mode entry determines which entry in the applicable logon mode table is to provide a set of session parameters for the application program if the application program is a secondary logical unit (SLU). The name specified on the DLOGMOD operand must be the name of an entry in a logon mode table.

For more information, see "Logon mode table" on page 1026.

**DMINWNL**
DMINWNL

- number_of_sessions

Statements: GROUP, APPL
Dependencies: APPC=YES, DMINWNL, DSESLIM
Range: 0–32767

Defines the minimum number of parallel sessions for a mode name group for a given remote logical unit of which this application program (the local logical unit) is guaranteed to be the contention winner and the remote logical unit is guaranteed to be the contention loser. A mode name group is a group of sessions for a given mode name.

The DMINWNL value must always be less than or equal to the DSESLIM value minus the DMINWNR value.

See z/OS Communications Server: SNA Network Implementation Guide and z/OS Communications Server: SNA Programmer’s LU 6.2 Reference for information on how VTAM negotiates the actual number of contention-winner sessions for this application (local logical unit).

DMINWNR

- number_of_sessions

Statements: GROUP, APPL
Dependencies: APPC=YES, DMINWNL, DSESLIM
Range: 0–32767

Defines the minimum number of parallel sessions for a mode name group for a given remote logical unit of which the remote logical unit is guaranteed to be the contention winner and this application program (local logical unit) is guaranteed to be the contention loser.

The DMINWNR value must always be less than or equal to the DSESLIM value minus the DMINWNL value.

See z/OS Communications Server: SNA Network Implementation Guide and z/OS Communications Server: SNA Programmer’s LU 6.2 Reference for information on how VTAM negotiates the actual number of contention-winner sessions for the remote logical unit.

DRESPL
Specifies whether the local application program (local logical unit) will deactivate sessions on receipt of a CNOS request that specifies the application program as the responsible logical unit.

See z/OS Communications Server: SNA Programmer’s LU 6.2 Guide for an explanation of draining.

**DRESPL=ALLOW**
- Allows the application program to deactivate sessions. The value indicating the responsible LU in the reserved CNOS is not negotiated.

**DRESPL=NALLOW**
- Does not allow the application program to deactivate sessions. Instead, VTAM assigns the responsibility for deactivating sessions to the remote logical unit that sent the CNOS request.
**DSESLIM**

```
DSESLIM = number_of_sessions
```

**Statements:** `GROUP, APPL`

**Dependencies:** `APPC=YES, DMINWNL, DMINWNR`

**Range:** 0–32767

Defines the maximum number of sessions to be allowed between the local application logical unit and a remote logical unit on a given mode name group.

The DSESLIM value is always greater than or equal to the combined values for DMINWNL and DMINWNR.

If you code DSESLIM=0 and the application program is the target of a CNOS request to start the session limit for a new mode, VTAM responds to the CNOS request with an indication that the requested mode is not available.

See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) and the [z/OS Communications Server: SNA Programmer’s LU 6.2 Reference](https://www.ibm.com) for information on how VTAM determines the negotiated session limit.

**DSPLYWLD**

```
DSPLYWLD = YES
```

**Statements:** `GROUP, APPL`

**Dependencies:** `AUTH=PPO or AUTH=SPO`

Indicates whether the program operator application can issue DISPLAY commands containing wildcards. See [z/OS Communications Server: SNA Operation](https://www.ibm.com) to determine which display commands are governed by this operand.

**DSPLYWLD=NO**

Specifies that wildcards are not permitted from this program operator application.

**DSPLYWLD=YES**

Specifies that wildcards are permitted from this program operator application when the DSPLYWLD start option is FULLWILD or POAONLY.

**EAS**
Specifies the estimated number of concurrent sessions this application program will have with other logical units (LU-LU sessions). VTAM uses this operand in a search to find the representation of a session between the application program and a logical unit.

The specified value should include LU 6.2 sessions and non-LU 6.2 sessions.

To avoid obtaining unnecessary common service area (CSA) storage for TSO and Telnet users, you should code EAS=1 when defining TSO and Telnet applications to VTAM. Do not use the default.

**Note:** You should code EAS on applications that have only a few concurrent sessions, to conserve storage.

Accurate coding of the EAS value for your applications can save storage in your system. For example, if you estimate that there will be less than 30 sessions with this application, but you let the EAS value default to 509, then an extra 4K table will be allocated from common storage. The size of the table is based on the EAS value that you code and is determined as follows:

<table>
<thead>
<tr>
<th>Table size</th>
<th>EAS value</th>
</tr>
</thead>
<tbody>
<tr>
<td>4K</td>
<td>30–4000</td>
</tr>
<tr>
<td>8K</td>
<td>4001–8000</td>
</tr>
<tr>
<td>16K</td>
<td>8001–16000</td>
</tr>
<tr>
<td>32K</td>
<td>16001–32000</td>
</tr>
<tr>
<td>64K</td>
<td>32001–48000</td>
</tr>
<tr>
<td>128K</td>
<td>48001–56000</td>
</tr>
<tr>
<td>256K</td>
<td>56001–64000</td>
</tr>
<tr>
<td>512K</td>
<td>greater than 64000</td>
</tr>
</tbody>
</table>

If your EAS value is specified as lower than the number of sessions that you actually have, sessions would still be established as usual. However, the efficiency of searching for the session representation could be impaired if a smaller table was allocated because of the lower EAS value.

**ENCR**
Specifies whether this application program has any special requirements for
enciphering and deciphering messages.

**ENCRLCOND**

Specifies that this application program will establish an encrypted session if
the session partner supports encryption and cryptographic product support is
available. If support is not available at session start, a clear session (without
encryption) is established.

VTAM starts sessions when all the following statements are true:

- ENCR=COND is coded on the APPL definition of the PLU
- The SLU has an APPL definition with ENCR=COND or an LU definition
  with ENCR=OPT
- A cryptographic support is unavailable

There is no attempt to start data encryption in the middle of a clear session
when the encryption product becomes available again.

In VTAM, the COND option includes a test to determine if cryptographic
product support is available on the primary logical unit (PLU). However, if
either end of a session specifies a higher level of encryption than
ENCRLCOND (ENCRLSEL or ENCR=REQD), an encrypted session is required
and no test is made to determine if cryptographic product support is available.

**ENCRLNONE**

Specifies that this application program cannot have cryptographic sessions.

**ENCRLOPT**

Specifies that the application program has no special cryptographic
requirements; its cryptographic capability is the same as the host processor’s
capability.

**ENCRLREQD**

Specifies that VTAM must encipher all messages to and from the application
program. If ENCR=REQD, this application program cannot be started unless
the host with which it is associated can handle cryptographic sessions.
ENCRLREQD also means that this application program cannot start a session
with a logical unit that cannot encipher and decipher its messages.

**ENCRLSEL**

Specifies that this application program can choose which messages are
enciphered by VTAM. ENCR=SEL also means that this application program
cannot start a session with a logical unit that cannot encipher and decipher its
messages. If you code ENCR=SEL, this application program cannot be started
unless the host in which it resides can handle cryptographic sessions.
ENCRTYPE

- ENCRTYPE=DES
- ENCRTYPE=TDES24

**statements:** GROUP, APPL  
**dependencies:** none

Specifies the minimum type of encryption that VTAM should use on behalf of the application when performing session level encryption.

**ENCRTYPE=DES**

Specifies that VTAM must use a minimum of DES encryption with an 8-byte key when performing session level encryption. This is the default.

**ENCRTYPE=TDES24**

Specifies that VTAM must use a minimum of Triple_DES encryption with a 24-byte key when performing session level encryption.

**Note:** The ENCRTYPE value coded on the GROUP will be sifted down to the appropriate resource (APPL).

The SYSDEFed ENCRTYPE of an APPL resource can be changed at runtime through the MODIFY SECURITY command which is described in [z/OS Communications Server: SNA Operation](https://www.ibm.com/support/docview.wss?uid=swg27047292). However, the ENCRTYPE value will not be able to be downgraded. In other words, if the application is currently operating at ENCRTYPE=TDES24, then the encryption level cannot be changed to ENCRTYPE=DES using the MODIFY SECURITY command.

**Note:** If downgrading the security level is preferred, the current major node being used must be deactivated and another major node value must be coded.

In addition, a user may use the LOGMODE TABLE to override the ENCRTYPE value of the SLU based on the logmode entry being used. The ENCRTYPE value will not be able to be downgraded from TDES24 to DES using the logmode table. Only TDES24 can be specified for the encryption level (ENCRTPY). See [“MODEENT macroinstruction” on page 1027](https://www.ibm.com/support/docview.wss?uid=swg21261307) for more information about MODEENT macroinstructions.

**FASTPASS**
FASTPASS=NO

Specifies that an unenhanced logon procedure requiring multiple binds is performed for application programs that issue the CLSDST PASS macroinstruction with the PASS option as part of their session establishment procedure.

FASTPASS=YES

Specifies that the logon performance enhancement is performed for application programs that issue the CLSDST macroinstruction with the PASS option as part of their session establishment procedure.

HAVAIL


**Determine whether the application can support extended recovery facility (XRF) sessions.**

**HAVAIL=NO**

Specifies that this application program cannot have an XRF session.

**HAVAIL=YES**

Specifies that this application program can have an XRF session, can initiate an XRF session, or can be the primary logical unit in an XRF session that is initiated by a secondary logical unit.

**LIMQSINT**

Specifies how often VTAM searches the queue for unused LU 6.2 contention winners.

If LIMQSINT=0, contention-winner LU 6.2 limited resource sessions are terminated immediately after their conversations end.

As a starting point, *timeout_value* should be at least one second less than half of the shortest time cost interval. For example, if the time cost interval for a line is 120 seconds, you should specify 59 seconds. Because applications can work differently, you might need to adjust the time cost interval as necessary.

Sessions that are active for longer than the *timeout_value* seconds are deactivated on the expiration of the timer and the next queue search.

**LMDENT**
LMDENT = number_of_entries

Specifies the number of entries to be used for this application program’s hash table of remote logical units. VTAM uses the hash table to identify and locate remote logical units based on their LU names.

The value you specify should take into account both the largest possible number of remote logical units and the degree of efficiency in accessing the remote logical unit definitions.

LOTERM

LOTERM=NORMAL
LOTERM=IMMED
LOTERM=SECOND

Specifies how VTAM terminates a session with a hung LU. You can force an LU to an inactive state when it becomes hung during a session with an application program that has a LOSTERM exit routine but no NSEXIT exit routine.

LOTERM=IMMED
VTAM sends an UNBIND to break the session on the first call to the LOSTERM exit.

LOTERM=NORMAL
VTAM does not terminate the session.

LOTERM=SECOND
VTAM allows the LOSTERM exit to issue a CLSDST on the first call to the LOSTERM exit. If a second call is made, VTAM sends an UNBIND to break the session.

LUAPFX
LUAPFX=string

Indicates the two characters to be used as the prefix for all dynamically generated LUALIAS names for dynamic cross-network CDRSCs in session with this application. You must take into account the VTAM naming conventions when choosing this prefix. See "Format of definition statements" on page 8 for naming conventions.

By using this prefix you can eliminate the need to predefine a CDRSC with LUALIAS for each cross-network resource which may want to establish a session with an application that does not support network qualified names. See "LUALIAS" on page 131 for a description of the operand and how it is used.

Note: If model definitions are being used to define multinode persistent application programs and if PERSIST=MULTI is specified, use an MVS system symbol as part of the LUAPFX value to reduce name collisions during recovery processing.

MAC
Specifies the level of message authentication support required for this application program.

**MAC=COND**
- Specifies that the application is capable of engaging in message supported sessions, and that MACing is required if the session partner is MAC capable.

**MAC=NONE**
- Specifies that this application program is not capable of engaging in message authentication supported sessions.

**MAC=REQD**
- Specifies that VTAM must use a message authentication code on all messages to and from this application program.

**Note:** A mandatory message authentication support and cryptographic enhancements session is established only under the following conditions:
- The owning host is capable of handling message authentication sessions.
- The session partner is capable of supporting message authentication.

**MACLNTH**
Specifies the minimum length, in bytes, of the message authentication code that is to be generated.

MACLNTH is based on MACTYPE, and can be one of the following values:

- \texttt{MACTYPE=CRC}
  - Specifies that VTAM use an internal cyclic redundancy check (CRC) algorithm to generate a message authentication code.

- \texttt{MACTYPE=DES}
  - Specifies that VTAM use the data encryption standard (DES) services provided by the available Common Cryptographic Architecture (CCA) product to generate a message authentication code for this application.

\textbf{Note:} ENCRTYPE affects the type of encryption used when calculating message authentication code for DES or TDES24.

\texttt{MAXPVT}
Specifies how much VTAM can use of the private storage allocated to the application program's task. In most cases, each task is associated with only one application program. If a task involves more than one application program, the effective MAXPVT value for the task is the sum of the values specified for all the application programs opened under the task.

*number_of_bytes* can be expressed as a single number up to 8 digits, or up to 7 digits with M or K. The limit for MAXPVT is approximately 2048M (2 gigabytes).

For CMIP services application programs, the minimum value is 2 megabytes.

**MAXPVT=0**

Specifies that no limit is defined and thus, when no storage is available in the address space associated with this application program, the session is cleared.

**MAXPVT=number_of_bytes**

**MAXPVT=number_of_bytesK**

The number of one-kilobyte segments that this application program task can use of its private area (address space). *number_of_bytes* is a decimal integer. Its value is rounded up to the next multiple of 4.

For CMIP services application programs, if the number of bytes specified is fewer than 2 megabytes, the value is rounded up to 2 megabytes.

**MAXPVT=number_of_bytesM**

The maximum amount of private area storage, in one-megabyte increments, that the application program task can use. *number_of_bytes* is a decimal integer.

For CMIP services application programs, if the number of bytes specified is less than 2 megabytes, the value is rounded up to 2 megabytes.

**MDLENT**
MDLENT = model_name_table_entry_name

---

**statements:** GROUP, APPL
**dependencies:** MDLTAB

Specifies the name of the model name table entry to be used for this application program.

If you code the MDLENT operand, you must also code the MDLTAB operand. If you omit MDLENT and include MDLTAB, VTAM uses the first entry in the table named in MDLTAB.

**MDLTAB**

---

MDLTAB = model_name_table_name

---

**statements:** GROUP, APPL
**dependencies:** none

Specifies the name of the model name table to be used for this application program. See "Model name table" on page 1054 for a description of how to define a model name table.

If you omit the MDLTAB operand, VTAM will not provide the name of the model name table of the application during LU-LU session initiation. However, you can provide the model name in other ways. You can enter it from the terminal on the MODEL operand of the LOGON command, or VTAM can send it to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information about the MODEL operand of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace a model name table. See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**MODETAB**
Specifies the logon mode table to be used to associate each logon mode name with a set of session parameters for the application program if the application program participates as the secondary end of the session (SLU). The name you code must be the name of a logon mode table created as described in "Logon mode table" on page 1026. If you do not supply a logon mode table for the application program, on the MODETAB operand, an IBM-supplied default logon mode table (ISTINCLM) is used. If you specify a table, both the table you specify and the default table are used.

**MODSRCH**

Specifies when the INQUIRE APPSTAT and INQUIRE STATUS macroinstructions will use this model definition. The VTAM containing the application that issued the INQUIRE first searches its tables for a previously defined application. If one is not found, it searches for a local model that can satisfy the request.

**Note:** If you plan to use model application program definitions with TSO/VTAM, then you must code MODSRCH=FIRST on the TSO application definition statement which represents the subordinate TSO APPLs. For more information about MODSRCH, see z/OS Communications Server: SNA Network Implementation Guide.

**MODSRCH=FIRST**

Specifies that VTAM should use this model if it is an appropriate match, and not search the network for the specified name. For example, because the OPEN ACB macroinstruction can use a model definition, you could specify MODSRCH=FIRST so this model can be found by an application issuing INQUIRE APPSTAT to determine whether an OPEN ACB will follow.

**MODSRCH=LAST**

Specifies that VTAM should use this model definition only if all other hosts in the network report that the application was not found.

**MODSRCH=NEVER**

Specifies that VTAM should never use this model to satisfy an INQUIRE macroinstruction.
Assigns a name to the resource being defined by the definition statement. The name is used as the network name for the resource.

**On VBUILD:**

*name* is optional. If a syntax or definition error is detected with this statement during activation of the major node, *name* appears in the message issued to identify the error.

The optional *name* you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node’s status.

**On GROUP:**

*name* is required.

**On APPL:**

*name* is required. It specifies a minor node name for this application program, as does the ACBNAME operand. This name, when used with the network identifier (NETID as specified on the VTAM START command) is the network-qualified name of the application program, in the form of netid.name. This name is sometimes called the real name to distinguish it from an alias or USERVAR name. This name must be unique in the following situations:

- Within the domain in which the application resides
- Throughout the network if the application program participates in cross-domain sessions
- Throughout all networks if the application program participates in cross-network sessions, unless network-qualified names, LUALIAS, or the alias translation function is used.

*name* must follow the rules for naming application programs as described in the [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) and can include the following wildcard characters:

**Asterisk (*)**

Represents 0 or more unspecified characters. An asterisk can be used as the second to eighth character.

**Question mark (?)**

Represents a single unspecified character. A question mark can be used as the first to eighth character.
A logical unit can request a session with an application by using one of the following names associated with the application:

- An application program network name, the `name` operand (from any domain in the network)
- The ACB name associated with the application (from within the same domain).

A logical unit using the application program’s network name (the `name` operand on the APPL definition statement) can establish a session with the application program from any domain in the network. A logical unit using the ACB name to establish a session with an application program must be in the same domain.

If you do not code `ACBNAME`, this name must be the same as the name in the `APPLID` operand of the ACB macroinstruction for this application program. z/OS Communications Server: SNA Programming describes the ACB macroinstruction.

See z/OS Communications Server: SNA Network Implementation Guide for considerations when coding the name to define terminal control address space (TCAS) or a time sharing option (TSO) user.

Note: If you duplicate names for application program minor nodes, it is not advisable to register more than one of the duplicate names at a time. If there are duplicate application program major node names and more than one is registered, you will receive messages about the duplication.

**OPERCNOS**

<table>
<thead>
<tr>
<th>OPERCNOS</th>
<th>NALLOW</th>
<th>ALLOW</th>
</tr>
</thead>
</table>

**statements:**

GROUP, APPL

**dependencies:**

APPC=YES

Specifies whether the operator can issue the MODIFY CNOS and MODIFY DEFINE commands.

The MODIFY CNOS command changes the CNOS parameters governing LU 6.2 sessions. The MODIFY DEFINE command establishes new entries or modifies existing entries in the LU-mode table for the specified application program.

**OPERCNOS=ALLOW**

Allows the operator to issue the MODIFY CNOS command to change the CNOS parameters governing LU 6.2 sessions.

**OPERCNOS=NALLOW**

Prevents the operator from issuing the MODIFY CNOS command to change the CNOS parameters governing LU 6.2 sessions.

**Note:** OPERCNOS applies only to the modification of CNOS parameters. Therefore, it does not affect the operator’s ability to display CNOS information.

**PARSESS**
Spefies whether this application program can have multiple concurrent (parallel) sessions with another application program or independent logical unit in the same domain or in another domain.

**PARSESS=NO**
- Restricts this application program to one LU-LU session, between itself and another logical unit, at a time. VTAM uses netnames rather than CIDs if PARSESS=NO.

**PARSESS=YES**
- Allows this application program to have multiple LU-LU sessions between the same session partners.


**PERSIST**
Specifies whether the application is capable of multinode persistent sessions.

**PERSIST=SINGLE**
Indicates single-node persistence only is allowed. SINGLE can be abbreviated as S.

**PERSIST=MULTI**
Indicates multinode persistence is allowed in addition to single-node persistence support. MULTI can be abbreviated as M.

**Note:**
1. VTAM will issue error message IST1350I if PERSIST=MULTI is coded and the application program is activated in an environment that does not support multinode persistent sessions. See [z/OS Communications Server: SNA Messages](https://www.ibm.com/support/docview.ws?rs=11404&uid=swg21076996) for details.
2. When coded at a Network Node (or ICN), PERSIST=MULTI is allowed only on a model application definition.
3. If model definitions are being used to define multinode persistent application programs and if LUAPFX is specified on the model definition statement, use an MVS system symbol as part of the LUAPFX value to reduce name collision during recovery processing.

**POAQLIM**
POAQLIM

---

`POAQLIM = number_of_messages`

**statements:** GROUP, APPL  
**dependencies:** AUTH=PPO or AUTH=SPO  
**range:** 1–2147483647

Specifies the number of messages that can be queued for this application program.

POAQLIM applies only to a program operator application program (AUTH=PPO or AUTH=SPO).

Code POAQLIM high enough so the limit will be reached only if there is an apparent error. A limit of five times the number of tasks using the program operator application (POA) times the number of lines in the longest expected message group should be enough to capture all necessary data without using excessive amounts of storage.

The limit for the PPO should be somewhat higher than the SPO to allow for unsolicited messages and PPOLOG messages.

It is also recommended that you increase VTAM private storage by the sum of all the POAQLIM values specified in your system times 152 bytes, if possible. This is done by increasing the REGION parameter on the VTAM start procedure.

**PRTCT**
PRTCT = password

Specifies a 1–8 character password that verifies this application program’s authority to run. VTAM compares the password to the one in the application program’s access method control block (ACB) when an OPEN macroinstruction is issued. If you do not code this operand, no password verification is done.

For TSO applications, you must use the same password for TCAS and for each TSO/VTAM user.

RAMPUP

RAMPUP = NO

Specifies whether a request for a larger pacing window will be honored for sessions using adaptive pacing on the subarea pacing stage. See z/OS Communications Server: SNA Network Implementation Guide for a description of adaptive session level pacing. The RAMPUP operand should be coded on the APPL definition statement associated with the application receiving data.

Note: RAMPUP, if specified as YES, on the GROUP definition statement, can be overridden on the APPL statement with RAMPUP=NO. In addition, RAMPUP, if specified as NO on the GROUP definition statement, can be overridden on the APPL statement with RAMPUP=YES.

RAMPUP=YES

Specifies that application program sessions that use adaptive pacing for the subarea pacing stage will honor a request for a larger pacing window.

RAMPUP=NO

Specifies that application program sessions that use adaptive pacing for the subarea pacing stage will not honor a request for a larger pacing window.

REGISTER
Specifies how a resource should be registered.

**REGISTER=CDSERV**

Indicates that an end node resource is registered to a network node server and directory resource registration is requested for it. A network node resource is registered at the central directory server. If the node is configured as a central directory server, this value has the same effect as **NO**.

**REGISTER=NETSRVR**

Indicates that the end node resource should be registered to the network node server, but that directory resource registration should not be requested for it.

**REGISTER=NO**

Indicates that the resource should not be registered.

**Note:** Because TSO subtasks are not required to have network-unique names, only TSO main tasks should be registered and **REGISTER=NO** should be coded on the definitions of all subtask application programs.

**SECACPT**
Defines whether LU 6.2 conversation requests containing access security subfields are accepted by this application's logical unit.

**SECACPT=ALREADYV**

Specifies that this logical unit supports conversation requests containing access security subfields. The logical unit also accepts the already-verified indications that it receives in conversation requests from partner LUs.

**SECACPT=AVPV**

Specifies that this logical unit supports conversation requests containing access security subfields. The logical unit also accepts the already-verified indications and the persistent verification indications that it receives in conversation requests.

**SECACPT=CONV**

Specifies that this logical unit supports conversation requests containing access security subfields.

**SECACPT=NONE**

Specifies that this logical unit does not support conversation requests containing access security subfields.

**SECACPT=PERSISTV**

Specifies that this logical unit supports conversation requests containing access security subfields. The logical unit also accepts the persistent verification indications that it receives in conversation requests.

**SECLVL**

---

**statements:** GROUP, APPL

**dependencies:** APPC=YES
Specifies whether VTAM uses the basic or enhanced security protocol when performing session-level LU-LU verification. SECLVL specifies the type of verification; the related VERIFY operand specifies whether VTAM performs verification.

This operand is valid only when APPC=YES and session-level LU-LU verification is used. (VERIFY=OPTIONAL or VERIFY=REQUIRED must be coded on this application program major node. If you code VERIFY=None or let it default, a parameter conflict results.)

**SECLVL=ADAPT**
- Specifies that either the enhanced or the basic protocol is used for sessions that use session-level LU-LU verification. VTAM attempts to use the enhanced protocol but accepts the use of the basic protocol if a partner LU does not support the enhanced protocol.

**SECLVL=LEVEL1**
- Specifies that the basic protocol is used for sessions that use session-level LU-LU verification. If a partner LU does not support the basic protocol, VTAM rejects the session and issues a sense code.

**SECLVL=LEVEL2**
- Specifies that VTAM uses only the enhanced protocol for sessions that use session-level LU-LU verification. If a partner LU does not support the enhanced protocol, VTAM rejects the session and issues a sense code.

**SESSLIM**
Determines whether this application program has a session limit between it and another application program, usually an SLU, which handles one session at a time.

**SESSLIM=NO**
- Specifies that this application program does not have a session limit.

**SESSLIM=YES**
- Specifies that this application program has a session limit of one.

### SIRFMSG

Controls the display of error messages IST663I, IST664I, IST889I, and their message groups. These messages are issued when a session initiation request originated by this application program fails because trial and error routing, using an adjacent SSCP table, exhausted the table and was unable to locate the destination LU. The messages might also be issued when a session initiation request targeted to this application program fails for similar reasons.

The SIRFMSG operand setting on the APPL definition statement controls the ASIRFMSG, ESIRFMSG, FSIRFMSG, RSIRFMSG, and SIRFMSG start option processing. See "Descriptions of start options" on page 794 for more details.

When a value other than SIRFMSG=STARTOPT is specified for an application resource, the resource level setting of SIRFMSG overrides whatever SIRFMSG start option setting is in effect at the node. If both resources (origin and destination logical units) for a given session initiation request have SIRFMSG values other than SIRFMSG=STARTOPT specified, then the less restrictive setting (where ALLSSCP is less restrictive than OLUSSCP which in turn is less restrictive than NONE) will be used to determine if the error messages are issued.

**SIRFMSG=ALLSSCP**
- Specifies that messages are issued for all session initiation request failures involving this application program.
SIRFMSG=OLUSSCP
Specifies that messages are issued only when the VTAM owning this application program is the origin logical unit (OLU) SSCP for the failing session initiation request.

SIRFMSG=STARTOPT
Specifies that the value of the SIRFMSG start option is to be used for this resource. This is the default setting.

SIRFMSG=NONE
Specifies that no messages are issued for any searches involving this application program.

SONSCIP

```
SONSCIP = NO
```

Determines whether the application program receives UNBIND RUs in its SCIP exit routine when the application program is acting as the primary end of a failing session.

SONSCIP=NO
VTAM schedules the application program’s NSEXIT routine with a CLEANUP RU. If the application program does not have an NSEXIT routine, VTAM schedules the LOSTERM exit routine.

SONSCIP=YES
VTAM ends the session for the application program (no CLSDST need be issued). VTAM schedules the application program’s SCIP exit routine with an UNBIND RU that contains information (type codes) describing the failure. For a description of these type codes, see z/OS Communications Server: SNA Programming.

Note: Because secondary application programs receive UNBIND RUs in their SCIP exit routines, it is not necessary for these application programs to code SONSCIP=YES to receive the UNBIND type codes.

SPAN
SPAN = _span_name_

statements:  GROUP, APPL
dependencies:  NetView

 Defines a span of control for VTAM minor node resources. Code this operand if you are using the NetView program. For a full description, see Tivoli NetView for z/OS Administration Reference.

The NetView program checks the SPAN value, but VTAM ignores it.

**SRBEXIT**

SRBEXIT = NO

SRBEXIT = YES

statements:  GROUP, APPL
dependencies:  none

Specifies whether this application program is authorized to use service request block (SRB) processing in its exit routines.

**Note:** If multiple applications open an ACB under one task, the SRBEXIT operand is recognized for the first application opening the ACB. Therefore, the first open ACB determines whether the exit routines execute in SRB or TCB mode. The SRBEXIT operand is ignored for all subsequent applications opening the ACBs under the same task. If SRBEXIT=YES is specified on the first application opening the ACB, the subsequent applications opening ACBs under the same task must be authorized.

**SRBEXIT=NO**

Specifies that the application program is not authorized to use SRB processing in its exit routines.

**SRBEXIT=YES**

Specifies that all the application program’s exit routines are branch-entered in SRB mode, supervisor state, key 0. An application must be APF-authorized, key 0–7, or in supervisor state to open an ACB whose corresponding APPL definition statement has SRBEXIT=YES coded.

SSCPF FM
Specifies the default USS table that is used to send a VTAM operator message to the program operator if the message is not included in a user-defined table. SSCPFM has no effect on the USS table that is used to process VTAM operator commands.

SSCPFM applies only to a program operator application program (AUTH=PPO or AUTH=SPO). It is ignored if you code it for an application program that is not a program operator.

**SSCPFM=USSNOP**
- Specifies that ISTINCNO is the name of the default USS table to be used for VTAM messages sent to this program operator. Note that ISTINCNO is used only for messages that are not included in a user-defined table.

**SSCPFM=USSPOI**
- Specifies that ISTCFCMM is the name of the default message table to be used for VTAM messages sent to this program operator. Note that ISTCFCMM is used only for messages that are not included in a user-defined table. After VTAM searches ISTCFCMM, it searches ISTINCNO. This value is recommended if you run command lists.

**SYNCLVL**
SYNCLVL = CONFIRM
Specifies that synchronization levels of NONE and CONFIRM are supported by this application program. This logical unit supports conversation-specified confirmation processing, but not sync point processing.

SYNCLVL = NONE
Specifies that the application program does not support conversation-specified confirmation processing or sync point processing.

SYNCLVL = SYNCPT
Specifies that synchronization levels of SYNCPT, CONFIRM, and NONE are supported. This logical unit supports both conversation-specified confirmation and sync point processing.

TYPE

statements: GROUP, APPL
dependencies: APPC=YES
VBUILD

Specifies that this VBUILD definition statement defines an application program major node.

USSTAB

Specifies a user-defined USS table that contains user-modified VTAM operator commands and VTAM operator messages to be used by the program operator.

USSTAB applies only to a program operator application program (AUTH=PPO or AUTH=SPO). It is ignored if you code it for an application program that is not a program operator.

If there is no USSTAB operand in the definition for a program operator:

- VTAM uses the IBM-supplied USS table (ISTINCNO) when processing USS commands issued by the program operator.
- VTAM uses the SSCPFM operand to determine which USS table to use when sending a VTAM operator message to a program operator, as described in the information on the SSCPFM operand.

VARYWLD
Indicates whether the program operator application can issue VARY commands containing wildcards.

**VARYWLD=NO**
- Specifies that wildcards are not permitted from this program operator application.

**VARYWLD=YES**
- Specifies that wildcards are permitted from this program operator application when the VARYWLD start option is FULLWILD or POAONLY.

**VCNS**

Specifies whether VTAM is allowed to perform VTAM Common Network Services (VCNS) functions.

**Note:** If you use VTAM Common Network Services, you must code VCNS=YES.

**VCNS=NO**
- Specifies that VCNS functions are not allowed.

**VCNS=YES**
- Specifies that VCNS functions are allowed.
Specifies whether VTAM performs session-level LU-LU verification during activation of LU-LU 6.2 sessions. The related SECLVL options specify the type of verification. If you do not code SECLVL, VTAM attempts to use the enhanced protocol if VERIFY=OPTIONAL or VERIFY=REQUIRED is specified, but will accept basic protocol.

If you code VERIFY=OPTIONAL or VERIFY=REQUIRED, a security management product, such as RACF® 1.9 or later, must be installed and active, and a profile for the LU needs to be in the security management product's database. With RACF 1.9, the APPCLU class needs to be active and a profile of the LU needs to be in the APPCLU class; if not, the OPEN ACB will fail. If the application specifies PARMS=(NQNAMES=YES) on the ACB, the profile for the LU must include the remote partner LU’s network identifier. See z/OS Communications Server: SNA Programmer’s LU 6.2 Guide for more information.

**VERIFY=NONE**

Specifies that no verification of the partner LU’s identity takes place during session activation.

Note that a parameter conflict results if you code SECLVL on the same application program major node as VERIFY=NONE.

**VERIFY=OPTIONAL**

Specifies that identity verification is performed for certain partner LUs during session activation. Determination for which partner LUs the LU-LU verification is performed depends on whether there is a password defined for the LU-LU pair in the installed security manager product. See z/OS Communications Server: SNA Network Implementation Guide for information about how VTAM determines when to perform LU-LU verification.

**VERIFY=REQUIRED**

Specifies that VTAM verifies the identity of all partner LUs during activation of sessions between LU 6.2 applications. Every partner LU must have an LU-LU password defined. Any partner LUs that do not have an LU-LU password defined cannot establish LU 6.2 sessions with this application program.

**VPACING**
VPACING = number of requests

Specifies the maximum number of normal-flow requests that another logical unit can send to this application program in a session before waiting to receive a pacing response. This value, which controls the pacing of requests to the application program, is exchanged when a session is established.

If the value is greater than 63, the default value is used. No pacing of requests to the application program is done if VPACING=0.

**Note:** Subarea VTAM and NCP support sessions between dependent LUs that are not paced. However, APPN sessions require pacing, so if you do not specify a value for VPACING, VTAM uses its default to ensure pacing.

The value of VPACING can be expressed with leading zeros and is limited to eight digits.

For more information about pacing, see [z/OS Communications Server: SNA Network Implementation Guide](https://pubs.opengroup.org/onlinepubs/009691439/bxref_z-os-comms-svr_sna_netw_implemnt_guide/)

**VTAMFRR**
VTAMFRR = NO
VTAM returns control after purging any VTAM FRRs.

VTAMFRR = YES
A VTAM FRR always takes effect when the application program:
• Receives control back from an SRB-authorized path application programming interface (API) request
• Receives control in an SRB exit routine

Channel-attachment major node
To define a channel-attachment major node, code a VBUILD definition statement for the major node and lower-level definition statements for each minor node.

The channel-attachment major node defines the following types of support:
• Channel-to-channel adapter (CTCA)
• Multipath channel (MPC)
• Channel-attached NCP

Each link group has a unique GROUP definition statement and contains a unique set of minor node statements. The operands and default values also vary according to the particular link group being defined.

Channel-to-channel adapter support
To define channel-to-channel adapter (CTCA) support, code the following definition statements:

VBUILD
GROUP LNCTL=CTCA
LINE
 PU

VBUILD definition statement
See "VBUILD definition statement" on page 89 for details on the VBUILD definition statement.

GROUP definition statement
Code a GROUP definition statement to define a channel-to-channel line group.
LINE definition statement
Code one LINE definition statement for each channel-to-channel adapter. The LINE
definition statement defines the characteristics of the line to VTAM.

PU definition statement
Code one PU definition statement for each LINE definition statement. VTAM uses
the PU definition statement to define the link station for an adjacent host
processor.

Table 5. Channel-attachment (CTCA) major node definition statements and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=CA</td>
<td>102</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>GROUP</td>
<td>Lnctl=Ctca</td>
<td>94</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>LINE</td>
<td>ADDRESS</td>
<td>91</td>
<td></td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
<td>ISTATUS</td>
<td>94</td>
<td></td>
<td>G</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
<td>MAXBFRU</td>
<td>95</td>
<td></td>
<td>G</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
<td>MiH</td>
<td>97</td>
<td></td>
<td>G</td>
<td>NO</td>
<td>Lnctl=SDLC</td>
</tr>
<tr>
<td>name</td>
<td></td>
<td>SPAN</td>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td>DELAY</td>
<td>93</td>
<td></td>
<td>G,L</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
<td>ISTATUS</td>
<td>94</td>
<td></td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
<td>Putype</td>
<td>99</td>
<td></td>
<td>G,L</td>
<td>4</td>
<td>NetView</td>
</tr>
<tr>
<td>name</td>
<td></td>
<td>SPAN</td>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
<td>TGN</td>
<td>102</td>
<td></td>
<td>1</td>
<td>PUTYPE=4, 5</td>
<td></td>
</tr>
</tbody>
</table>

Legend:
R     Required operand.
G     Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or
code it on this definition statement.
L     Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code
it on this definition statement.

Multipath channel support
To define multipath channel (MPC) support, code the following definition
statements:

```
VBUILD
GROUP LNCTL=MPC
LINE
PU
```
**VBUILD definition statement**
See “VBUILD definition statement” on page 89 for details on the VBUILD definition statement.

**GROUP definition statement**
Code a GROUP definition statement to define a multi-path channel connection between two hosts.

**LINE definition statement**
Code only one LINE definition statement for multi-path channel support. If you code more than one LINE definition statement for the group, the activation will fail for the entire group. The LINE definition statement defines the characteristics of the line to VTAM.

**PU definition statement**
Code one PU definition statement for the LINE definition statement. VTAM uses the PU definition statement to define the link station for an adjacent host processor.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=CA</td>
<td>102</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFGDS</td>
<td>92</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFGPW</td>
<td>93</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>GROUP</td>
<td>LNCTL=MPC</td>
<td>94</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>REPLYTO</td>
<td>100</td>
<td></td>
<td>3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>LINE</td>
<td>READ</td>
<td>100</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WRITE</td>
<td>103</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISTATUS</td>
<td>94</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXBFRU</td>
<td>95</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MPCDYN</td>
<td>97</td>
<td>G</td>
<td>NO</td>
<td></td>
<td>Multiple WRITE devices must be defined</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td>ISTATUS</td>
<td>94</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PUTYPE</td>
<td>99</td>
<td>G,L</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TGN</td>
<td>102</td>
<td></td>
<td></td>
<td></td>
<td>PUTYPE=4, 5</td>
</tr>
</tbody>
</table>

Legend:
- **R** Required operand.
- **G** Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.
- **L** Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code it on this definition statement.
Channel-attached NCP Support

To define a channel-attached NCP, code the following definition statements:

```
VBUILD
GROUP LNCTL=NCP
LINE
PU
```

**VBUILD definition statement**

See “VBUILD definition statement” on page 89 for details on the VBUILD definition statement.

**GROUP definition statement**

Code a GROUP definition statement to define channel-to-NCP link groups. If you define a line in more than one line group, only one group can be active at a time.

**LINE definition statement**

Code one LINE definition statement for each channel-to-NCP link. The LINE definition statement defines the following characteristics to VTAM:

- The name for the link
- The device number for the link
- Certain procedural options to be used for this line

**PU definition statement**

Code one PU definition for each LINE definition statement. VTAM uses the PU definition statement to define the link station for the adjacent communication controller.

You can define a channel attachment between a host and an NCP in a channel-attachment major node. This data host attachment eliminates the requirement that the host have ownership (control) of network resources. With this channel attachment defined, VTAM can contact a channel-attached NCP without activating the NCP.

When you define a channel attachment between a host and an NCP, be sure to consider the following items:

- Providing a data host to back up a communication management configuration (CMC) host
- Coding the PCCU definition statement in a CMC
- Coding the HOST definition statement in a CMC

For more information on these considerations, see the z/OS Communications Server: SNA Network Implementation Guide.

**Coding the PCCU definition statement**

Several operands on the PCCU definition statement are used to define channel characteristics. If you do not define a channel in a channel-attachment major node, the values on the PCCU definition statement are used because the channel is implicitly defined and activated automatically when the NCP is activated.

However, when you define the channel in a channel-attachment major node, the definition statements in the channel-attachment major node contains operands similar to those on the PCCU definition statements.
For a data host, the CHANCON, MAXDATA, and DELAY operands on the PCCU definition statement are replaced by the same operands on the PU definition statement in the channel-attachment major node. In addition, the CUADDR operand on the PCCU definition statement is replaced by the ADDRESS operand on the LINE definition statement in the channel-attachment major node.

Activating the channel-attachment major node to activate the channel causes VTAM to use the values defined in the channel-attachment major node. Later, if the NCP is activated, these same operands, if they appear on the NCP’s PCCU definition statements, are ignored. If you define the channel in a channel-attachment major node and would like to avoid confusion concerning these operands, avoid coding duplicate values on the PCCU definition statement.

The RNAME operand on the PCCU definition statement can name a link station associated with a channel link defined in a channel-attachment major node. When the host activates each NCP, the link station and link defined in the channel-attachment major node representing the channel are activated automatically.

**Coding the HOST definition statement**

The HOST definition statement in the NCP major node has a MAXBFRU operand for defining the number of buffers that the host can use for receiving data from the NCP.

If you do not define the channel in a channel-attachment major node, the MAXBFRU value on the HOST definition statement is used because the channel is implicitly defined and activated automatically when the NCP is activated.

However, when you define the channel in a channel-attachment major node,

- The MAXBFRU operand on the LINE definition statement for that major node defines the number of buffers the host can use for receiving data.
- VTAM ignores the MAXBFRU value on the HOST definition statement when the link station for the channel is activated automatically. This is because the value is determined by the RNAME operand when the NCP is activated.

**Recommendation:** The MAXBFRU operand on the LINE definition statement of the CA definition should contain the same value as the MAXBFRU operand specified on the HOST definition statement in the NCP major node definition.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=CA</td>
<td>102</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFIGDS</td>
<td>92</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFIGPW</td>
<td>93</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>GROUP</td>
<td>LNCTL=NCP</td>
<td>94</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>name</td>
<td>LINE</td>
<td>ADDRESS</td>
<td>91</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISTATUS</td>
<td>94</td>
<td>G</td>
<td>ACTIVE</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXBFRU</td>
<td>95</td>
<td>G</td>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 7. Channel-attachment (NCP) major node definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPAN</td>
<td></td>
<td></td>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td>CHANCON</td>
<td>92</td>
<td>G,L</td>
<td>COND</td>
<td></td>
<td>LNCTL=NCP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DELAY</td>
<td>93</td>
<td>G,L</td>
<td></td>
<td>0</td>
<td>LNCTL=NCP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISTATUS</td>
<td>94</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXDATA</td>
<td>96</td>
<td>G,L</td>
<td>65535</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PUTYPE</td>
<td>99</td>
<td>G,L</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPAN</td>
<td></td>
<td></td>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>TGN</td>
<td></td>
<td></td>
<td>102</td>
<td>1</td>
<td></td>
<td>PUTYPE=4, 5</td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R  Required operand.
G  Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.
L  Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code it on this definition statement.

Channel-attachment major node coding notes

This information contains coding notes for the channel-attachment major node.

**VBUILD definition statement**

Code one VBUILD definition statement for each channel-attachment major node.

You can file the channel-attachment definition statements for the different types of support as one major node or as separate major nodes. Use operational considerations such as activation processing overhead or switched-network backup capabilities to determine a practical number of major nodes.

If multiple support definition statements are contained in a major node, code only one VBUILD definition statement for the major node.

**Serial channel-to-channel mode**

Serial channel-to-channel (SCTC) mode should be specified for any VTAM CTC. Do not specify basic channel-to-channel (BCTC) mode. The mode is specified in an IOCP deck’s CNTLUNIT statement as the UNIT= operand (for example, UNIT=SCTC).

**Channel-attachment major node full syntax**

The full syntax for the channel-attachment major node follows. The syntax for each operand is repeated in “Channel-attachment major node operand descriptions” on page 91.

**VBUILD**
Notes:
1 MAXBFRU defaults to 1 when LNCTL=CTCA or LNCTL=MPC.
2 MAXBFRU defaults to 3 when LNCTL=NCP.

**PU**

```
name PU
  CHANCON
    COND
    UNCOND
  CHANCON
    COND
    UNCOND
  ISTATUS
    ACTIVE
    INACTIVE
  MAXDATA
    65535
    number_of_bytes
  PUTYPE
    4
  SPAN
    span_name
  TGN
    1
    ANY
    tg_number
```

Notes:
1 DELAY defaults to 0 when LNCTL=NCP is coded.
2 DELAY defaults to 0 when LNCTL=CTCA is coded.

**Channel-attachment major node operand descriptions**

The following information describes the operands you can code for the channel-attachment major node.

**ADDRESS**
ADDRESS

DEVICE ADDRESS = device_address

Specifies the three- or four-digit device number of the channel-to-channel adapter (CTCA) or the channel-attached NCP. If you do not code this operand, the VTAM operator must supply the device number by specifying the U operand of the VARY ACT command. See z/OS Communications Server: SNA Operation for details on using the U operand.

Note: For a channel-attached NCP, this channel-link definition might conflict with a channel-link name automatically defined by VTAM if you attempt to have this channel definition and an automatically defined channel link for the same device number active at the same time. (A channel link is defined automatically if an NCP is activated and a CUADDR value is included on its PCCU definition statement, or if the operator issues a VARY ACT command with U=device number.) The second activation (either this definition or the automatic definition) will fail. For more information on channel-attached NCPs, see the z/OS Communications Server: SNA Network Implementation Guide.

CHANCON

CHANCON = COND

Determines whether VTAM sends a conditional or an unconditional channel Contact request to the NCP when the host is to activate an NCP over a channel.

CHANCON=COND

VTAM allows the NCP to reject the host’s Contact request if the host’s subarea number is already in use by another host (attached to another of the NCP’s channels). This results when duplicate subareas are assigned to more than one host (attached to more than one channel interface). For instance, two hosts might have inadvertently been assigned the same number, or they might have been assigned duplicate numbers to allow mutual backup capabilities.

CHANCON=UNCOND

VTAM sets an indicator in the channel Contact request that forces the NCP to accept the Contact request regardless of whether a host with that subarea number is currently channel-attached to that NCP. If another host using the same subarea number is channel-attached to the NCP, the NCP breaks contact with the host on the other channel.

CONFIGDS
CONFGDS

```
CONFGDS='confgds_name'
```

**statements:** VBUILD  
**dependencies:** none

Specifies a 1–8 character data definition name that identifies the configuration restart data set defined by the user for this major node. Put a data definition (DD) statement that has the CONFGDS name in the VTAM start procedure.

**CONFGPW**

```
CONFGPW='confgds_password'
```

**statements:** VBUILD  
**dependencies:** CONFGDS

Specifies the 1–7 character alphanumeric password that VTAM uses to access the configuration restart data set. If you do not code CONFGPW, but VSAM requires it, VSAM prompts the VTAM operator for the correct password when VTAM attempts to open the data set.

Code CONFGPW only if you also code CONFGDS.

**DELAY**

For LNCTL=CTCA:  
```
DELAY=0
```

For LNCTL=NCP:  
```
DELAY=timeout_value
```

Specifies a delay for network initialization. Use this option to delay the start of VTAM until the network becomes available. This delay is usually 0.

The network defined by the NETWORK statement is available when you code the DEC parameter in the NETWORK statement for the NCP.
For **LNCTL=CTCA:**

- **statements:** GROUP, LINE, PU
- **dependency:** none
- **range:** 0-9.999 seconds

For **LNCTL=NCP:**

- **statements:** GROUP, LINE, PU
- **dependency:** none
- **range:** 0-9.999 seconds

Specifies the maximum amount of time, in seconds and thousandths of seconds, that VTAM waits before transmitting low-priority data (path information unit (PIU) transmission priority 0 or 1) to the physical unit.

For more efficient operation, code this value as high as possible. For maximum speed across the interface, set this value to 0. See the description of the DELAY operand in a channel-to-channel-attached host processor in the [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) for more information.

**ISTATUS**

- **ISTATUS=ACTIVE**
  - The resource becomes active when its major node becomes active.

- **ISTATUS=INACTIVE**
  - The resource remains inactive until the operator activates it.

  The operator can override this value by specifying the SCOPE operand on the VARY ACT command when the major node is activated. See the [z/OS Communications Server: SNA Operation](https://www.ibm.com) for information on using the VARY ACT command.

**LNCTL**
Specifies the type of line support being defined.

**LNCTL=CTCA**

Specifies that the lines are channel-to-channel attachment lines.

**LNCTL=MPC**

Specifies that the lines are multipath channel-attachment lines.

**LNCTL=NCP**

Specifies that the lines are channel-to-NCP lines.

When you code LNCTL=NCP, it is not necessary for VTAM to record the resources attached to the NCP.

**MAXBFRU**

Specifies the amount of storage VTAM uses to receive data from the channel-attached resource once it's activated. The value you code for this operand depends on the type of channel support you are defining.

For CTCA, MPC:

```
maximum_buffer_count
```

Notes:

1. MAXBFRU defaults to 1 when HPR=NO, XID=NO, CONNTYPE=LEN, or the VTAM node is not HPR capable.
2. MAXBFRU defaults to 5 when HPR=YES, XID=YES, CONNTYPE=APPN, and the VTAM node is HPR capable.
For NCP:

\[ \text{MAXBFRU} = 3 \]

\[ \text{MAXBFRU} = \text{number of buffers} \]

**CTCA and MPC:**

The value coded is the number of 4K-buffer pages. The resulting buffer size is \text{number of pages} multiplied by 4K. Make sure that the buffer size is large enough to contain the largest anticipated data transfer.

VTAM automatically substitutes a value of 16 for any coded value larger than 16 without issuing a warning message.

**CTCA:**

Some devices might have a hardware restriction on the maximum number of buffers, but VTAM does not verify the value for those devices. See the appropriate publication for the device before coding this operand.

**MPC:**

The same MAXBFRU value is used for each of the READ subchannels. The resulting buffer size is \text{number of pages} multiplied by 4K. The total buffer space used is 4K times \text{number of pages}, times the number of READ subchannels.

If there is a hardware buffer size for MPC, VTAM compares the MAXBFRU value and the hardware buffer size and uses the smaller of the two to avoid wasting fixed storage. MAXBFRU is not used to determine the buffer size of the WRITE subchannels coded on this host. The WRITE subchannel buffer size is determined dynamically by VTAM depending on the setting at the READ end of the channel.

You can reduce this value after examining VTAM multipath channel tuning statistics so that excessive storage is not allocated for each subchannel. To obtain tuning statistics output, code the TNSTAT start option or use the MODIFY TNSTAT command. See the z/OS Communications Server: SNA Network Implementation Guide for more information on MPC tuning statistics.

**NCP:**

Specifies the number of buffers to be allocated by the host for receiving data from an NCP. This operand has the same meaning as the MAXBFRU operand on the HOST definition statement in the NCP major node.

The value coded for MAXBFRU must satisfy the condition that MAXBFRU times the IOBUF size is less than 65535.

**MAXDATA**
**MAXDATA**

Specifies the maximum number of bytes that the NCP can receive in one segment of a path information unit (PIU), including the transmission header (TH) and the request or response header (RH).

If a MODIFY LOAD command will be issued for a local or remote NCP, the size of MAXDATA must be at least 2082 (2048 plus 34, the size of the PIU header).

**Note:** If you are not defining the channel to the NCP in a channel-attachment major node, code the MAXDATA operand on the PCCU definition statement. Otherwise, code it on the PU definition statement.

See the [z/OS Communications Server: SNA Network Implementation Guide](https://publib.boulder.ibm.com/infocenter/zos/v2r12/topic/com.ibm.zos.r12.cics.doc/cicsinst_pccu escalation.html) for more information about choosing a value for MAXDATA.

**MIH**

Causes the link to become inoperative if a start I/O timeout occurs because the receiving host is in disabled compatibility mode. MIH stands for missing-interrupt handler. The default time value associated with this operand is 3 minutes, although you can change the default with the MIHTMOUT start option.

**MIH=NO**

The MIH default of NO means that VTAM does not deactivate the link if REPLYTO plus MIHTMOUT time units expire without receipt of an interrupt.

**MIH=YES**

If MIH=YES is coded on a CTCA LINE statement (or sifts down to it from the GROUP statement), VTAM deactivates the link if the following time elapses after START I/O is issued from the link: REPLYTO (from the GROUP statement, default 3 seconds) plus MIHTMOUT (start option, default 3 minutes), and no interrupt has yet been received.

**Note:** If you do not code MIH, a link can appear active but is most likely hung or unusable. Therefore, coding MIH=YES is recommended.

**MPCDYN**
Subarea MPC Dynamics support allows subarea MPC groups to support MPC dynamics. This allows subchannel read and write devices to be dynamically added and removed from the MPC group under the system operator's control using the MVS VARY ONLINE and VARY OFFLINE operator commands. Subarea MPC dynamics is functionally equivalent to the APPN MPC dynamics for TRLE based MPC groups (PU Type2). In addition to enabling dynamics, the MPCDYN option will also provide better subchannel utilization by balancing the workload across the entire MPC group. This workload balancing is provided by using a round-robin technique across all the active WRITE subchannels within the MPC group. The third function which the MPCDYN option controls is MPC extended retransmission support. When a write subchannel device (or channel) becomes unavailable because of channel failure, the current and previous write operations are retried over an alternate active write device. The extended retransmission support provides an improvement in MPC group reliability and availability. This support can be very beneficial in hardware configurations that deploy hardware channel extenders. Therefore, it is recommended that the MPCDYN support should be considered for use for configurations using hardware channel extenders.

**MPCDYN=NO**

The MPCDYN default of NO means that VTAM will not allow subarea MPC dynamics and will not extend the MPC DLC retransmission support. Data will be retransmitted on an MPC write subchannel device only if the current write operation fails. After a write operation completes, the data is no longer eligible for retransmission.

**MPCDYN=YES**

If MPCDYN=YES is coded on an MPC group statement, VTAM will enable subarea MPC dynamics for this MPC group and extend the MPC DLC retransmission capability for a longer period of time. Outbound data will be preserved for one additional write operation. If a channel error is detected prior to the completion of the next write operation, the previous write data will be retransmitted over another active subchannel write device within the same MPC group.

**Rule:** The MPCDYN function must be enabled on both sides of the MPC connection at the same time. Each side of the MPC connection must have the MPCDYN parameter configured the same as the partner. If MPCDYN=YES is coded on one side of the MPC connection, the partner must also code MPCDYN=YES. If the MPCDYN function is not enabled on both hosts at the same time, incompatibilities exist which may cause the MPC connection to hang.

**name**

Assigns the name to the resource being defined. The name is used as the network name for network resources.
On **VBUILD**: 

*name* is optional. If a syntax or definition error is detected with this statement during activation of the major node, *name* appears in the message issued to identify the error.

The optional *name* you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node's status.

On **CLUSTER**: 

*name* is required. It provides the minor node name of the cluster controller represented by this definition statement.

On **GROUP**: 

*name* is required. It provides the minor node name of the line group represented by this definition statement.

On **LINE**: 

*name* is required. It provides the minor node name of the line represented by this statement.

On **LU**: 

*name* is required. It provides the minor node name of the logical unit represented by this definition statement.

On **PU**: 

*name* is required. It provides the minor node name of the physical unit represented by this definition statement.

On **TERMINAL**: 

*name* is required. It provides the minor node name of the logical unit (terminal) represented by this definition statement.

**PUTYPE**
PUTYPE = 4

Specifies the physical unit type. PUTYPE=4 is the only valid value.

For an MPC node, when any other value is coded, an error message is issued and the value is changed to the default value of 4.

READ

Specifies the hexadecimal channel_unit_address used to read data from the adjacent host. You must code at least one READ channel_unit_address. If you code an address that is not defined to the system, the activation will fail for the entire group.

Channel_unit_address can be defined as a single address, as a range of addresses, or both. A range is indicated by a hyphen. VTAM generates addresses for the range. For example, READ=(132, 128-130, 13A) will define the addresses 132, 128, 129, 12A, 12B, 12C, 12D, 12E, 12F, 130, and 13A.

For each address on the READ operand, you must code the corresponding address on the WRITE operand in the adjacent host to provide a complete path.

Do not code the same channel_unit_address for the READ operand and the WRITE operand in the same host.

Note: All READ and WRITE channels that you configure must have similar physical characteristics. If some channels take different physical paths with longer distances, this might result in data arriving out of order on the partner host. The out of order conditions add extra processing on the receiving side to reorder the inbound data. The reordering might result in performance degradation.

REPLYTO
Specifies the amount of time VTAM will wait for completion of an operation. If this timeout expires during normal operation, a message is written to inform the operator that a timeout has occurred.

A 3.0 seconds timeout_value should be adequate unless the attached resource is disabled for long intervals.

**CTCA:**
Specifies the number of seconds that VTAM waits for completion after starting a channel program.

A STATION INOP can be scheduled at this time if a missing interrupt handler (MIH) purge has already taken place. The differences between the MIH and REPLYTO operands are:

- The REPLYTO operand causes a message to be generated if a channel program is not completed after the time specified (or 3 seconds, if no time is specified).
- The MIH=YES operand causes the link to become inoperative, defaulting to 3 minutes after the start I/O, if a timeout occurs because of the receiving host.

**MPC:**
Specifies how long VTAM waits for completion of an MPC XID I/O operation after starting a channel program. After the XID completes, REPLYTO has no meaning.
**SPAN**

```
[SPMN=span_name]
```

**statements:**
GROUP, LINE, PU, LU, CLUSTER, TERMINAL

**dependencies:**
NetView

Defines a span of control for VTAM minor node resources. Code this operand if you are using the NetView program. For a full description, see the IBM Tivoli NetView for z/OS Installation: Configuring Additional Components.

The NetView program checks the SPAN value, but VTAM ignores it.

**TGN**

```
<table>
<thead>
<tr>
<th>TGN=1</th>
</tr>
</thead>
<tbody>
<tr>
<td>TGN=ANY</td>
</tr>
<tr>
<td>TGN=tg_number</td>
</tr>
</tbody>
</table>
```

**statements:**
PU

**dependencies:**
PUTYPE=4 or 5

**range:**
1–255

Assigns a transmission group (TG) number that represents the channel connecting two subareas. The TG number (assigned to the same channel) coded in the two subareas must match.

**TGN=tg_number**

Specifies a decimal number assigned to the transmission group.

**TGN=ANY**

Allows the adjacent subarea node to accept whatever valid transmission group number it receives on the exchange identifier (XID). You can specify TGN=ANY only for the PU at one end of a transmission group. The PU at the other end must have a valid transmission group number or default to 1. An error occurs if you specify TGN=ANY for both PUs.

**TYPE**
Means that the VBUILD definition statement defines a channel-attachment major node to VTAM.

**WRITE**

```plaintext
statements: LINE
dependencies: LNCTL=MPC
range: X'000'–X'FFFF'
```

Specifies the hexadecimal channel_unit_address used to write data to the adjacent host. If you specify an address that is offline or not defined to the system, you must code at least one WRITE channel_unit_address.

Channel_unit_address can be defined as a single address, as a range of addresses, or both. A range is indicated by a hyphen. VTAM generates addresses for the range. For example, WRITE=(132, 128-130, 13A) will define the addresses 132, 128, 129, 12A, 12B, 12C, 12D, 12E, 12F, 130, and 13A.

For each address on the WRITE operand, you must code the corresponding address on the READ operand in the adjacent host to provide a complete path.

Do not code the same channel_unit_address for the READ operand and the WRITE operand in the same host.

**Note:** All READ and WRITE channels that you configure must have similar physical characteristics. If some channels take different physical paths with longer distances, this might result in data arriving out of order on the partner host. The out of order conditions add extra processing on the receiving side to reorder the inbound data. The reordering might result in performance degradation.

**Cross-domain resource major node**

The cross-domain resource (CDRSC) major node is used to define:

- Resources in another domain
- Independent LUs

To define one or more cross-domain resource (CDRSC) major nodes, code a VBUILD definition statement, an optional GROUP definition statement, and a CDRSC definition statement for each minor node.

In an SNA-interconnected network, define a CDRSC major node with a VBUILD definition statement, optional NETWORK definition statements, and one or more CDRSC definition statements.
If VTAM is part of a multiple-domain network, two definitions with the same resource name (a logical unit and a CDRSC) can co-exist. In a backup and recovery situation where one host is assuming ownership of a logical unit from another host in the same network, the logical unit can be activated by the new host even though an application program within it currently has a cross-domain session with the logical unit. If the physical units and logical units being recovered support ACTPU(ERP) and ACTLU(ERP) requests, the sessions with the logical units are not affected when the physical units and logical units are activated. The CDRSC definition automatically becomes a shadow resource and the logical unit is now defined as a same-domain (APPL, LU, or LOCAL) resource. If the current host relinquishes ownership of the logical unit, it does so by releasing or deactivating the logical unit and its associated physical unit, causing VTAM to activate its CDRSC again, making it a cross-domain resource.

Independent LU

There are several ways to define independent LUs:
- Dynamically define CDRSCs
- Statically define CDRSCs
- Statically define LUs (LOCADDR=0)
- Statically define model CDRSCs, which are then used to build clone CDRSCs.
  Clone CDRSCs are then treated the same as statically defined (predefined) CDRSCs. References to predefined CDRSCs in this topic can be assumed to include clone CDRSCs.

See [z/OS Communications Server: SNA Network Implementation Guide](#) for detailed information about the methods of defining independent LUs. This information contains a description of the definition statements for statically defining an independent LU through the CDRSC definition statement.

Registering independent LUs

Because the characteristics of an independent LU and of the node on which it resides are not recorded by VTAM, it is difficult to determine the appropriate level of registration for all cases. Therefore, independent LUs are not registered by default because registration of the same resource from multiple nodes can result in the generation of alerts in the APPN network or sub-optimal routing of LU-LU sessions.

However, if an independent LU is owned by a LEN node (that is not acting as an end node in the network), the registration of the independent LU is recommended. Because LEN nodes cannot participate in APPN searches, their LUs cannot be found unless the LUs are defined to an APPN node connected to the LEN node. In this case, you can code the REGISTER operand on the CDRSC or LU definition to register the independent LU to the network node server, or to the network node server and the central directory server. If a large group of independent LUs requires registration, the REGISTER operand can be specified on a higher level definition statement to take advantage of the sift-down effect.

**Note:** Do not specify the REGISTER operand on CDRSC statements on which the CPNAME operand has been coded because the CPNAME operand indicates that the independent LU resides on another APPN node that will be responsible for registering the LU. If the two operands are coded on the same CDRSC statement, either explicitly or by the sift-down effect, the REGISTER operand will default to NO and be ignored.
If you dynamically change an independent LU, the rules cited in “Rules for multiple definition of resources” are followed.

**APPN cross-domain resources**

Defining APPN cross-domain resources is optional, and it is recommended that you not define them to save time. However, there are cases where you might need to define APPN cross-domain resources. For instance, if you want APPN and LEN connectivity to a given independent LU, then both APPN and LEN adjacent link stations need to be defined in the independent LU's adjacent link station list.

In an APPN network, you can define APPN cross-domain resources at network nodes and end nodes. If you decide to define APPN cross-domain resources at an end node, the APPN directory will not be updated, but the information will be used to improve searching.

To define an APPN cross-domain resource, code the CPNAME operand on the CDRSC definition statement. When you code CPNAME, the APPN directory is updated with the location of the cross-domain resource. However, if the resource is found in the directory to have been registered or dynamically defined, the update is ignored, but no message is generated.

The CPNAME and NETSRVR operands are both valid on the CDRSC definition statement. If both operands have the same value, the resource specified by CPNAME is assumed to be the network node. If CPNAME and NETSRVR have different values, the resource specified by NETSRVR is assumed to be the network node.

If only CPNAME is defined under the CDRSC definition statement, the resource it specifies is assumed to be a network node. If the resource is not a network node, an error message indicates that the APPN information is not valid.

The CPNAME and CDRM operands can be specified on the same CDRSC definition statement to allow a path to be defined through the subarea and APPN networks to reach the resource.

The NATIVE and SUBAREA operands are valid with the CPNAME operand and will be used to update the APPN directory to improve searching. The NATIVE operand will improve searching in multiple subnetwork configurations. The SUBAREA operand will improve searching when the resource can only be reached across a CDRM session.

**Rules for multiple definition of resources**

The connection of an independent LU (CDRSC) to an adjacent link station is only for the duration of the independent LU's sessions. In addition, the independent LU can have multiple concurrent sessions using different adjacent link stations.

The rules for multiple definition of resources are presented in Table 8 on page 106. This table assumes that a CDRSC is currently defined to VTAM (existing resource). It also assumes that the operator issues a command to activate a major node that contains new definition statements (input resource). Multiple definition of resources occurs when the name of an input resource matches the name of an existing resource.
### Table 8. Rules for multiple definition of resources

<table>
<thead>
<tr>
<th>Existing resource</th>
<th>Input resource</th>
<th>Major node after integration</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic CDRSC</td>
<td>Dynamic CDRSC</td>
<td>Not applicable</td>
<td>1</td>
</tr>
<tr>
<td>Dynamic CDRSC</td>
<td>Predefined Independent LU</td>
<td>ISTPDILU</td>
<td>2, 3</td>
</tr>
<tr>
<td>Dynamic CDRSC</td>
<td>Predefined CDRSC</td>
<td>CDRSC major node</td>
<td>2, 3</td>
</tr>
<tr>
<td>Predefined Independent LU</td>
<td>Dynamic CDRSC</td>
<td>Not applicable</td>
<td>1</td>
</tr>
<tr>
<td>Predefined Independent LU</td>
<td>Predefined Independent LU</td>
<td>Not applicable</td>
<td>4</td>
</tr>
<tr>
<td>Predefined Independent LU</td>
<td>Predefined CDRSC</td>
<td>ISTPDILU</td>
<td>2, 3</td>
</tr>
<tr>
<td>Predefined CDRSC</td>
<td>Dynamic CDRSC</td>
<td>Not applicable</td>
<td>1</td>
</tr>
<tr>
<td>Predefined CDRSC</td>
<td>Predefined Independent LU</td>
<td>ISTPDILU</td>
<td>2, 3</td>
</tr>
<tr>
<td>Predefined CDRSC</td>
<td>Predefined CDRSC</td>
<td>Not applicable</td>
<td>4</td>
</tr>
<tr>
<td>Predefined APPL</td>
<td>Predefined CDRSC</td>
<td>CDRSC major node</td>
<td>5</td>
</tr>
<tr>
<td>Predefined LU</td>
<td>Predefined CDRSC</td>
<td>CDRSC major node</td>
<td>5</td>
</tr>
<tr>
<td>MNPS application</td>
<td>Predefined CDRSC</td>
<td>Not applicable</td>
<td>8 on page 107</td>
</tr>
</tbody>
</table>

**Note:**

1. This case does not apply to multiple definition of resources. When a resource currently exists, no new dynamic CDRSC is created or considered.

2. If the two resource definitions have different NETID values, both are specified with NQNMODE=NAME, either by definition or start option, and the existing resource has active or pending sessions, the input resource is not defined to VTAM. If the existing resource does not have any sessions, the input resource’s NETID value is also the resulting resource’s NETID value.

If the two resource definitions have different NETID values and at least one is specified with NQNMODE=NQNAME, either by definition or start option, the resource definitions are not merged.

3. If the two resource definitions have different NETID values and both are specified with NQNMODE=NAME, either by definition or start option, the input resource is not defined to VTAM. This is true whether the existing resource has sessions or not. If the NETID values of the two resource definitions do not conflict, the input resource’s NETID value is also the resulting resource’s NETID value.

If the two resource definitions have different NETID values and at least one is specified with NQNMODE=NQNAME, either by definition or start option, the resource definitions are not merged.

4. If both are specified with NQNMODE=NAME, either by definition or start option, a duplicate definition of the resource occurs, and the input resource is not defined to VTAM.

If the two resource definitions have different NETID values and at least one is specified with NQNMODE=NQNAME, either by definition or start option, the resource definitions are not merged.

5. All operands from the input resource apply to the existing resource.

6. CDRSC operands from the input resource apply to the existing resource.
7. Only LU operands from the input resource apply to the existing resource.

8. Predefining a CDRSC at a network node will fail if an MNPS application is already active using the same name. IST489I is issued.

Exceptions to the merging of multiple resource definitions might occur. These exceptions occur because of different NETID values for the existing and new resource definition, which are both specified with NQNMODE=NAME (either by definition or start option). Changes in the NETID of a given resource can occur only if the resource does not have any existing or pending sessions.

If either the existing resource or the input resource is specified with NQNMODE=NQNAME and they have different NETID values, the resource definitions are not merged.

Merging multiple resource definitions causes operands from the input resource to overlay values in the existing resource only in cases where the operands of the existing resource are default values. Defined values on operands of the existing resource are not overlaid with default values on operands of the input resource. Default values and sifted values from the input resource apply to the existing resource definition as well, so long as they do not override defined values on operands of the existing resource. However, not all operands are necessarily integrated as a result of the second resource definition.

The method used for determining what operands to integrate from the input definition to the existing definition is as follows:

- If the existing resource is a dynamic or predefined CDRSC and the input resource is a predefined independent LU, the following operands on the CDRSC definition statement are integrated:
  - ASLENT
  - ASLTAB
  - DLOGMOD
  - EAS
  - ENCR
  - ENCRTYPE
  - LOGAPPL
  - LUALIAS
  - MDLENT
  - MDLTAB
  - MODETAB
  - NETID
  - PACING
  - RESSCB
  - VPACING

- If the existing resource is a predefined independent LU and the input resource is a predefined CDRSC, the following operands on the CDRSC definition statement are integrated:
  - ALSREQ
  - CDRM

3. LUALIAS is only integrated in some cases. See Table 9 on page 108.
If the existing resource is a predefined CDRSC, the input resource is a predefined CDRSC, they have different NETID values, and both are specified with NQNMODE=NAME, either by definition or start option, the second definition fails. (See note [2 on page 106])

If the two resource definitions have different NETID values and at least one is specified with NQNMODE=NQNAME, either by definition or start option, the resource definitions are not merged.

If the existing resource is a member of ISTPDILU, the input resource is a predefined independent LU, they have different NETID values, and both are specified with NQNMODE=NAME, either by definition or start option, all operands on the LU definition statement are integrated.

If the two resource definitions have different NETID values and at least one is specified with NQNMODE=NQNAME, either by definition or start option, the resource definitions are not merged.

When multiple resource definitions are merged, the resulting values for LUALIAS and NQNMODE are determined by several factors. The following tables show when LUALIAS and NQNMODE values are integrated.

**LUALIAS values after a merge of resource definitions**

If the existing resource is in session and has an LUALIAS name defined to it, the LUALIAS name cannot be removed or changed as a result of a resource definition merge. However, when a resource definition merge occurs because the input resource is a predefined CDRSC without NETID specified, the LUALIAS is removed. See Table 9 for specific examples.

**Table 9. Examples of how LUALIAS values are merged**

<table>
<thead>
<tr>
<th>Existing resource</th>
<th>LUALIAS</th>
<th>Session</th>
<th>Input resource</th>
<th>LUALIAS</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predefined Independent LU</td>
<td>N</td>
<td>No</td>
<td>Predefined CDRSC</td>
<td>Y</td>
<td>[2 on page 109]</td>
</tr>
<tr>
<td>Predefined Independent LU</td>
<td>N</td>
<td>Yes</td>
<td>Predefined CDRSC</td>
<td>Y</td>
<td>[2 on page 109]</td>
</tr>
<tr>
<td>Predefined Independent LU</td>
<td>Y</td>
<td>No</td>
<td>Predefined CDRSC</td>
<td>Y</td>
<td>[2 on page 109] [3 on page 109]</td>
</tr>
<tr>
<td>Predefined Independent LU</td>
<td>Y</td>
<td>No</td>
<td>Predefined CDRSC</td>
<td>N</td>
<td>[3 on page 109] [4 on page 109]</td>
</tr>
<tr>
<td>Predefined Independent LU</td>
<td>Y</td>
<td>Yes</td>
<td>Predefined CDRSC</td>
<td>Y</td>
<td>[1 on page 109]</td>
</tr>
</tbody>
</table>

---

4. The list of adjacent link stations from the input resource and the existing resource are always combined.

5. NQNMODE is only integrated in some cases. See “NQNMODE values after a merge of resource definitions” on page 109.
### Table 9. Examples of how LUALIAS values are merged (continued)

<table>
<thead>
<tr>
<th>Existing resource</th>
<th>LUALIAS</th>
<th>Session</th>
<th>Input resource</th>
<th>LUALIAS</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predefined</td>
<td>Y</td>
<td>Yes</td>
<td>Predefined CDRSC</td>
<td>N</td>
<td>1</td>
</tr>
<tr>
<td>Independent LU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Predefined</td>
<td>S</td>
<td>Yes</td>
<td>Predefined CDRSC</td>
<td>S</td>
<td>5</td>
</tr>
<tr>
<td>Independent LU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Predefined</td>
<td>Y</td>
<td>No</td>
<td>Predefined Independent LU</td>
<td>N</td>
<td>2</td>
</tr>
<tr>
<td>CDRSC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Predefined</td>
<td>Y</td>
<td>Yes</td>
<td>Predefined Independent LU</td>
<td>N</td>
<td>2</td>
</tr>
<tr>
<td>CDRSC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Predefined</td>
<td>Y</td>
<td>Yes</td>
<td>Predefined CDRSC</td>
<td>PWN</td>
<td>4</td>
</tr>
<tr>
<td>Independent LU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Predefined</td>
<td>Y</td>
<td>No</td>
<td>Predefined CDRSC</td>
<td>PWN</td>
<td>3</td>
</tr>
<tr>
<td>Independent LU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Predefined</td>
<td>Y</td>
<td>Yes</td>
<td>Predefined Independent LU</td>
<td>PWN</td>
<td>4</td>
</tr>
<tr>
<td>CDRSC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Predefined</td>
<td>Y</td>
<td>No</td>
<td>Predefined Independent LU</td>
<td>PWN</td>
<td>3</td>
</tr>
<tr>
<td>CDRSC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note:**
1. The resource definitions are not merged because the existing resource is in session.
2. The resource definitions are merged, and the LUALIAS of the CDRSC applies to the ILU.
3. The LUALIAS of the ILU is removed.
4. An LUALIAS is no longer associated with the ILU.
5. The resource definitions are merged.
6. An LUALIAS is no longer associated with the CDRSC.
7. The LUALIAS of the CDRSC is removed.

**Symbols in LUALIAS column:**

- **Y:** Resource has an LUALIAS, that is different from the LUALIAS of the resource it is being integrated with.
- **N:** Resource does not have an LUALIAS; either LUALIAS was not specified or it was specified with `no lualias` value.
- **S:** Resource has an LUALIAS that is the same as the LUALIAS of the resource it is being integrated with.
- **PWN:** The resource is predefined without a network identifier, so LUALIAS is not valid.

### NQNMODE values after a merge of resource definitions

NQNMODE values after merges of resources are explained in the following tables:

- Table 10 on page 110
- Table 11 on page 111
- Table 12 on page 112
- Table 13 on page 112
Using the NQNMODE merge tables:

- In Table 10, choose the resources that are being merged (one input resource and one existing resource) by looking at the resource types and the NQNMODE status, and then choosing the corresponding resource type numbers.
- Using the two numbers (one for the input resource and one for the existing resource), look at Table 11 on page 111 for input resources 1–5, or Table 12 on page 112 for input resources 6–10, and find where the column for the input resource intersects the row for the existing resource.

**Note:** Only existing resources can be dynamic, in session, applications, or LUs. So, all resource types in Table 10 are valid for the existing resource, but only resource types 1–10 are valid for the input resource.
- The intersection has a letter followed by numbers in parentheses. These letters and numbers correspond to notes about the merge. These notes are listed in Table 13 on page 112.

**Table 10. Resource types for NQNMODE merges.** This table shows the resource types that are involved in the merge. Information includes whether the resource has sessions, and the current value of NQNMODE. NQNMODE must be either NAME or NQNAME.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>NQNMODE</th>
<th>Defined with NETID</th>
<th>Sessions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - Predefined CDRSC</td>
<td>NQNAME def</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>2 - Predefined CDRSC</td>
<td>NQNAME def</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>3 - Predefined CDRSC</td>
<td>NAME def</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>4 - Predefined CDRSC</td>
<td>NAME def</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>5 - Predefined CDRSC</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>6 - Predefined Independent LU</td>
<td>NQNAME def</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>7 - Predefined Independent LU</td>
<td>NQNAME def</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>8 - Predefined Independent LU</td>
<td>NAME def</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>9 - Predefined Independent LU</td>
<td>NAME def</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>10 - Predefined Independent LU</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>11 - Predefined CDRSC</td>
<td>NQNAME def</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>12 - Predefined CDRSC</td>
<td>NQNAME def</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>13 - Predefined CDRSC</td>
<td>NAME def</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>14 - Predefined CDRSC</td>
<td>NAME def</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>15 - Predefined CDRSC</td>
<td>n/a</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>16 - Predefined Independent LU</td>
<td>NQNAME def</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>17 - Predefined Independent LU</td>
<td>NQNAME def</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>18 - Predefined Independent LU</td>
<td>NAME def</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>19 - Predefined Independent LU</td>
<td>NAME def</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>20 - Predefined Independent LU</td>
<td>n/a</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>21 - Dynamic CDRSC</td>
<td>NAME</td>
<td>n/a</td>
<td>No</td>
</tr>
<tr>
<td>22 - Dynamic CDRSC</td>
<td>NQNAME</td>
<td>n/a</td>
<td>No</td>
</tr>
<tr>
<td>23 - Dynamic CDRSC</td>
<td>NAME</td>
<td>n/a</td>
<td>Yes</td>
</tr>
<tr>
<td>24 - Dynamic CDRSC</td>
<td>NQNAME</td>
<td>n/a</td>
<td>Yes</td>
</tr>
<tr>
<td>25 - Application</td>
<td>n/a</td>
<td>n/a</td>
<td>No</td>
</tr>
<tr>
<td>26 - Application</td>
<td>n/a</td>
<td>n/a</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 10. Resource types for NQNMODE merges (continued). This table shows the resource types that are involved in the merge. Information includes whether the resource has sessions, and the current value of NQNMODE. NQNMODE must be either NAME or NQNAME.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>NQNMODE</th>
<th>Defined with NETID</th>
<th>Sessions</th>
</tr>
</thead>
<tbody>
<tr>
<td>27 - LU</td>
<td>n/a</td>
<td>n/a</td>
<td>No</td>
</tr>
<tr>
<td>28 - LU</td>
<td>n/a</td>
<td>n/a</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 11. Matrix for NQNMODE merges, part 1. In this table, input resources 1–5 are read left to right across the top of the five columns. The existing resources are read top to bottom, in the far left column. Letters and numbers at the intersection of a particular input resource with an existing resource are explained in Table 13 on page 112.

<table>
<thead>
<tr>
<th>Exist. res.</th>
<th>Input resource 1</th>
<th>Input resource 2</th>
<th>Input resource 3</th>
<th>Input resource 4</th>
<th>Input resource 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>2</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>3</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>4</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>5</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>6</td>
<td>M(2 5 12 18)</td>
<td>M(2 5 12 14 18)</td>
<td>M(1 3 12 18)</td>
<td>M(1 3 12 18 14)</td>
<td>M(1 15 18 16)</td>
</tr>
<tr>
<td>7</td>
<td>M(2 5 12 18 14)</td>
<td>M(2 5 12 18)</td>
<td>M(2 5 12 18 6)</td>
<td>M(1 3 12 18)</td>
<td>M(1 15 18 16)</td>
</tr>
<tr>
<td>8</td>
<td>M(2 4 18)</td>
<td>M(2 4 18 14)</td>
<td>M(1 5 18) F(20)</td>
<td>M(1 5 18 14) F(20)</td>
<td>M(1 18 16)</td>
</tr>
<tr>
<td>9</td>
<td>M(1 5 18 14 6)</td>
<td>M(2 4 18 14)</td>
<td>M(1 5 18) F(20)</td>
<td>M(1 5 18 14) F(20)</td>
<td>M(1 18 16)</td>
</tr>
<tr>
<td>10</td>
<td>M(2 4 23 22)</td>
<td>M(2 4 23 22)</td>
<td>M(1 5 22)</td>
<td>M(1 5 22)</td>
<td>M(1 16 5)</td>
</tr>
<tr>
<td>11</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>12</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>13</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>14</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>15</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
<td>F(8)</td>
</tr>
<tr>
<td>16</td>
<td>M(2 5 12 18)</td>
<td>M(2 5 12 14 18)</td>
<td>M(1 3 12 18)</td>
<td>M(1 3 12 14 18)</td>
<td>M(1 15 18 16)</td>
</tr>
<tr>
<td>17</td>
<td>M(2 5 12 13)</td>
<td>M(2 5 12 18)</td>
<td>M(2 5 6 12 13)</td>
<td>M(6 9)</td>
<td>M(11 15 16 18)</td>
</tr>
<tr>
<td>18</td>
<td>M(2 4 12 18)</td>
<td>M(2 4 12 18 14)</td>
<td>M(1 5 18) F(20)</td>
<td>M(1 5 18 14) F(20)</td>
<td>M(1 16 18)</td>
</tr>
<tr>
<td>19</td>
<td>M(1 5 6 12 13)</td>
<td>F(6 9)</td>
<td>M(1 5 18 14) F(20)</td>
<td>M(1 5 18 14) F(20)</td>
<td>M(1 16 18)</td>
</tr>
<tr>
<td>20</td>
<td>M(2 4 17 18)</td>
<td>M(2 4 17 18 14)</td>
<td>M(2 4 17 18)</td>
<td>M(2 4 17 18 14)</td>
<td>M(11 18)</td>
</tr>
<tr>
<td>21</td>
<td>M(2 4 7 12 18 21)</td>
<td>M(2 4 7 12 18 21 14)</td>
<td>M(5 18 21) F(20)</td>
<td>M(5 18 21) F(20)</td>
<td>M(7 11 15 16 18 21)</td>
</tr>
<tr>
<td>22</td>
<td>M(5 18 21)</td>
<td>M(5 18 21)</td>
<td>M(1 3 7 12 18 21)</td>
<td>M(1 3 7 12 18 21 14)</td>
<td>M(7 11 15 16 18 21)</td>
</tr>
<tr>
<td>23</td>
<td>M(2 4 7 12 18 21)</td>
<td>M(2 4 7 12 18 21 14)</td>
<td>M(5 18 21) F(20)</td>
<td>M(5 18 21) F(20)</td>
<td>M(7 11 15 16 18 21)</td>
</tr>
<tr>
<td>24</td>
<td>M(5 18 21)</td>
<td>M(5 18 21)</td>
<td>M(1 3 7 12 18 21)</td>
<td>M(1 3 7 12 18 21 14)</td>
<td>M(7 11 15 16 18 21)</td>
</tr>
<tr>
<td>25</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>26</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>27</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>28</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
</tr>
</tbody>
</table>
Table 12. Matrix for NQNMODE merges, part 2. In this table, input resources are read left to right 6–10 across the top of the five columns. The existing resources are read top to bottom, in the far left column. Letters and numbers at the intersection of a particular input resource with an existing resource are explained in Table 13.

<table>
<thead>
<tr>
<th>Exist. res.</th>
<th>Input resource 6</th>
<th>Input resource 7</th>
<th>Input resource 8</th>
<th>Input resource 9</th>
<th>Input resource 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>M(2 5 12 13)</td>
<td>F(19)</td>
<td>M(1 3 12 13)</td>
<td>F(19)</td>
<td>M(22 15 13 17 2)</td>
</tr>
<tr>
<td>2</td>
<td>M(2 5 12 13 14)</td>
<td>F(19)</td>
<td>M(2 5 12 13 6)</td>
<td>F(19)</td>
<td>M(2 6 15 13 17 22)</td>
</tr>
<tr>
<td>3</td>
<td>M(2 4 13)</td>
<td>F(19)</td>
<td>M(1 5 13) F(20)</td>
<td>F(19)</td>
<td>M(1 22 13 6 17)</td>
</tr>
<tr>
<td>4</td>
<td>M(1 5 13 14 6)</td>
<td>F(19)</td>
<td>M(1 5 13) F(20)</td>
<td>F(19)</td>
<td>M(1 22 13 6 17)</td>
</tr>
<tr>
<td>5</td>
<td>M(2 4 13 17)</td>
<td>F(19)</td>
<td>M(1 5 13 17)</td>
<td>F(19)</td>
<td>M(11 5 13 16)</td>
</tr>
<tr>
<td>6</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>7</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>8</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>9</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>10</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>11</td>
<td>M(2 5 12 13)</td>
<td>F(19)</td>
<td>M(1 3 12 13)</td>
<td>F(19)</td>
<td>M(22 15 13 17 2)</td>
</tr>
<tr>
<td>12</td>
<td>M(2 5 12 13)</td>
<td>F(19)</td>
<td>M(2 5 6 12 13)</td>
<td>F(19)</td>
<td>M(2 6 15 13 17 22)</td>
</tr>
<tr>
<td>13</td>
<td>M(2 4 12 13)</td>
<td>F(19)</td>
<td>M(1 5 13) F(20)</td>
<td>F(19)</td>
<td>M(1 22 13 6 17)</td>
</tr>
<tr>
<td>14</td>
<td>M(1 5 6 12 13)</td>
<td>F(19)</td>
<td>M(1 5 13) F(20)</td>
<td>F(19)</td>
<td>M(1 22 13 6 17)</td>
</tr>
<tr>
<td>16</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>17</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>18</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>19</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>20</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
<td>F(19)</td>
<td>F(8)</td>
</tr>
<tr>
<td>26</td>
<td>F(24)</td>
<td>F(19)</td>
<td>F(24)</td>
<td>F(19)</td>
<td>F(24)</td>
</tr>
<tr>
<td>27</td>
<td>S</td>
<td>F(19)</td>
<td>S</td>
<td>F(19)</td>
<td>S</td>
</tr>
<tr>
<td>28</td>
<td>S</td>
<td>F(19)</td>
<td>S</td>
<td>F(19)</td>
<td>S</td>
</tr>
</tbody>
</table>

Table 13. Notes for NQNMODE merges tables. This table explains the letters and numbers seen in Table 11 on page 111 and Table 12.

<table>
<thead>
<tr>
<th>Letter, number</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>Merge the resources.</td>
</tr>
<tr>
<td>F</td>
<td>Fail the integration request. Do not merge the</td>
</tr>
<tr>
<td></td>
<td>resources. Existing resource remains unchanged.</td>
</tr>
</tbody>
</table>

z/OS V2R1.0 Communications Server: SNA Resource Definition Reference
Table 13. Notes for NQNMODE merges tables (continued). This table explains the letters and numbers seen in Table 11 on page 111 and Table 12 on page 112.

<table>
<thead>
<tr>
<th>Letter, number</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>Input resource is put into shadow. If a shadow already exists, the shadow is treated as the existing resource, and is merged with the input resource.</td>
</tr>
<tr>
<td>1</td>
<td>Resulting NQNMODE value for this resource is NAME.</td>
</tr>
<tr>
<td>2</td>
<td>Resulting NQNMODE value for this resource is NQNAME.</td>
</tr>
<tr>
<td>3</td>
<td>An SRT entry is added for this resource using the netid of this host, if the existing resource is cross net.</td>
</tr>
<tr>
<td>4</td>
<td>The SRT entry is deleted for this resource.</td>
</tr>
<tr>
<td>5</td>
<td>No SRT entry addition or deletion is needed.</td>
</tr>
<tr>
<td>6</td>
<td>Do not override a defined NQNMODE value with a defaulted one. Do not change the NQNMODE value.</td>
</tr>
<tr>
<td>7</td>
<td>When the existing resource is a dynamic CDRSC, the resources will be merged regardless of conflicting NQNMODE values, whether or not the existing CDRSC has sessions. The NQNMODE value of the existing resource becomes the NQNMODE value of the input resource.</td>
</tr>
<tr>
<td>8</td>
<td>It is not valid to have duplicate predefined resource definitions of the same type (for example, two CDRSC definitions, or two ILU definitions).</td>
</tr>
<tr>
<td>9</td>
<td>NQNMODE conflict, and the existing resource is in session. Also, the existing resource has NQNMODE defined.</td>
</tr>
<tr>
<td>10</td>
<td>Conflict in two defined NQNMODE values.</td>
</tr>
<tr>
<td>11</td>
<td>NQNMODE cannot be defined for a predefined alias.</td>
</tr>
<tr>
<td>12</td>
<td>Because NQNMODE=NQNAME on one of the resources, the netids of the input and existing resource must be the same to have a collision.</td>
</tr>
<tr>
<td>13</td>
<td>Only ILU parameters for the existing resources are overridden by parameters from the input resources. The RDTE for the existing resources is moved (if not already there) to ISTPDILU.</td>
</tr>
<tr>
<td>14</td>
<td>NQNMODE is now considered a defined value for the resulting resource.</td>
</tr>
<tr>
<td>15</td>
<td>The existing resource must have a netid of this host's netid for there to have been a collision.</td>
</tr>
<tr>
<td>16</td>
<td>The resulting resource becomes a predefined alias (RCDPDALS in RCRDS is turned on).</td>
</tr>
<tr>
<td>17</td>
<td>RCDPDALS in RCRDS is turned off.</td>
</tr>
<tr>
<td>18</td>
<td>The definition parameters for the existing resources are overridden by parameters from the input resources.</td>
</tr>
<tr>
<td>19</td>
<td>Impossible scenario. NQNMODE cannot be defined on a predefined ILU when the ILU is the input resource. NQNMODE is defined for an ILU only when the ILU has merged with a CDRSC which has NQNMODE defined.</td>
</tr>
<tr>
<td>20</td>
<td>Fail this integration request if the netids of the existing and input resources are different.</td>
</tr>
<tr>
<td>21</td>
<td>Resource becomes a predefined resource.</td>
</tr>
<tr>
<td>22</td>
<td>Resulting resource takes the netid of the input resource.</td>
</tr>
<tr>
<td>23</td>
<td>The input resource must have a netid of this host's netid for there to have been a collision.</td>
</tr>
<tr>
<td>24</td>
<td>The definition should be failed because there might be the case where an appl and an ILU represent the same resource.</td>
</tr>
<tr>
<td>Name</td>
<td>Definition statement</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>name</td>
<td>VBUILD</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>NETWORK</td>
</tr>
<tr>
<td>name</td>
<td>GROUP</td>
</tr>
<tr>
<td>cdrscname</td>
<td>CDRSC</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Definition statement</td>
</tr>
<tr>
<td>-----------</td>
<td>----------------------</td>
</tr>
<tr>
<td>VFYOWNER</td>
<td></td>
</tr>
<tr>
<td>VPACING</td>
<td></td>
</tr>
</tbody>
</table>

Legend:

- **R** Required operand.
- **G** Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.

¹You should not exceed the MAXSESS value coded on the NCP’s BUILD macro. VTAM does not check the value coded here against the value coded on the NCP’s BUILD macro. Session setups will fail if the maximum number of sessions attempted exceeds the MAXSESS value coded on the NCP’s BUILD macro.

**Cross-domain resource major node coding notes**

This information contains coding notes for the cross-domain resource major node.

**VBUILD definition statement**

Code a VBUILD definition statement for each set of CDRSC definition statements.

**NETWORK definition statement**

Code a NETWORK definition statement to identify the network in which the logical units represented by the CDRSC definition statements reside. One or more NETWORK definition statements can appear anywhere after the VBUILD definition statement. If you code consecutive NETWORK definition statements (without intervening CDRSC definition statements), only the last one is used.

If no NETWORK definition statement exists before a CDRSC definition statement, the resource is considered valid for any network. Such CDRSCs can be resources within VTAM’s network or alias names for resources in another network.

**GROUP definition statement**

Code a GROUP definition statement to group cross-domain resources. Each minor node in the group is defined with the values specified on the operands on the GROUP definition statement unless an operand is specified on the CDRSC definition statement.

**Restrictions:**

- The LU_ALIAS operand is not supported on the GROUP definition statement because an LU_ALIAS name must be unique to a CDRSC.
- The DELETE operand is not supported on the GROUP definition statement because DELETE is valid only for model CDRSCs.
CDRSC definition statement

Code CDRSC definition statements to define CDRSC minor nodes or model
CDRSCs (from which clone CDRSCs can be built) for those cross-domain resources
with which an application program or logical unit within this domain can have a
session.

Tips:
1. Information on coding the CDRSC definition statement for TSO/VTAM is in
   the z/OS Communications Server: SNA Network Implementation Guide
2. The same set of CDRSC definition statements can be used throughout the
   network.

Dynamic change of operands for cross-domain resource
major node

You can dynamically change the following CDRSC major node operands by editing
the VTAMLST and then issuing the VARY ACT,UPDATE=ALL command.

   ADJLIST       MDLTAB
   ALSREQ        MODETAB
   ASLENT        NQNMODE
   ASLTAB        PACING
   DELETE        REGISTER
   DLOGMOD       RESCB
   EAS           SIRFMSG
   ENCR          SPAN
   ENCRTYPE      SRCOUNT
   ISTATUS       SRTIMER
   LOGAPPL       VFYOWNER
   MAXSESS       VPACING
   MDLENT

You can also change the NETID operand on the NETWORK statement that applies
to one or more CDRSC definition statements below it, but this change will be
effective only for those CDRSCs that have NQNMODE=NQNAME coded or
defaulted from the current value of the VTAM start option NQNMODE. For any
CDRSC with NQNMODE=NAME, an error message will be issued and the change
to NETID will be ignored.

To dynamically change the operand, the resource to which it applies must be
inactive. You can dynamically change the operands associated with tables
(ASLTAB, MDLTAB, and MODETAB) at any time without deactivating resources.

Model CDRSCs are CDRSCs whose cdrscname value contains wildcards. Clone
CDRSCs are CDRSCs built using model CDRSC definitions. See z/OS
Communications Server: SNA Network Implementation Guide for more
information about model CDRSCs and clone CDRSCs.

Rules: The following rules apply to the use of the VARY ACT,UPDATE=ALL
command to change operands on model CDRSCs:
• The model CDRSC must be inactive to be updated.
• Changes to a model CDRSC do not propagate to already existing clone CDRSCs,
  but will affect future clone CDRSCs created using that model.
• The value of the DELETE operand on the model CDRSC is a characteristic of the model CDRSC that can be changed; the value of the model CDRSC's DELETE operand at the time of clone CDRSC inactivation or clone CDRSC's session termination determines whether or not the clone CDRSC is deleted.
• A VARY ACT,UPDATE=ALL command cannot delete a model CDRSC that has existing clone CDRSCs.
• A VARY ACT,UPDATE=ALL command cannot create a conventional CDRSC when there is an existing clone CDRSC with the same name; delete the clone CDRSC first, using VARY INACT,DELETE=YES, and then use VARY ACT,UPDATE=ALL to add the conventional CDRSC.
• A VARY ACT,UPDATE=ALL command cannot be used to delete clone CDRSCs.

See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) and [z/OS Communications Server: SNA Operation](https://www.ibm.com) for more information about the VARY ACT,UPDATE=ALL command.

**Cross-domain resource major node full syntax**

The full syntax for the cross-domain resource (CDRSC) major node follows. The syntax for each operand is repeated in "Cross-domain resource major node operand descriptions" on page 119.

```plaintext
VBUILD

```name```

VBUILD TYPE=CDRSC

```
CONFGDS=confgd_name

CONFGPW=confgd_password
```

NETWORK
Cross-domain resource major node operand descriptions

The following information describes the operands you can code for the cross-domain resource (CDRSC) major node.
ADJLIST

```
ADJLIST=list_name
```

**statements:**  GROUP, CDRSC  
**dependencies:**  none

Specifies the name of the ADJLIST definition statement which contains the list of adjacent SSCPs to be used for all session requests for this CDRSC.

When an adjacent SSCP list is identified for a CDRSC, session setup requests are sent to only the SSCPs in the list. SORDER and SSCPORD coding does not affect the search sequence for ADJLIST.

**Note:** The ADJLIST operand will be ignored in a pure APPN node. A pure APPN node is a VTAM that is started with start options NODETYPE=NN or NODETYPE=EN, but without HOSTSA or with HOSTSA and SCONNS=NO.

ALSLIST

```
ALSLIST=(pu_name)
```

**statements:**  GROUP, CDRSC  
**dependencies:**  ALSREQ

Specifies a list of adjacent link station names that VTAM can use to gain session connectivity to the independent LU you are defining. The adjacent link station might be currently defined. Validation of the names occurs during session activation.

You can include ISTAPNPU in the list instead of a real adjacent link station name. ISTAPNPU is a generic representation for an APPN adjacent link station. If VTAM finds ISTAPNPU in the adjacent link station list for an independent LU, VTAM treats ISTAPNPU as if it were a real APPN adjacent link station when trying to determine whether to attempt routing through the APPN network.

If ALSREQ=YES, all session requests for this independent LU must be received over an adjacent link station that matches one in the adjacent link station list. For a LEN adjacent link station, the names must be the same to be considered a match. For an APPN adjacent link station, if any other APPN adjacent link station is in the adjacent link station list (including ISTAPNPU) it is considered a match.

**Note:** For a LEN adjacent link station, the NETID must match the NETID value of the independent LU at session establishment. You are responsible for ensuring that PUs named in an adjacent link station list have NETIDs that correspond to the independent LU resource. An APPN adjacent link station has no such restriction.

There might be special ALSLIST requirements for certain migration configurations involving LEN connections. See z/OS Communications Server: SNA Network Implementation Guide for details.
**ALSREQ**

Specifies whether adjacent link station names must match an entry in the default adjacent link station list of the independent LU at a session request.

If you do not code ALSREQ, the value coded on the ALSREQ start option is used.

**ALSREQ=NO**
- Specifies that an adjacent link station name does not have to match an entry in the default adjacent link station list of the independent LU at a session request.

**ALSREQ=YES**
- Specifies that an adjacent link station name must match an entry in the default adjacent link station list of the independent LU at a session request.

If the adjacent link station names are required, only session requests over an adjacent link station which matches one of the adjacent link stations in the list are allowed. An APPN adjacent link station matches any APPN adjacent link stations in the list (including ISTAPNPU). A LEN adjacent link station matches only an adjacent link station in the list that has the same name.

If adjacent link station names are not required, any adjacent link station can receive a session request for this LU. Furthermore, if the adjacent link station is LEN, its name is automatically added to the adjacent link station list for the independent LU. If the adjacent link station is APPN, the generic APPN adjacent link station name (ISTAPNPU) is automatically added to the adjacent link station list for the independent LU instead of the real adjacent link station name.

**ASLENT**
ASLENT

\[ \text{ASLENT} \rightarrow \text{associated lu table entry name} \]

| statements: | GROUP, CDRSC |
| dependencies: | ASLTAB |

Specifies the name of the associated LU table entry to be used for this logical unit.

If you code the ASLENT operand, you must also code the ASLTAB operand. If you omit ASLENT and include ASLTAB, VTAM uses the first entry in the table named in ASLTAB by default.

**ASLTAB**

\[ \text{ASLTAB} \rightarrow \text{associated lu table name} \]

| statements: | GROUP, CDRSC |
| dependencies: | none |

Specifies the name of the associated LU table to be used for the logical unit. See "Associated LU table" on page 981 for a description of how to define an associated LU table.

If you omit the ASLTAB operand, VTAM will not provide the names of associated logical units to the application during LU-LU session initiation. However, you can provide the associated LU names in other ways. You can enter them from the terminal on the PRINTER1 and PRINTER2 operands of the LOGON command, or VTAM can send them to the application program in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the PRINTER1 and PRINTER2 operands of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace an associated LU table (ASLTAB). See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**CDRM**
Specifies the name of the CDRM that controls the CDRSC. The CDRM must be in the same domain as the CDRSC.

If you do not code the CDRM operand, VTAM sends session setup requests to SSCP in the default SSCP list, until either the owning SSCP is found or the end of the list is reached. When the owning SSCP is found, VTAM saves the name of the owning SSCP. If the end of the list is reached, the session setup fails.

If you code CDRM, VTAM moves the CDRM name to the top of the default SSCP list, regardless of whether it is included in the list. Session setup requests are then sent to the SSCPs in the default SSCP list until the owning SSCP is found or the end of the list is reached.

If you coded NETID on the NETWORK definition statement, thereby associating it with the CDRSC, and the CDRM name is not the SSCP name of the CDRM in which the resource resides, the session setup could fail. See “VFYOWNER” on page 143 for more details.

If a NETID is not associated with the CDRSC, the CDRM operand, if coded, is used as an adjacent SSCP. Adjacent SSCP routing will be attempted first, before other entries in the adjacent SSCP table are tried.

**Note:**
1. The preceding actions affect only the internal copy of the adjacent SSCP table maintained for each CDRSC. They do not affect the actual adjacent SSCP table. When the last session for this resource ends, the CDRM name is reset to the CDRM value specified on the CDRSC definition statement.
2. In the SSCP of the originating logical unit (OLU), the CDRM name of the destination logical unit (DLU), if identified, must be correct. If the CDRM name of the DLU is not correct, the operator must issue a MODIFY CDRM command to change the CDRM name.
3. You can permanently change the owning CDRM name with the MODIFY CDRM command. See z/OS Communications Server: SNA Operation for more information about the MODIFY CDRM command.

**cdrscname**

**statements:**
GROUP, CDRSC

**dependencies:**
one
Specifies the name of a resource (a logical unit or application program) in another
domain or network that this CDRSC is to represent.

The `cdrscname` value is required. The `cdrscname` value can be the name of the logical
unit or application as known in this network (the alias name). If the resource is a
VTAM application program, and the CDRSC is to satisfy the real name, `cdrscname`
must be the same as the `name` (not the ACBNAME operand) of the application
program’s APPL definition statement in the VTAM definition in the other domain.

If you do not code the NETWORK statement, the `cdrscname` is either the real name
of the resource or the alias name of the resource. If a CDRSC is predefined in the
same domain for the real name, and another CDRSC is predefined in that domain
for the alias name, the session setup fails. VTAM does not allow two static
definitions of the same resource in a single domain.

In an SNA network interconnection, only one CDRSC identified by the name in
`cdrscname` can be defined using NQNMODE=NAME (either by the start option or
by the definition statement). Same-network CDRSCs and CDRSCs defined without
a NETWORK statement are always considered to be defined using
NQNMODE=NAME. If a CDRSC is defined using NQNMODE=NAME, then
`cdrscname` cannot conflict with the name of any non-CDRSC resource in that
SNA-interconnected network.

In addition to the characters normally acceptable in a SNA name, the `cdrscname` can
include the following wildcard characters:

**Asterisk (*)**
- Represents zero or more unspecified characters. An asterisk can be used as the
  second through eighth character.

**Question mark (?)**
- Represents a single unspecified character. A question mark can be used as the
  first through eighth character.

When wildcard characters are used in the `cdrscname` value, this is considered to be
a model CDRSC definition, from which clone CDRSCs can later be created.

**Restriction:** The `cdrscname` value (including wildcard characters) for all model
CDRSCs must be unique within a given host (including model applications)
regardless of the value of the NQNMODE operand or NQNMODE start option.
For example, two model CDRSC definitions named CD??* in NETA and CD??* in
NETB cannot be activated at the same time on the same host even if
NQNMODE=NQNAME is used for one or both of these model CDRSCs because
the `cdrscname` value is not unique.
**CONFGDS**

```
CONFGDS = confgds_name
```

**statements:** VBUILD  
**dependencies:** none

Specifies a 1–8 character data definition name that identifies the configuration restart data set defined by the user for this major node. Include a DD statement that has this data definition name in the VTAM start procedure.

**CONFGPW**

```
CONFGPW = confgds_password
```

**statements:** VBUILD  
**dependencies:** CONFGDS

Specifies the 1–7 character alphanumeric password that VTAM uses to access the configuration restart data set. If you do not code CONFGPW, but VSAM requires it, VSAM prompts the VTAM operator for the correct password when VTAM attempts to open the data set.

Code CONFGPW only if you code CONFGDS.

**CPNAME**
Identifies the resource's owning control point name. To allow definition of casually connected resources, this operand can be fully qualified with a network ID. If you define an APPN resource to update the APPN directory the CPNAME is required.

CPNAME can be the same name as the host's control point name (SSCP name). However, CPNAME cannot match the name of the CDRSC minor node.

If the CPNAME or NETSRVR name is qualified with a network ID, that network ID is used regardless of the NETID operand value. Otherwise, the value of the NETID operand is used to qualify the name. If NETID is not coded, CPNAME and NETSRVR are qualified with the host's network ID.

Note that for the cross-domain resource major node, CPNAME and REGISTER are conflicting options. If you code CPNAME and a REGISTER value of anything other than NO, an error message is issued and REGISTER is set to NO.

**DELETE**

Indicates whether any clone CDRSC built from this model CDRSC will be deleted when all of the clone CDRSC's sessions have terminated or the clone CDRSC is deactivated.

**DELETE=YES**
Indicates that any clone CDRSC built from this model CDRSC will be deleted when all of the clone CDRSC's sessions have terminated or when that clone CDRSC is deactivated.

**DELETE=NO**
Indicates that any clone CDRSC built from this model CDRSC will be retained when all of the clone CDRSC's sessions have terminated or when that clone CDRSC is deactivated.

**Tip:** Deleting the clone CDRSCs immediately might result in storage thrashing (freeing and getting clone CDRSCs) and search thrashing (losing search history and then rebuilding it). However, retaining the clone CDRSCs means that they continue to consume system resources.

**DLOGMOD**
**DLOGMOD**

Specifies the name of the default logon mode table entry used by default if one is not otherwise provided. If you do not code this operand and the name of a logon mode table entry is not otherwise provided, VTAM uses the first entry in the applicable logon mode table (specified on the MODETAB operand or used by default).

If you specify MODETAB, the entry must be in either the specified table or in ISTINCLM, an IBM-supplied default logon mode table. If you do not specify MODETAB, the entry must be in ISTINCLM. For more information on logon mode entries, see "Logon mode table" on page 1026.

**EAS**

Specifies the estimated number of concurrent sessions that will be active with this independent logical unit.

If your estimated value is greater than 256, for better performance, use the next higher multiple of 256.

You can use this operand in conjunction with the **BSBUF** buffer pool start option to support peak session levels. See "Buffer Pool start option" on page 804 for information about **BSBUF**.
Specifies whether this resource has any special requirements for enciphering and deciphering messages.

**ENCR=NONE**

Specifies that this resource cannot engage in cryptographic sessions.

**ENCR=OPT**

Specifies that this resource can engage in cryptographic sessions, but allows the application program to determine whether to use cryptography.

**ENCR=REQD**

Specifies that VTAM must encipher all messages to and from this resource. If ENCR=REQD, no sessions can be established with this resource unless the host with which it is associated can handle cryptographic sessions.

**ENCR=SEL**

SEL has no meaning for CDRSC resources. If you specify ENCR=SEL, ENCR=OPT is used instead.

**Note:** The setting of ENCR in the Cross Domain Major Node is only acted upon for a resource owned by the VTAM in which it is coded, such as an independent LU (ILU).

**ENCRTYPE**
ENCRTYPE = DES

Specifies that VTAM must use a minimum of DES encryption with an 8-byte key when performing session level encryption. This is the default.

ENCRTYPE = TDES24

Specifies that VTAM must use a minimum of Triple_DES encryption with a 24-byte key when performing session level encryption.

Note: The ENCRTYPE value coded on the GROUP will be sifted down to the appropriate resource (CDRSC).

The SYSDEFed ENCRTYPE of a CDRSC resource can be changed at runtime through the MODIFY SECURITY command. See z/OS Communications Server: SNA Operation for specific details. However, the ENCRTYPE value will not be able to be downgraded. In other words, if the application is currently operating at ENCRTYPE=TDES24, then the encryption level cannot be changed to ENCRTYPE=DES using the MODIFY SECURITY command.

Note: If downgrading the security level is preferred, the current major node being used must be deactivated and another major node value must be coded.

In addition, a user may use the LOGMODE TABLE to override the ENCRTYPE value of the SLU based on the logmode entry being used. The ENCRTYPE value will not be able to be downgraded from TDES24 to DES using the logmode table. Only TDES24 can be specified for the encryption level (ENCRTYP). See "MODEENT macroinstruction" on page 1027 for more information about MODEENT macroinstructions.

ISTATUS
**ISTATUS**

Specifies the initial status of this CDRSC.

**ISTATUS=ACTIVE**

Specifies that this resource is logically active to this domain, which means that session-establishment requests can be sent to the adjacent SSCP using the resource. The resource is not necessarily active in its own domain.

**ISTATUS=INACTIVE**

Specifies that this resource is not logically active to this domain, though it is not necessarily inactive in its own domain. Session requests from this domain to the resource will fail.

**LOGAPPL**

Identifies the name of the primary logical unit (PLU) that this logical unit will automatically log on to as a secondary logical unit (SLU) when this logical unit is activated. The name specified can also be a USERVAR, which will be resolved into its current value before being used to establish or redrive the session with the controlling primary logical unit.

For independent logical units, the LOGAPPL operand establishes a session with the primary logical unit designated on the LOGAPPL operand if a session does not currently exist with this logical unit. The session is established regardless of whether sessions with other logical units exist. The LOGAPPL operand cannot contain the name of an independent logical unit because independent logical units cannot be the controlling logical units for an automatic logon session.

To ensure retry of a pending automatic logon, code the AUTORTRY or AUTOTI start options.

You can specify a network-qualified name for LOGAPPL. You cannot specify an LUALIAS name for LOGAPPL.

**Note:** If you specify the NetView or TSO program as the application program on the LOGAPPL operand, you will get unpredictable results in recovery situations.

For more information on establishing sessions, see the [z/OS Communications Server: SNA Network Implementation Guide](http://www.ibm.com).
LUALIAS

```
LUALIAS = lualias
```

**statements:** GROUP, CDRSC  
**dependencies:** see description

Defines a 1–8 character name used by application programs in this domain to represent a cross-network or cross-domain resource that is defined under a NETWORK definition statement that precedes the CDRSC definition statement.

**Note:** LUALIAS is valid only for cross-domain and cross-network resources that are defined under a NETWORK definition statement, and is recommended only for cross-network CDRSCs which are defined using NQNMODE=NQNAME because same-network CDRSCs and NQNMODE=NAME CDRSCs are known by their unqualified resource names.

Session requests from this domain's application programs are routed to the CDRSC, when the destination resource has the same name as defined by LUALIAS. Session requests from any resource other than a VTAM application in this domain are routed to the resource in this domain with the same name.

If you define two CDRSCs with the same LUALIAS name, only one can be active at a time. This means that a shadow resource with an LUALIAS name that duplicates a real resource with the same LUALIAS name, regardless of whether the resource names for the two resources are the same, cannot be activated until the real resource is deactivated. A shadow resource that attempts to become active and cannot because it has an LUALIAS name that duplicates an active resource is put into RESET state. After it is in RESET state, the major node of the shadow resource must be deactivated, the active resource with the LUALIAS name must be deactivated, and then the major node of the shadow resource must be activated before the shadow resource can be activated.

An LU in shadow (which does not have a duplicate LUALIAS name) also can be activated when it comes out of shadow if the LU’s initial status (ISTATUS operand) was active while it was in shadow, and the CDRSC that held the LU in shadow was not deactivated.

**Note:** LUALIAS takes effect only in the following situations:

- When the LUALIAS name has been defined for a resource that is defined on the CDRSC definition statement that is preceded by a NETWORK definition statement.
- For session requests that cross the application programming interface.
- For session requests whose origin or destination is the VTAM for which LUALIAS is coded. VTAMs along the session setup path are not involved with or affected by LUALIAS.

**Note:** LUALIAS can be used in DISPLAY commands issued in the VTAM where LUALIAS is defined. LUALIAS cannot be used in MODIFY or VARY commands, and cannot be specified for LOGAPPL.

LUALIAS is optional for cross-network or cross-domain resources predefined with the NETWORK definition statement. If it is not defined and there are duplicate
names in the SNA network interconnection, application programs can use the network-qualified name [if PARMS=(NQNAMES=YES) is specified on the ACB macroinstruction] or any of the other types of alias name translation, such as:

- Alias name translation with NetView
- Session management exit routine
- USERVAR

**Restriction:** LUALIAS does not work with ALIAS name translation (regardless of whether translation is performed by NetView or the Session Management Exit). If an ALIAS name is received on a session setup request from a resource that is predefined as a CDRSC with an LUALIAS, then the received ALIAS name is presented to local applications, rather than the predefined LUALIAS name.

If you are predefining this resource just to specify an LUALIAS, consider coding LUAPFX on the APPL definition statement which does not support network-qualified names and allow the CDRSC to be created dynamically. VTAM will dynamically generate an LUALIAS for each dynamic cross-network CDRSC as needed. See “LUAPFX” on page 59 for more information about dynamic LUALIAS.

**MAXSESS**

```
MAXSESS—n—max_number_LU-LU_sessions
```

**statements:** GROUP, CDRSC
**dependencies:** Independent LUs attached to an NCP
**range:** 1–65535

Specifies the maximum number of concurrent LU-LU sessions in which an independent LU can participate. The MAXSESS value applies to each connection the LU is using to establish LU-LU sessions. For example, if MAXSESS=2 was specified for LUA and LUA appears on three PUs, LUA can have two sessions on each of the three PUs for a total of six sessions.

If you do not code MAXSESS, or code a value outside the range, the MAXSESS value set in the NCP definition is used.

**Note:** You should not exceed the MAXSESS value coded on the NCP’s BUILD macro. VTAM does not check the value coded here against the value coded on the NCP’s BUILD macro. Session setups will fail if the maximum number of sessions attempted exceeds the MAXSESS value coded on the NCP’s BUILD macro.

**MDLENT**
**MDLENT**

- **statements:** `GROUP, CDRSC`
- **dependencies:** `MDLTAB`

Specifies the name of the model name table entry to be used for this logical unit.

If you code the MDLENT operand, you must also code the MDLTAB operand. If you omit MDLENT and include MDLTAB, VTAM uses the first entry in the table named in MDLTAB.

**MDLTAB**

- **statements:** `GROUP, CDRSC`
- **dependencies:** `none`

Specifies the name of the model name table to be used for the logical unit. See "Model name table" on page 1054 for a description of how to define a model name table.

If you omit the MDLTAB operand, VTAM will not provide the name of the model name table to the application during LU-LU session initiation. However, you can provide the model name in other ways. You can enter it from the terminal on the MODEL operand of the LOGON command, or VTAM can send it to the application program in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the MODEL operand of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace a model name table. See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**MODETAB**
MODETAB = logon_mode_table_name

Specifies the name of a logon mode table to be used for the logical unit. The name you code must be the name of a logon mode table created as described in “Logon mode table” on page 1026. If you do not code a logon mode table for the logical unit on the MODETAB operand, an IBM-supplied default logon mode table (ISTINCLM) is used. If you specify a table, both the table you specify and the default table are used.

name

On VBUILD:
name is optional. If a syntax or definition error is detected with this statement during activation of the major node, name appears in the message issued to identify the error.

The optional name you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node’s status.

On NETWORK:
Identifies this statement in error messages from the VTAM system definition processor. To avoid confusion, use the same value for both name and the NETID operand. name is optional, but its use is highly recommended.

On GROUP:
Specifies the required name of the CDRSC group.

NATIVE
**NATIVE**

Indicates that the resource is located in the native APPN subnetwork or across an APPN subnetwork boundary.

**NATIVE=**YES

- Indicates the resource is located in this APPN subnetwork, or across an SNI connection that originates in this APPN subnetwork.

**NATIVE=**NO

- Indicates that the resource is located across an APPN network or subnetwork boundary.

**NETID**

Specifies an optional 1–8 character identifier of the network in which the CDRSC, or set of CDRSCs, resides. This NETID pertains to the CDRSC, or set of CDRSCs, defined after the NETWORK definition statement and before any other NETWORK definition statement in the same set of definitions.

**Note:**

1. If a CDRSC is defined with a network ID of the host network, the resource is considered real.
2. A CDRSC defined with a network ID of the host network must reside in the host network.
3. The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers (NETIDs). Although VTAM allows you to use #, other products might enforce this restriction.
Identifies the resource's network node server. To allow predefinition of casually connected resources, this operand can be fully qualified with a network ID. If the operand is not network-qualified, the NETID value coded on the NETWORK statement is used.

The NETSRVR operand can have the same name as the host's control point name if the node is the network node server. However, NETSRVR cannot match the name of the CDRSC minor node.

**Note:** Specifying this operand does not initiate CP-CP sessions with the network node server. The name is used only to suggest a network node server for a control point.

### NQNMODE

Indicates whether VTAM defines the cross-network resource by its non-network-qualified name and its network-qualified name or by only its network-qualified name.

NQNMODE is valid only for cross-domain and cross-network resources that are defined under a NETWORK definition statement. Same-network CDRSCs are always considered to be defined with NQNMODE=NAME because these resources are always identified by their non-network-qualified names.

When NQNMODE is specified on the CDRSC definition statement, it overrides the value specified on the NQNMODE start option. Dynamic CDRSCs are controlled by the NQNMODE start option.

Table 15 on page 137 shows the various ways NQNMODE applies to different types of CDRSCs.

**NQNMODE=NAME**

The cross-network resource is defined by both its network-qualified name and its non-network-qualified name. The same-network CDRSC is always considered to be defined using NQNMODE=NAME because it is always identified by its non-network-qualified name. This is the mode in which pre-V4 releases of VTAM operate.
If a CDRSC is defined using NQNMODE=NAME (either by definition or start option), then no other resource can be defined to that host by the same non-network-qualified name.

**NQNMODE=NQNAME**

The cross-network resources is defined by its network-qualified name and is not defined to this host by its non-network-qualified name.

If you have duplicate names in your SNA-interconnected networks and want to avoid using other forms of alias name translation, you can specify (either by definition or start option) NQNMODE=NQNAME.

<table>
<thead>
<tr>
<th>Table 15. How NQNMODE relates to CDRSCs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of CDRSC</td>
</tr>
<tr>
<td>Predefined CDRSCs with NETID</td>
</tr>
<tr>
<td>Predefined CDRSCs without NETID</td>
</tr>
<tr>
<td>Dynamic CDRSCs defined by their network-qualified names and their implied alias names (consisting of their non-network-qualified names and the network identifiers of their session partners)</td>
</tr>
<tr>
<td>Same-network CDRSCs and native independent LUs</td>
</tr>
<tr>
<td>Nonnatively attached independent LUs defined as LUs with LOCADDR=0 coded</td>
</tr>
<tr>
<td>Nonnatively attached independent LUs defined as CDRSCs</td>
</tr>
</tbody>
</table>

**PACING**
Specifies how VTAM is to pace the flow of data from the boundary node, which performs pacing for a channel-attached SNA device, to the SLU. The PACING value is not used in all configurations. See z/OS Communications Server: SNA Network Implementation Guide.

Note:
1. PACING=0 indicates that no pacing is performed for sessions with the logical unit.
2. You can use the FIXED option on this operand. FIXED is an NCP option that allows you to override adaptive session pacing and use fixed pacing instead. To use the FIXED option, you must code (N,,FIXED). The second comma represents M or a minimum value that is used by some pre-ENA nodes.

For more information on pacing, see z/OS Communications Server: SNA Network Implementation Guide.
REGISTER = NO

REGISTER = CDSERVR

REGISTER = NETSRVR

REGISTER = NO

Specifies how a resource should be registered on a definition statement.

Note that for the cross-domain resource major node, CPNAME and REGISTER are conflicting options. If you code CPNAME and a REGISTER value of anything other than NO, an error message is issued and REGISTER is set to NO.

REGISTER=CDSERVR

Indicates that an end node resource is registered to a network node server and directory resource registration is requested for it. A network node resource is registered at the central directory server. If the node is configured as a central directory server, this value has the same effect as NO.

REGISTER=NETSRVR

Indicates that the end node resource should be registered to the network node server, but that directory resource registration should not be requested for it.

REGISTER=NO

Indicates that the resource should not be registered.

RESSCB
RESSCB = number_of_session_control_blocks

Specifies the number of session control blocks reserved by the NCP for the logical unit.

This operand applies only to type 2.1 peripheral nodes.

For information about NCP restrictions on this operand, see NCP, SSP, and EP Resource Definition Reference.

SIRFMSG

Controls the display of error messages IST663I, IST664I, IST889I, and their message groups. These messages are issued when a session initiation request originated by this CDRSC fails because trial and error routing, using an adjacent SSCP table, exhausted the table and was unable to locate the destination LU. The messages might also be issued when a session initiation request (or searches that may not result in a session being established) to this CDRSC fails for similar reasons. The messages might also be issued when an APPN locate search request fails to locate a target resource.

The SIRFMSG operand setting on the CDRSC controls the ASIRFMSG, DSIRFMSG, ESIRFMSG, FSIRFMSG, LSIRFMSG, RSIRFMSG, and SIRFMSG start option processing. See “Descriptions of start options” on page 794 for more details.

When a value other than SIRFMSG=STARTOPT is specified for a CDRSC, the resource level setting of SIRFMSG overrides whatever SIRFMSG start option setting is in effect at the node. If both resources (origin and destination logical units) for a given session initiation request have SIRFMSG values other than SIRFMSG=STARTOPT specified, then the less restrictive setting (where ALLSSCP is less restrictive than OLUSSCP which in turn is less restrictive than NONE) will be used to determine if the error messages are issued. When CPNAME= is coded on a CDRSC, the SIRFMSG setting will also be used to override the LSIRFMSG start option setting for APPN locate searches.
Tip: When SIRFMSG is used to override the LSIRFMSG start option, then
ALLSSCP is equivalent in function to ALLNNS and OLUSSCP is equivalent in
function to OLUNNS.

SIRFMSG=ALLSSCP
   Specifies that messages are issued for all session initiation request failures
   involving this CDRSC. For APPN locate searches SIRFMSG=ALLSSCP will
   result in failure messages being issued for searches when this node is acting as
   a network node.

SIRFMSG=OLUSSCP
   Specifies that messages are issued only when the VTAM owning this CDRSC is
   the origin logical unit (OLU) SSCP for the failing session initiation request. For
   APPN locate searches, SIRFMSG=OLUSSCP will result in failure messages
   being issued for searches where this node is acting as a network node server
   for the OLU or when acting as the origin directory server of the search.

SIRFMSG=STARTOPT
   Specifies that the value of the SIRFMSG start option is to be used for this
   resource. This is the default setting.

SIRFMSG=NONE
   Specifies that no messages are issued for any searches involving this CDRSC.

SPAN

```
[SPAN=span_name]
```

statements: GROUP, CDRSC
dependencies: NetView

Defines a span of control for VTAM minor node resources. Code this operand if
you are using the NetView program. For a full description, see IBM Tivoli NetView
for z/OS Installation: Configuring Graphical Components.

The NetView Program checks the SPAN value, but VTAM ignores it.

SRCOUNT
Specifies for this resource how many search requests can be limited before VTAM performs another resource discovery search. It is meaningful only when search reduction is active (SRCHRED=ON). This operand overrides the value of the SRCOUNT start option for this CDRSC.

Note:
1. If SRCOUNT=0 is coded, then the search count threshold is not used and the SRTIMER value alone will control when to attempt another search.
2. If neither SRCOUNT or SRTIMER is coded on the CDRSC statement and SRCHRED=ON, then the SRCOUNT and SRTIMER start option values will be used.
3. If SRCOUNT is coded, an APPN directory entry will be created (if one does not exist for the resource) in addition to the CDRSC defined by this definition statement.

SRTIMER

Specifies the number of seconds that a search reduction entry limits incoming search requests or session requests. SRTIMER is meaningful only when search reduction is active (SRCHRED=ON). This operand overrides the value of the SRTIMER start option for this CDRSC.

Note:
1. If SRTIMER=0 is coded, then the search time threshold is not used and the SRCOUNT value alone will control when to attempt another search.
2. If neither SRTIMER or SRCOUNT is coded on the CDRSC statement and SRCHRED=ON, then the SRCOUNT and SRTIMER start option values will be used.
3. If SRCOUNT is coded, an APPN directory entry will be created (if one does not exist for the resource) in addition to the CDRSC defined by this definition statement.

SUBAREA
indicates whether or not a CDRM session must be crossed to reach the resource.

**SUBAREA=NO**
Indicates that the resource can be reached using an entirely APPN path.

**SUBAREA=YES**
Indicates that a subarea CDRM session must be used to reach the resource (for example, the resource is located in or through a subarea network).

**TYPE**

Indicates a CDRSC major node. This operand is required.

**VFYOWNER**
Specifies whether owner verification takes place when sessions are attempted with this resource.

Note: Owner verification is supported only in subarea sessions.

**VFYOWNER=**

- **YES**
  - Specifies that the session is rejected if the owner specified on the CDRSC does not match the owner specified in the session initiation request.
  - VFYOWNER=YES is valid only if:
    - The CDRSC is predefined with the NETID operand on the NETWORK definition statement.
    - You code CDRM=cdrmname.

- **NO**
  - Specifies that, if the owning SSCP received on an initiation request does not match the owner specified on the CDRSC, the CDRSC is updated with the owning SSCP name received, unless the CDRSC is across the APPN-subarea boundary and CDRM=cdrmname is coded.

**VPACING**
VPACING = 2

VPACING = number_of_requests

specifies how VTAM paces the flow of data from the boundary node containing the PLU to the boundary node that performs pacing for a channel-attached SNA device.

**Note:** You can use the FIXED option on this operand. FIXED is an NCP option that allows you to override adaptive session pacing and use fixed pacing instead.

Note: You can use the FIXED option on this operand. FIXED is an NCP option that allows you to override adaptive session pacing and use fixed pacing instead.

To use the FIXED option, you must code (N,,FIXED). The second comma represents M or a minimum value that is used by some pre-ENA nodes.

For more information on pacing, see z/OS Communications Server: SNA Network Implementation Guide.

**Cross-Domain resource manager (CDRM) major node**

To define a cross-domain resource manager (CDRM) major node, code a VBUILD definition statement for the major node, and a CDRM definition statement for each minor node.

You can define one or more major nodes for CDRMs.

For SNA network interconnection, define a cross-domain resource manager major node with:

- A VBUILD definition statement
- Optional NETWORK definition statements
- One or more CDRM definition statements
- Optional GWPATH definition statements

**Note:** Within a domain, one cross-domain resource manager, called an external CDRM, must be defined for every other domain (in this network or another network) with which this domain directly communicates. Each domain must also have only one definition for its own cross-domain resource manager, called the host CDRM.
Table 16. Cross-Domain resource manager major node definition statement and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name VBUILD</td>
<td>TYPE=CDRM</td>
<td>CONFIGDS</td>
<td>169</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFIGPW</td>
<td>157</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name VBUILD</td>
<td></td>
<td></td>
<td>158</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name NETWORK</td>
<td>NETID</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cdrmname CDRM</td>
<td></td>
<td>SUBAREA</td>
<td>168</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CAPACITY</td>
<td>153</td>
<td></td>
<td>8K</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CDRDYN</td>
<td>154</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CDRSC</td>
<td>156</td>
<td></td>
<td>REQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE</td>
<td>158</td>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTTIME</td>
<td>159</td>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DISJOINT</td>
<td>159</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ELEMENT</td>
<td>160</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>GWSELECT</td>
<td>162</td>
<td></td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HPR</td>
<td>162</td>
<td></td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISTATUS</td>
<td>163</td>
<td></td>
<td></td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>NN</td>
<td>165</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY</td>
<td>165</td>
<td></td>
<td></td>
<td>TERRESTR</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RECOVERY</td>
<td>166</td>
<td></td>
<td>YES</td>
<td>UNSECURE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SECURITY</td>
<td>167</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>168</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TGP</td>
<td>169</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>UPARM1</td>
<td>170</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>UPARM2</td>
<td>170</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>UPARM3</td>
<td>170</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>VPACING</td>
<td>171</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>VRTG</td>
<td>171</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>VRTGCPCP</td>
<td>172</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- CONFIGDS
- APPN only
- VRTG=YES; APPN only
- SSCP-SSCP sessions
- VRTG=YES; APPN only
- VRTG=YES; APPN only
- APPN only
- VRTG=YES; APPN only
- cross-network
- CDRMs
- VRTG=YES; APPN only
- VRTG=YES; APPN only
- VRTG=YES; APPN only
- NetView
- VRTG=YES; APPN only
- VRTG=YES; APPN only
- VRTG=YES; APPN only
- APPN only
- VRTG=start option
- APPN only; NODETYPE; HOSTSA
- VRTG=start option
- VRTG=YES; APPN only; NODETYPE; HOSTSA
- VRTG=YES; APPN only; NODETYPE; HOSTSA
Table 16. Cross-Domain resource manager major node definition statement and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>GWPATH</td>
<td>ADJNET</td>
<td>151</td>
<td></td>
<td></td>
<td>netid</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ADJNETCS</td>
<td>152</td>
<td></td>
<td></td>
<td>ISTVTCOS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ADJNETEL</td>
<td>152</td>
<td></td>
<td></td>
<td>1</td>
<td>ADJNET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ADJNETSA</td>
<td>153</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ELEMENT</td>
<td>160</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>GWN</td>
<td>161</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SUBAREA</td>
<td>168</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R Required operand.

Cross-domain resource manager (CDRM) major node coding notes

This information contains coding notes for the cross-domain resource manager major node.

VBUILD definition statement

Code a VBUILD definition statement to define a cross-domain resource manager major node.

NETWORK definition statement

Code the NETWORK definition statement in an SNA-interconnected network, to name the network in which each of the CDRMs being defined resides. The NETWORK definition statement appears after the VBUILD definition statement and immediately before one or more CDRM definition statements defining CDRMs residing in the network.

You can include more than one NETWORK definition statement in a CDRM major node definition. If you include consecutive NETWORK definition statements (without intervening CDRM definitions), only the last one is used.

If no NETWORK definition statement exists before one or more CDRM definition statements, the CDRMs are assumed to reside within this VTAM network. The network name for this network is the name specified on the NETID start option for the VTAM in which the definition is filed.

Note: There are two considerations regarding a receiving (internal) CDRM and session setup:

- If the receiving CDRM is not a gateway SSCP, and its definition for an external CDRM does not include NETID, then its VTAM will accept whatever NETID is sent during session setup, regardless of whether it is correct. The CDRM session will set up, but LU-LU sessions will not. To prevent the above, code GWSSCP=YES on the receiving CDRM definition, or code NETID on the NETWORK definition statement of all external CDRMs.
If the receiving CDRM is a gateway SSCP, and no NETID is coded for an external CDRM, then the NETID sent during session setup must match the value on the NETID start option of the receiving host or the session will not set up.

**CDRM definition statement**

Code a CDRM definition statement to define the host CDRM and each external CDRM that will be in direct communication with this domain. Define the host CDRM before defining any external CDRMs.

**GWPATH definition statement**

Code a GWPATH definition statement to define a possible cross-network session path between the gateway host CDRM and a CDRM in another network.

The gateway path (GWPATH) definition statement is optional. You should code a GWPATH definition statement only if VTAM is started with GWSSCP=YES.

One or more GWPATH definition statements can follow a CDRM definition statement for a CDRM in another network. VTAM processes GWPATH definition statements in the order in which they are assigned. Before VTAM uses the list, the GWPATH definition statement assigned for the SSCP-SSCP session is placed first in the list.

**Note:** The operands on the GWPATH definition statement are listed here in alphabetical order for easy reference. However, you might find it easier to study and code them in this order: GWN, SUBAREA, ELEMENT, ADJNET, ADJNETSA, ADJNETEL, and ADJNETCS.

**Cross-domain resource manager (CDRM) major node full syntax**

The full syntax for the cross-domain manager (CDRM) major node follows. The syntax for each operand is repeated in “Cross-domain resource manager (CDRM) major node operand descriptions” on page 151.
Notes:

1. NEGLIGIB is the default if the TRLE operand is coded.
Cross-domain resource manager (CDRM) major node operand descriptions

The following information describes the operands you can code for the cross-domain resource manager (CDRM) major node.

**ADJNET**
ADJNET

\[ \text{ADJNET} \rightarrow \text{netid_of_preceding_network_def_statement} \]

\[ \text{ADJNET} \rightarrow \text{adjacent_network_identifier} \]

**statements:**

**GWPATH**

**dependencies:**

none

Specifies, for this cross-network session path to the CDRM through the associated gateway NCP, the ID of the network that is adjacent to the gateway SSCP’s network. The gateway NCP is determined either by the GWN operand or by the SUBAREA operand.

If you do not code ADJNET, the adjacent network defaults to the ID of the network defined on the preceding NETWORK definition statement. The value coded must not be the same as the host’s network ID.

**Note:** In a back-to-back configuration, if multiple GWPATH definition statements are filed describing multiple GWN paths to the same CDRM, and those statements specify the same ADJNET, the ADJNETSA and ADJNETEL values must agree on all the GWPATH definition statements.

ADJNETCS

\[ \text{ADJNETCS} \rightarrow \text{ISTVTCOS} \]

\[ \text{ADJNETCS} \rightarrow \text{adjacent_network_cos_name} \]

**statements:**

**GWPATH**

**dependencies:**

none

Specifies the 8-character name that defines the Class of Service (CoS) table entry to be used for the SSCP-SSCP session. The table entry is located in the network that is adjacent to the gateway SSCP’s network.

This operand must be coded if you do not want VTAM to go to the NetView alias application to determine the CoS name for the SSCP-SSCP session. If this operand is not coded, the adjacent CoS name defaults to ISTVTCOS, and the alias application, if active, is called with the default name.

ADJNETEL
ADJNETEL

Statements: GWPATH
Dependencies: ADJNET
Range: 1–32767

Specifies the element address of the CDRM for the adjacent network. Code ADJNETEL if the actual network of the CDRM is not the same as ADJNET and the default value of 1 is not correct.

Although it is not required, you should code ADJNETEL for all GWPATH definition statements. Its value should be the same as that defined on the ELEMENT operand of the GWNAU definition statement. If the GWNAU definition statement is not coded, ADJNETEL should be the same as the element number of the adjacent SSCP.

ADJNETSA

Statements: GWPATH
Dependencies: none
Range: 1–65535

Specifies the subarea address of the CDRM for the adjacent network. Code ADJNETSA for all GWPATH definition statements.

CAPACITY
CAPACITY = number of bits

Statements:

CDRM

Dependencies:

VRTG=YES coded; APPN only

Range:

I - 1000Kb

I - 1000Mb

I - 100Gb

Specifies the effective capacity of the link that comprises the transmission group (TG). Specify the value in Kb per second (for example, 100K), Mb per second (for example, 100M), or Gb per second (for example, 10G). This number approximates the bits per second that the link can transmit (the transmission rate of the link, times the maximum load factor expressed as a percentage).

Note: Because the value for capacity is represented as a single byte, the precision of the specified number of bits might be lost. For example, numbers that are close (such as 100K and 101K) can be interpreted by VTAM, and displayed, as the same value. See Table 45 on page 722 for a list of CAPACITY values you can specify and their corresponding values when displayed.

CDRDYN
Specifies whether the host CDRM is authorized to dynamically define CDRSC representations of cross-domain or cross-network resources when a session request is received from or sent to an adjacent SSCP or CP.

**CDRDYN=NO**
Causes the failure of any cross-domain or cross-network session request for a cross-domain or cross-network resource that has no predefined CDRSC entry.

**CDRDYN=YES**
Authorizes dynamic definition of cross-domain or cross-network resources by the host CDRM, eliminating the need to predefine certain CDRSCs. Both origin logical units and destination logical units can be dynamically defined. Note that for subarea nodes (HOSTSA start option is specified), the DYNLU=YES operand is meaningful only when CDRDY=YES.

**Note:**
1. To allow dynamic resource definition, code CDRDYN=YES on the CDRM definition statement for the VTAM host CDRM and CDRSC=OPT on the CDRM definition statement for the owning SSCP.
2. An intermediate host can also build dynamic definitions for CDRSCs.
3. The CDRDYN start option overrides the CDRDYN operand value. You can use the MODIFY VTAMOPTS command to change the value of CDRDYN.

**cdrmname**
CDRM

CDRM statements:

CDRM dependencies:

Should match the SSCPNAME start option of the CDRM being defined. Provides the name of the SSCP. It should match the name assigned to that SSCP in its own network by the SSCPNAME start option.

Specifies, in a multiple-domain network, the 1–8 character name of the cross-domain resource manager minor node of this network. For CDRMs in other networks, cdrmname is the name by which this VTAM identifies the other-network CDRMs. cdrmname is required for this network and other networks.

Because cdrmname is used in the ID operand of VTAM operator commands, it must be a name unique to this VTAM. As a result, the name fields of all CDRM definition statements that are active at one time must be unique and not conflict with other resource names in the same network. Because the CDRM definition statement in an adjacent SSCP table requires the SSCPNAME of an external CDRM when the CDRSC associated with the table has been coded with a NETWORK definition statement, you might have to change the labels on the CDRM definition statements.

To simplify system definition and the network operator's job, use the same name for the host CDRM and for the SSCPNAME start option if the NODETYPE start option is specified. If NODETYPE is coded in the start list and the names do not match, the following actions are taken:

- The host CDRM name is ignored and the SSCPNAME is required.
- Messages are generated to inform the operator that the SSCPNAME is being used instead of the host CDRM name.

Note: CDRM sessions might be affected by the name change. If you do not use the same cdrmname, the session might fail in some cross-network configurations.
Determines whether resources owned by the CDRM being defined by this statement can be dynamically defined at the time a session request is received from or sent to the external CDRM that manages the resource. The value you specify determines whether dynamic CDRSC definitions are allowed when cross-domain or cross-network session requests are received from or sent to certain domains.

**Note:** CDRSC is used only for SSCP-SSCP sessions.

**CDRSC=OPT**

Authorizes dynamic definition of cross-domain or cross-network resources from this CDRM, eliminating the need to predefine certain CDRSCs.

To allow dynamic resource definition, code CDRSC=OPT on the CDRM definition statement for the owning SSCP and code CDRDYN=YES on the CDRM definition statement for the VTAM host CDRM.

**CDRSC=REQ**

Causes the failure of any cross-domain or cross-network request to or from this CDRM for a cross-domain resource that has no predefined CDRSC entry.

**CONFGDS**
CONFGDS = _confgds_name_

statements: VBUILD
dependencies: none

Specifies a 1–8 character data definition name that identifies the configuration restart data set defined by the user for this major node. Include a DD statement that has this data definition name in the VTAM start procedure.

CONFGPW = _confgds_password_

statements: VBUILD
dependencies: CONFGDS

Specifies the 1–7 character alphanumeric password that VTAM uses to access the configuration restart data set. If you do not code CONFGPW, but VSAM requires it, VSAM prompts the VTAM operator for the correct password when VTAM attempts to open the data set.

Code CONFGPW only if you code CONFGDS.

COSTBYTE
**COSTBYTE**

Specifies a cost-per-byte-transmitted to be associated with the transmission group. A value of 0 is the least expensive cost per byte and 255 is the most expensive.

**COSTTIME**

Specifies a cost-per-unit-of-time to be associated with the transmission group. A value of 0 is the least expensive cost per unit of time and 255 is the most expensive.

**DISJOINT**
Indicates whether a search from an SSCP should be sent to another SSCP, when the SSCP's have matching NETIDs and their only connection is through an APPN network. If the NETIDs match, the coding of the DISJOINT operand determines whether the search is sent.

In effect, coding DISJOINT=NO restricts APPN searches, especially broadcast searches, from entering the subarea network multiple times through different interchange nodes (or even through the same interchange node).

**DISJOINT=NO**
- Indicates that the APPN network should not send the search to the second SSCP.

**DISJOINT=YES**
- Indicates that the APPN network should send the search to the second SSCP.

**Note:** If the NETIDs of the SSCP's do not match, the search is always sent to the second SSCP, regardless of the coding of the DISJOINT operand.

**ELEMENT**
On CDRM:

Specifies the element part of the network address of the CDRM. This operand, with the SUBAREA value, defines the network address of the CDRM.

For the host CDRM, the ELEMENT value must be 1. For external CDRMs (CDRMs in other domains or networks), this number can be a decimal integer in the range 0–32767. The default value is 1 for both same- and cross-network CDRMs.

On GWPATH:

Specifies the element part of the alias address for the CDRM in the VTAM network. The alias address is an address in the associated gateway NCP. The gateway SSCP uses this value only if it does not have a session with the associated gateway NCP when it starts the CDRM for the other network.

\textit{element address} must be a decimal integer in the range 0–32767. If you use the default of 1, or if you code ELEMENT and SUBAREA, VTAM assigns the network address to the CDRM if it is active. The network address must be unique in the network. The element address must match the element address predefined in the GWNNAU definition statement for the CDRM.
GWN

statements: GWPATH
dependencies: PUNAME on BUILD statement

Provides the 1–8 character name of the gateway NCP that accesses the CDRM. If you code PUNAME in the NCP BUILD definition statement, `gateway_ncp_name` must match the PUNAME. If you do not code PUNAME, `gateway_ncp_name` is the name used to file the gateway NCP definition statements.

If you do not code this operand, you must code the SUBAREA operand, because VTAM uses the SUBAREA operand value to determine which gateway NCP it uses to access the CDRM. It is to your advantage to omit GWN if you use different node names, but the same subarea number, for a gateway NCP. (For example, the NCP might have one name in a test system and another name in a production system.) In that case, VTAM uses the subarea number when it accesses the CDRM, without regard to the node name.

GWSELECT

statements: CDRM
dependencies: cross-network CDRMs

Defines whether the host CDRM is to perform gateway path selection for LU-LU sessions flowing to or through this cross-network external CDRM.

GWSELECT=NO
Prevents the host CDRM from performing gateway path selection for an LU-LU session flowing to or through this cross-network external CDRM.

GWSELECT=YES
Authorizes the host CDRM to perform gateway path selection for an LU-LU session flowing to or through this cross-network external CDRM.

Note: Code GWSELECT only for cross-network CDRMs. It is ignored for all others.

HPR
Specifies whether High-Performance Routing (HPR) support is enabled over the VR-based TG to a CDRM. This operand is applicable only to external CDRM definitions; it is not applicable to host CDRM definitions.

**HPR=YES**
Indicates that HPR support is provided over the VR-based TG to this CDRM.

**HPR=NO**
Indicates that HPR support is not provided over the VR-based TG to this CDRM.

**Note:** HPR defaults to NO when the HPR start option is specified as (RTP,NONE) or (ANR,NONE). If the HPR start option is specified as NONE, this operand is ignored.

**ISTATUS**

Specifies the initial status of the host or external CDRM.

**For the host CDRM:**

**ISTATUS=ACTIVE**
Specifies that it can establish a session with an external CDRM.

**ISTATUS=INACTIVE**
Specifies that it cannot establish a session with an external CDRM until it is started.

**For the external CDRM:**

**ISTATUS=ACTIVE**
Specifies that a session is attempted between the host in which this statement is being defined and the CDRM named by this definition statement. Any attempt to establish this session fails if either the host CDRM or the external CDRM is not active. Code ISTATUS=ACTIVE only if:

- This CDRM is in the same major node as the host CDRM (for which ISTATUS=ACTIVE is coded), or
- This CDRM major node is to be started after the host CDRM is active.
Specifies that a session between CDRMs is not automatically attempted when the CDRM major node is started.

After its major node has been started, a CDRM defined as INACTIVE can be started with the VARY ACT command, or for a CDRM in another domain, by receiving a session-activation request from the domain.

**name**

---

**statements:** VBUILD, NETWORK, GWPATH
**dependencies:** none

---

**On VBUILD:**
 Specifies the optional name of this major node.

If a syntax or definition error is detected with this statement during activation of the major node, *name* appears in the message issued to identify the error.

The optional *name* you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node’s status.

**On NETWORK:**
 Identifies this NETWORK definition statement in error messages from the VTAM system definition processor. To avoid confusion, you should code the same value for both *name* and the NETID operand. *name* is optional, but its use is highly recommended.

**On GWPATH:**
 Identifies this GWPATH definition statement in error messages from the VTAM system definition processor. *name* is optional, but its use is highly recommended.

**NETID**
**NETID**

```plaintext
NETID = network_id
```

**statements:** NETWORK  
**dependencies:** none

Specifies a 1–8 character identifier for the network in which the CDRM or set of CDRMs reside. This NETID pertains to the CDRM or set of CDRMs defined after the NETWORK definition statement and before any other NETWORK definition statement in the same set of definitions.

NETID is required only on external CDRMs if GWSSCP=YES; otherwise, NETID is optional.

**Note:** The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

**NN**

```plaintext
NN
```

**statements:** CDRM  
**dependencies:** VRTG=YES coded; APPN only

Specifies that the virtual-route-based transmission group represents a connection to an interchange node or a migration data host. When this parameter is specified, the network capabilities specified by the adjacent CP during VR-based transmission group activation are validated against the coded value. If the indicators do not match, both the VR-based transmission group and SSCP-SSCP session will be terminated.

When this operand is not specified, no network capabilities are validated.

**NN=YES**

- Indicates that the adjacent node must be an interchange network node.

**NN=NO**

- Indicates that the adjacent node must be a migration data host.

**Restriction:** In a peripheral subnetwork boundary configuration between a border node and a network node without border node function, the border node appears to the non-border node as an end node. In this case, coding NN=YES on the definition statement that represents the border node, a connection failure will occur. Avoid the failure by not coding the NN operand or by coding NN=NO.

**PDELAY**
PDELAY = NEGLIGIB

1. NEGLIGIB is the default if the TRLE operand is coded.

Notes:
1. NEGLIGIB is the default if the TRLE operand is coded.

Specifications: 
- CDRM
- VRTG=YES coded; APPN only

Specifies the maximum propagation delay of the link for the transmission group. Propagation delay represents the time needed for a signal to travel from one end of the link to the other.

PDELAY=NEGLIGIB
- Indicates a local area network delay (less than .48 milliseconds).

PDELAY=TERRESTR
- Indicates telephone network delay (between .48 and 49.152 milliseconds).

PDELAY=PACKET
- Indicates a packet-switched network delay (between 49.152 and 245.76 milliseconds).

PDELAY=LONG
- Indicates a satellite delay (greater than 245.76 milliseconds).

RECOVERY
Determines whether the SSCP-SSCP session between the named CDRM and the host CDRM should be restarted automatically when a session fails.

Both the host CDRM and the external CDRM must have RECOVERY=YES to enable automatic recovery.

**RECOVERY=NO**
- Specifies that the SSCP-SSCP session should not be restarted automatically.

**RECOVERY=YES**
- Specifies that the SSCP-SSCP session should be restarted automatically.

### SECURITY

Specifies the security level of the transmission group. The following options range from the least secure level to the most secure level.

**SECURITY=UNSECURE**
- Specifies no security level.

**SECURITY=PUBLIC**
- Specifies a public switched network.

**SECURITY=UNDERGRO**
- Specifies an underground cable, not guarded.

**SECURITY=SECURE**
- Specifies a secure conduit, not guarded.

**SECURITY=GUARDED**
- Specifies a guarded conduit, physical only.

**SECURITY=ENCRYPT**
- Specifies link encryption.
**SECURITY=SHIELDED**
Specifies a guarded conduit, physical and radiation shielded.

**SPAN**

```
[SPAN=span_name]
```

statements: CDRM
dependencies: NetView

Defines a span of control for VTAM minor node resources. Code this operand if you are using the NetView program. For a full description, see *IBM Tivoli NetView for z/OS Installation: Getting Started*.

The NetView program checks the SPAN value, but VTAM ignores it.

**SUBAREA**

On CDRM

```
SUBAREA=suba_addr
```

statements: CDRM
dependency: none
range: 1–value of MXSUBNUM start option

On GWPATH

```
SUBAREA=suba_addr
```

statements: GWPATH
dependency: none
range: 1–value of MXSUBNUM start option

**On CDRM:**
Specifies the number of the subarea in which this CDRM resides.

The SUBAREA value for each subarea used in cross-domain or cross-network communications must be unique throughout the network, but not throughout a set of SNA interconnected networks. For a host CDRM, this SUBAREA value must be the same as the host’s subarea specified by the HOSTSA start option.

If no GWPATH definition statements follow the other-network CDRM definition statement, code the SUBAREA operand on that CDRM definition statement. The SUBAREA operand must represent a valid subarea number in this network for a gateway NCP.

GWPATH provides subarea information. It is not required for cross-domain communication, but is required for cross-network communication if the following conditions exist:
- CDLM is defined for the same network
- GWSSCP=YES in the START options and this CDLM is defined for cross-network sessions

**On GWPATH:**
Specifies the subarea part of the real address for the CDLM in the network of the gateway SSCP. The real address is an address in the associated gateway NCP.
For a gateway SSCP, code SUBAREA if you do not code GWN. You can use SUBAREA instead of GWN when the gateway NCP has different node names but always has the same subarea number. See the description of the GWN operand.

The gateway SSCP uses this value if you do not code GWN, or if it does not have a session with the associated gateway NCP when VTAM starts the other network’s CDRM.

**TGP**

\[
\text{TGP} = \text{tgp\_name}
\]

*statements:*

*dependencies:*

- CDRM
- VRTG=YES coded; APPN only

Specifies the name of a transmission group (TG) profile definition. The characteristics defined in the TG profile (along with any modifiers specified on this statement) become the characteristics of the VR-based transmission group. If *tgp\_name* is not specified or the specified name cannot be found, the default TG characteristics are assigned.

See “APPN transmission group profile” on page 720 for more information on transmission group profiles.

**TYPE**
TYPE = CDRM

Indicates a CDRM major node. This operand is required.

UPARM1

UPARM1 = 128

UPARM1 = user-defined_operand_value

Specifies a user-defined operand value.

See z/OS Communications Server: SNA Network Implementation Guide for an example of using the UPARM operands.

UPARM2

UPARM2 = 128

UPARM2 = user-defined_operand_value

Specifies a user-defined operand value.

UPARM3
UPARM3 = 128

UPARM3 = user-defined_operand_value

Specifies a user-defined operand value.

**VPACING**

VPACING = 63

VPACING = number_of_requests

Specifies the maximum number of requests that other CDRMs can send to this CDRM before waiting to receive a pacing response. This value is exchanged when a session is established between CDRMs. This operand is effective only when the CDRM being defined is the host's own CDRM. VPACING affects only SSCP-SSCP session traffic.

If you specify a value greater than 63, the maximum value of 63 is used. If VPACING=0, no pacing is done.

The value of VPACING can be expressed with leading zeros and is limited to eight digits.

For more information on pacing, see z/OS Communications Server: SNA Network Implementation Guide.

**VRTG**
VRTG = value_of_vrtg_start_option

VRTG = YES

NO

Indicates for this node, whether virtual-route-based transmission group connections are requested when SSCP-SSCP sessions are established. This operand is valid only for interchange nodes and migration data hosts (when start options NODETYPE and HOSTSA are specified).

NO Indicates that VR-based transmission group connection is not requested when SSCP-SSCP sessions are established.

YES Indicates that VR-based transmission group connection is requested when SSCP-SSCP sessions are established.

VRTGCPCP

VRTGCPCP = value_of_vrtgcpcp_start_option

VRTGCPCP = YES

NO

Indicates whether CP-CP sessions are supported over the virtual-route-based transmission group. VRTGCPCP is valid only for interchange nodes and migration data hosts (when start options NODETYPE and HOSTSA are specified).

NO Indicates that CP-CP sessions are not supported over the VR-based transmission group.

YES Indicates that CP-CP sessions are supported over the VR-based transmission group.

External communication adapter (XCA) major node

The external communication adapter (XCA) major node defines VTAM's connections to the following shared access transport facilities (SATFs):

- Local area networks (LANs)
- Asynchronous transfer mode (ATM) networks accessed through:
  - LAN emulation
  - Native ATM
- Enterprise Extender (EE) networks
LAN and ATM LAN emulation connections

VTAM attaches to LANs through an IBM 3172 Nways Interconnect Controller, an Open Systems Adapter, or through an IP network for Enterprise Extender. VTAM attaches to ATM networks through LAN emulation access through an Open Systems Adapter. VTAM can communicate with the following types of nodes over LAN and ATM LAN emulation connections:

- Subarea
  Subarea nodes are nodes with a PU type of 4 or 5. Connections to subarea nodes are non-switched, DIAL=NO.

- Peripheral
  Peripheral nodes are nodes with a PU type of 2 or 2.1. Connections to peripheral nodes are switched, DIAL=YES.

Connections to subarea nodes, DIAL=NO

To define a subarea external communication adapter major node, code the VBUILD and PORT definition statements, followed by the GROUP definition statement, and LINE and PU definition statements as pairs in the non-switched line group.

**VBUILD definition statement:**
See “VBUILD definition statement” on page 185 for details on the VBUILD definition statement.

**PORT definition statement:**
See “PORT definition statement” on page 185 for details on the PORT definition statement.

**GROUP definition statement:**
Code a GROUP definition statement to group the similar characteristics of peer subarea nodes (specified with DIAL=NO).

Code a separate GROUP definition statement to describe each set of characteristics.

The major node can contain different link groups, and each link group must have a unique GROUP definition statement and contain a unique set of minor node definition statements.

**LINE definition statement:**
Code a LINE definition statement to define a single line to a peer processor or station on a LAN attached through an IBM 3172 Nways interconnect controller or through an IP network for Enterprise Extender. If the device has more than one line connected, code a separate LINE definition statement for each line.

**PU definition statement:**
Code a PU definition statement to define the physical unit associated with a line if the line defines a SNA resource. USER=SNA on the LINE definition statement defines SNA resources. Code only one PU definition statement for each LINE definition statement.

**Note:** If USER=VCNS is specified on the LINE definition statement, you cannot code a PU definition statement.
### Table 17. External communication adapter major node–LAN or ATM LAN emulation connections, subarea node

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=XCA</td>
<td>“TYPE” on page 261</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFIGDS</td>
<td>“CONFIGDS” on page 211</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFGPW</td>
<td>“CONFGPW” on page 211</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PORT</td>
<td>ADAPNO</td>
<td>“ADAPNO” on page 203</td>
<td>R</td>
<td></td>
<td>MEDIUM</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CUADDR</td>
<td>“CUADDR” on page 210</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MEDIUM</td>
<td>“MEDIUM” on page 241</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DELAY</td>
<td>“DELAY” on page 216</td>
<td>R</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SAPADDR</td>
<td>“SAPADDR” on page 252</td>
<td></td>
<td></td>
<td>4 USER=SNA</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TIMER</td>
<td>“TIMER” on page 260</td>
<td></td>
<td></td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>GROUP</td>
<td>DIAL</td>
<td>“DIAL” on page 217</td>
<td>NO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>“SPAN” on page 255</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>name</td>
<td>LINE</td>
<td>ISTATUS</td>
<td>“ISTATUS” on page 234</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>“SPAN” on page 255</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USER</td>
<td>“USER” on page 265</td>
<td></td>
<td>SNA</td>
<td></td>
<td>DIAL=NO</td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td>ALLOWACT</td>
<td>“PU” on page 189</td>
<td>NO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISTATUS</td>
<td>“ISTATUS” on page 234</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LANTEST</td>
<td>“LANTEST” on page 235</td>
<td>LOCAL</td>
<td>See</td>
<td></td>
<td>“LANTEST” on page 235</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MACADDR</td>
<td>“MACADDR” on page 239</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MULTGSUP</td>
<td>“MULTGSUP” on page 244</td>
<td>YES</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PUTYPE</td>
<td>“PUTYPE” on page 250</td>
<td>G, L</td>
<td>5 USER=VCNS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SAPADDR</td>
<td>“SAPADDR” on page 252</td>
<td></td>
<td></td>
<td>4 USER=SNA</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>“SPAN” on page 255</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SUBAREA</td>
<td>“SUBAREA” on page 256</td>
<td>in_xid_value</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 17. External communication adapter major node–LAN or ATM LAN emulation connections, subarea node (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>TGN</td>
<td>&quot;TGN&quot; on page 258</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XNETALS</td>
<td>&quot;XNETALS&quot; on page 272</td>
<td></td>
<td></td>
<td></td>
<td>XNETALS start option value</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:
- **R** Required operand.
- **G** Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.
- **L** Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code it on this definition statement.
- **P** Code this operand on the higher-level PU definition statement to take advantage of the sift effect, or code it on this definition statement.

Connections to peripheral nodes, DIAL=YES

To define a peripheral external communication adapter node, code the VBUILD and PORT definition statements followed by the GROUP definition statement, and LINE and PU definition statements as pairs in the switched line group.

**VBUILD definition statement:**
See "VBUILD definition statement" on page 185 for details on the VBUILD definition statement.

**PORT definition statement:**
See "PORT definition statement" on page 185 for details on the PORT definition statement.

**GROUP definition statement:**
Code a GROUP definition statement to define a line group for type 2 or 2.1 nodes.

The major node can contain different link groups, and each link group must have a unique GROUP definition statement and contain a unique set of minor node definition statements.

**LINE definition statement:**
Code a LINE definition statement for each switched line. The LINE definition statement defines to VTAM the name of the line and certain procedural options to be used for this line.

**PU definition statement:**
Code a PU definition statement for each peripheral node with which VTAM will communicate over this switched link.
### Table 18. External communication adapter major node—LAN, ATM LAN emulation, or Enterprise Extender connections, peripheral node

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=XCA</td>
<td>261</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CONFIGS</td>
<td></td>
<td>211</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CONFIGPW</td>
<td></td>
<td>211</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VTAMTOPO</td>
<td></td>
<td>270</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PORT</td>
<td>ADAPNO</td>
<td>203</td>
<td>R</td>
<td></td>
<td>MEDIUM</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CUADDR</td>
<td></td>
<td>216</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CAPACITY</td>
<td></td>
<td>209</td>
<td></td>
<td></td>
<td>4M³</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8M⁴</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COSTBYTE</td>
<td></td>
<td>212</td>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>APPN only; VNNNAME; VNGROUP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COSTTIME</td>
<td></td>
<td>213</td>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>APPN only; VNNNAME; VNGROUP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HPREELIV</td>
<td></td>
<td>229</td>
<td></td>
<td>YES</td>
<td>MEDIUM=HPRIP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IPRESOLV</td>
<td></td>
<td>233</td>
<td></td>
<td>0</td>
<td>MEDIUM=HPRIP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IPTOS</td>
<td></td>
<td>234</td>
<td></td>
<td>20,40,80,C0,C0</td>
<td>MEDIUM=HPRIP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LIVTIME</td>
<td></td>
<td>237</td>
<td></td>
<td>10</td>
<td>MEDIUM=HPRIP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAXSLOW²</td>
<td></td>
<td>240</td>
<td></td>
<td>0</td>
<td>(0,180)</td>
<td>CUADDR</td>
</tr>
<tr>
<td></td>
<td>MEDIUM</td>
<td></td>
<td>241</td>
<td></td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PDELAY</td>
<td></td>
<td>248</td>
<td></td>
<td></td>
<td>TERRESTR⁵</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>APPN only; VNNNAME; VNGROUP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ROUTEDIR</td>
<td></td>
<td>252</td>
<td></td>
<td>INBOUND</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SAPADDR</td>
<td></td>
<td>252</td>
<td></td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SRQRETRY</td>
<td></td>
<td>255</td>
<td></td>
<td>3</td>
<td>MEDIUM=HPRIP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SRQTIME</td>
<td></td>
<td>256</td>
<td></td>
<td>15</td>
<td>MEDIUM=HPRIP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SECURITY</td>
<td></td>
<td>254</td>
<td></td>
<td>UNSECURE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>APPN only; VNNNAME; VNGROUP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TGP</td>
<td></td>
<td>259</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TIMER</td>
<td></td>
<td>260</td>
<td></td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>UNRCHTIM</td>
<td></td>
<td>266</td>
<td></td>
<td>UNRCHTIM</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>start option value</td>
<td>MEDIUM=HPRIP; VNNNAME or VNTYPE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UPARM1</td>
<td></td>
<td>263</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>APPN only; VNNNAME; VNGROUP</td>
<td></td>
</tr>
</tbody>
</table>
Table 18. External communication adapter major node—LAN, ATM LAN emulation, or Enterprise Extender connections, peripheral node (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>UPARM2</td>
<td></td>
<td></td>
<td>264</td>
<td></td>
<td></td>
<td>128</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>APPN only; VNNNAME; VNGROUP</td>
</tr>
<tr>
<td>UPARM3</td>
<td></td>
<td></td>
<td>265</td>
<td></td>
<td></td>
<td>128</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>APPN only; VNNNAME; VNGROUP</td>
</tr>
<tr>
<td>VNGROUP</td>
<td></td>
<td></td>
<td>267</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>VNNNAME; DIAL=YES</td>
</tr>
<tr>
<td>VNNAME</td>
<td></td>
<td></td>
<td>268</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>VNGROUP; DIAL=YES</td>
</tr>
<tr>
<td>VNTYPE</td>
<td></td>
<td></td>
<td>269</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>GROUP</td>
<td>DIAL</td>
<td>217</td>
<td>R</td>
<td>NO</td>
<td>DIAL=YES</td>
<td></td>
</tr>
<tr>
<td>AUTOGEN</td>
<td></td>
<td></td>
<td>206</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAPACITY</td>
<td></td>
<td></td>
<td>209</td>
<td></td>
<td></td>
<td>4M</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNNNAME or VNTYPE</td>
</tr>
<tr>
<td>COSTBYTE</td>
<td></td>
<td></td>
<td>212</td>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNNNAME or VNTYPE</td>
</tr>
<tr>
<td>COSTTIME</td>
<td></td>
<td></td>
<td>213</td>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNNNAME or VNTYPE</td>
</tr>
<tr>
<td>DYNPU</td>
<td></td>
<td></td>
<td>223</td>
<td></td>
<td></td>
<td>NO</td>
<td>DYNPU=YES</td>
</tr>
<tr>
<td>DYNPUPFX</td>
<td></td>
<td></td>
<td>224</td>
<td></td>
<td></td>
<td>CN</td>
<td>DYNPU=YES</td>
</tr>
<tr>
<td>DYNVNPFX</td>
<td></td>
<td></td>
<td>225</td>
<td></td>
<td></td>
<td>DYNVPFX start option value</td>
<td>VNGROUP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EEVERIFY</td>
<td></td>
<td></td>
<td>226</td>
<td></td>
<td></td>
<td>EEVERIFY start option value</td>
<td>MEDIUM=HPRIP; VNNNAME or VNTYPE</td>
</tr>
<tr>
<td>HOSTNAME</td>
<td></td>
<td></td>
<td>227</td>
<td></td>
<td></td>
<td>HOSTNAME start option value²</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HPREEARB</td>
<td></td>
<td></td>
<td>229</td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNNNAME or VNTYPE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HPRPSDLY</td>
<td></td>
<td></td>
<td>230</td>
<td></td>
<td></td>
<td>HPRPSDLY start option value</td>
<td>MEDIUM=HPRIP; VNNNAME or VNTYPE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPADDR</td>
<td></td>
<td></td>
<td>231</td>
<td></td>
<td></td>
<td>IPADDR start option value²</td>
<td>MEDIUM=HPRIP²</td>
</tr>
<tr>
<td>LIVTIME</td>
<td></td>
<td></td>
<td>237</td>
<td>P</td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP</td>
</tr>
</tbody>
</table>

Chapter 2. Major nodes  177
### Table 18. External communication adapter major node—LAN, ATM LAN emulation, or Enterprise Extender connections, peripheral node (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTU</td>
<td></td>
<td></td>
<td>243</td>
<td></td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>PDELAY</td>
<td></td>
<td></td>
<td>248</td>
<td></td>
<td>TERRESTR</td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>SECURITY</td>
<td></td>
<td></td>
<td>254</td>
<td></td>
<td>UNSECURE</td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>SRQRETRY</td>
<td></td>
<td></td>
<td>255</td>
<td>P</td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>SRQTIME</td>
<td></td>
<td></td>
<td>256</td>
<td>P</td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>SPAN</td>
<td></td>
<td></td>
<td>255</td>
<td>P</td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>TGP</td>
<td></td>
<td></td>
<td>259</td>
<td></td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>UNRCHTIM</td>
<td></td>
<td></td>
<td>266</td>
<td></td>
<td>UNRCHTIM</td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>UPARM1</td>
<td></td>
<td></td>
<td>263</td>
<td></td>
<td></td>
<td>128</td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>UPARM2</td>
<td></td>
<td></td>
<td>264</td>
<td></td>
<td></td>
<td>128</td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>UPARM3</td>
<td></td>
<td></td>
<td>265</td>
<td></td>
<td></td>
<td>128</td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>VNNAME</td>
<td></td>
<td></td>
<td>268</td>
<td></td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>VNTYPE</td>
<td></td>
<td></td>
<td>269</td>
<td></td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>VTAMTOPO</td>
<td></td>
<td></td>
<td>270</td>
<td></td>
<td></td>
<td></td>
<td>MEDIUM=HPRIP; VNAME or VNTYPE</td>
</tr>
<tr>
<td>name</td>
<td>LINE</td>
<td></td>
<td>204</td>
<td>G</td>
<td></td>
<td>ON</td>
<td>CALL=OUT</td>
</tr>
<tr>
<td></td>
<td>ANSWER</td>
<td></td>
<td>208</td>
<td>G</td>
<td></td>
<td>IN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALL</td>
<td></td>
<td>208</td>
<td>G</td>
<td></td>
<td>IN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ISTATUS</td>
<td></td>
<td>234</td>
<td>G</td>
<td></td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KEEPACT</td>
<td></td>
<td>235</td>
<td></td>
<td></td>
<td>YES</td>
<td>Enterprise Extender only</td>
</tr>
<tr>
<td></td>
<td>LIMRES</td>
<td></td>
<td>236</td>
<td>G</td>
<td></td>
<td>NO</td>
<td>LIMQSINT; LU 6.2; PUTYPE=1, 2</td>
</tr>
<tr>
<td></td>
<td>SPAN</td>
<td></td>
<td>255</td>
<td>G</td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td></td>
<td>SWNORDER</td>
<td></td>
<td>257</td>
<td>G</td>
<td></td>
<td>CPNAME, FIRST</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VTAMTOPO</td>
<td></td>
<td>270</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td></td>
<td>234</td>
<td>G</td>
<td></td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ISTATUS</td>
<td></td>
<td>234</td>
<td>G</td>
<td></td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SPAN</td>
<td></td>
<td>255</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
</tbody>
</table>
Table 18. External communication adapter major node—LAN, ATM LAN emulation, or Enterprise Extender connections, peripheral node (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>XNETALS</td>
<td></td>
<td></td>
<td>272</td>
<td></td>
<td></td>
<td>XNETALS start option value</td>
<td></td>
</tr>
</tbody>
</table>

Legend:

- **R** Required operand.
- **G** Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.
- **L** Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code it on this definition statement.
- **P** Code this operand on the higher-level PORT definition statement to take advantage of the sift effect, or code it on this definition statement.

**Note:**
1. For LAN or ATM emulation connections only.
2. If both HOSTNAME and IPADDR are coded on the GROUP definition statement, the value for IPADDR is ignored.
3. If connection is a virtual node.
4. For connections other than virtual nodes.
5. For an Enterprise Extender connection.
6. For connections other than Enterprise Extender connections.
7. The LIVTIME, SRQTIME, and SRQRETRY operands must be unique for each local static VIPA. For example, if multiple GROUP definition statements are associated with the same local static VIPA (IPADDR or HOSTNAME), and these statements specify values or use default values that are different, the group that is activated first sets the values that are used. After the first group is activated, the other groups receive error messages indicating that their LDLC timer parameters are being ignored; the LDLC values previously set are also used for these EE connections.

**VBUILD definition statement**
Code a VBUILD definition statement for the Enterprise Extender XCA major node.

**PORT definition statement**
Code a PORT definition statement for each VBUILD TYPE=XCA definition statement. The PORT definition statement defines the VTAM connection to the Enterprise Extender shared access transport facility (SATF). Note that MEDIUM=HPRIP is required. However, only one XCA Major Node for Enterprise Extender (MEDIUM=HPRIP) can be active at a time.

VNNAME specifies the name of the virtual node representing the SATF and VNGROUP specifies the name of the DIAL=YES group for connections through the connection network. VNTYPE specifies if an Enterprise Extender connection network is permitted to span APPN network boundaries.

See z/OS Communications Server: SNA Network Implementation Guide for an example of an Enterprise Extender XCA major node using Connection Network.

SAPADDR=address is optional and specifies the local SAP value; the default is 4.
GROUP definition statement
Code a GROUP definition statement to group the similar characteristics of peer nodes in the IP network. VNNAME and VNTYPE can be used to define a virtual node. Code a separate GROUP definition statement to describe each set of characteristics.

See z/OS Communications Server: SNA Network Implementation Guide for an example of an Enterprise Extender XCA major node using Connection Network.

The major node can contain different link groups and each link group must have a unique GROUP definition statement and contain a unique set of minor node definition statements.

Note: DIAL=YES is required for Enterprise Extender.

LINE definition statement
Code a LINE definition statement to define a single line to a peer node in the IP network. The number of LINE statements should equal the maximum number of simultaneous connections.

PU definition statement
Code a PU definition statement for each peer node in the IP network with which VTAM will communicate over this switched link.

ATM native connections
VTAM attaches to ATM networks through native ATM access through an Open Systems Adapter. VTAM can communicate with HPR APPN nodes over ATM native connections. Communication occurs over the following types of connections:
- Transmission groups (TGs) over permanent virtual channels (PVCs), DIAL=NO
- TGs over switched virtual channels (SVCs), DIAL=YES
- TGs to connection networks, DIAL=YES

To define an external communication adapter major node for an ATM native connection, code the VBUILD and PORT definition statements followed by GROUP, LINE, and PU definition statements to define the TGs.

VBUILD definition statement
See “VBUILD definition statement” on page 185 for details on the VBUILD definition statement.

PORT definition statement
See “PORT definition statement” on page 185 for details on the PORT definition statement.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=XCA</td>
<td>261</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFIGDS</td>
<td>211</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFGPW</td>
<td>211</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>VTAMTOPO</td>
<td>270</td>
<td></td>
<td>REPORT</td>
<td></td>
<td>CONFIGDS</td>
</tr>
<tr>
<td>name</td>
<td>PORT</td>
<td>PORTNAME</td>
<td>249</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 19. External communication adapter major node—ATM native connection, port (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEDIUM</td>
<td></td>
<td></td>
<td>241</td>
<td></td>
<td>ATM</td>
<td></td>
<td>PORTNAME</td>
</tr>
<tr>
<td>SAPADDR</td>
<td></td>
<td></td>
<td>252</td>
<td></td>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

- **R** Required operand.
- **G** Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.
- **L** Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code it on this definition statement.

### Transmission groups (TGs) over permanent virtual channels (PVCs), DIAL=NO

To define TGs over PVCs, code a GROUP definition statement, followed by one LINE and one PU definition statement, for each PVC to which you subscribe through the port defined by this XCA major node.

**GROUP definition statement:**

Code a GROUP definition statement to define a line group for remote nodes connected through the port defined by this XCA major node.

**LINE definition statement:**

Code a LINE definition statement for each PVC to which you subscribe through the port defined by this XCA major node. The LINE definition statement defines to VTAM the name of the PVC and certain procedural options to be used for the PVC.

**PU definition statement:**

Code a PU definition statement for a remote node with which VTAM can communicate over a PVC. The PU definition statement should follow the LINE definition statement that defines the PVC through which the remote node is connected.

### Table 20. External communication adapter major node—ATM native connection, TGs over PVCs

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>GROUP</td>
<td>DIAL=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>non-switched lines</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>name</td>
<td>LINE</td>
<td>PVCNAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISTATUS</td>
<td></td>
<td></td>
<td></td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USER</td>
<td></td>
<td></td>
<td></td>
<td>SNA</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td>AUTHLEN</td>
<td></td>
<td>YES</td>
<td></td>
<td>PU type 2.1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CAPACITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONNTYPE</td>
<td></td>
<td>APPN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE</td>
<td></td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 20. External communication adapter major node—ATM native connection, TGs over PVCs (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>COSTTIME</td>
<td></td>
<td></td>
<td>213</td>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>CPCP</td>
<td></td>
<td></td>
<td>214</td>
<td></td>
<td></td>
<td>CPCP start option</td>
<td></td>
</tr>
<tr>
<td>CPNAME</td>
<td></td>
<td></td>
<td>215</td>
<td></td>
<td></td>
<td>CPNAME start option</td>
<td></td>
</tr>
<tr>
<td>DYNADJCP</td>
<td></td>
<td></td>
<td>221</td>
<td></td>
<td></td>
<td>DYNADJCP start option</td>
<td></td>
</tr>
<tr>
<td>DYNLU</td>
<td></td>
<td></td>
<td>222</td>
<td></td>
<td></td>
<td>DYNLU start option</td>
<td></td>
</tr>
<tr>
<td>HPR</td>
<td></td>
<td></td>
<td>223</td>
<td></td>
<td></td>
<td>YES</td>
<td></td>
</tr>
<tr>
<td>ISTATUS</td>
<td></td>
<td></td>
<td>234</td>
<td>G</td>
<td></td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td>NATIVE</td>
<td></td>
<td></td>
<td>245</td>
<td></td>
<td></td>
<td>BN=YES</td>
<td></td>
</tr>
<tr>
<td>NETID</td>
<td></td>
<td></td>
<td>246</td>
<td></td>
<td></td>
<td>XNETALS</td>
<td></td>
</tr>
<tr>
<td>NN</td>
<td></td>
<td></td>
<td>247</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDELAY</td>
<td></td>
<td></td>
<td>248</td>
<td></td>
<td></td>
<td>NEGLIGIB</td>
<td></td>
</tr>
<tr>
<td>PUTYPE</td>
<td></td>
<td></td>
<td>250</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>SAPADDR</td>
<td></td>
<td></td>
<td>252</td>
<td></td>
<td></td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>SECURITY</td>
<td></td>
<td></td>
<td>254</td>
<td></td>
<td></td>
<td>UNSECURE</td>
<td></td>
</tr>
<tr>
<td>SPAN</td>
<td></td>
<td></td>
<td>255</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>TGN</td>
<td></td>
<td></td>
<td>256</td>
<td></td>
<td></td>
<td>ANY</td>
<td></td>
</tr>
<tr>
<td>TGP</td>
<td></td>
<td></td>
<td>259</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOPO</td>
<td></td>
<td></td>
<td>261</td>
<td></td>
<td></td>
<td>CONNECT</td>
<td></td>
</tr>
<tr>
<td>UPARM1</td>
<td></td>
<td></td>
<td>263</td>
<td></td>
<td></td>
<td>128</td>
<td></td>
</tr>
<tr>
<td>UPARM2</td>
<td></td>
<td></td>
<td>264</td>
<td></td>
<td></td>
<td>128</td>
<td></td>
</tr>
<tr>
<td>UPARM3</td>
<td></td>
<td></td>
<td>265</td>
<td></td>
<td></td>
<td>128</td>
<td></td>
</tr>
<tr>
<td>VERALSID</td>
<td></td>
<td></td>
<td>266</td>
<td></td>
<td></td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>XID</td>
<td></td>
<td></td>
<td>272</td>
<td></td>
<td></td>
<td>YES</td>
<td></td>
</tr>
<tr>
<td>XNETALS</td>
<td></td>
<td></td>
<td>272</td>
<td></td>
<td></td>
<td>XNETALS start option</td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R  Required operand.

G  Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.

L  Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code it on this definition statement.

Transmission groups (TGs) over switched virtual channels (SVCs), DIAL=YES

To define TGs over SVCs, code a GROUP definition statement, followed by LINE and PU definition statements for the connections you expect to have over the SVCs through the port defined by this XCA major node. The LINE definition statements serve as placeholders for lines used to connect to remote nodes that are defined either dynamically, or in a switched major node, and are used when those nodes are activated.

GROUP definition statement:

Code a GROUP definition statement to define a line group for remote nodes connected through the port defined by this XCA major node.
LINE definition statement:
Code LINE definition statements for the connections you expect to have over the SVCs through the port defined by this XCA major node. The total number of LINE definition statements should equal the maximum number of simultaneous connections you can have. The LINE definition statement defines to VTAM certain procedural options to be used for an SVC assigned to this line.

PU definition statement:
Code a PU definition statement for a remote node with which VTAM can communicate over the SVCs through the port defined by this XCA major node.

Table 21. External communication adapter major node—ATM native connection, TGs over SVCs

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>GROUP</td>
<td>DIAL=YES</td>
<td>217</td>
<td>R</td>
<td>NO</td>
<td>switched lines</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AUTOGEN</td>
<td></td>
<td>206</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DYNPU</td>
<td></td>
<td>222</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DYNPUFX</td>
<td></td>
<td>224</td>
<td></td>
<td>CN</td>
<td>DYNPU=YES</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SPAN</td>
<td></td>
<td>255</td>
<td></td>
<td></td>
<td>NetView</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VTAMTOPO</td>
<td></td>
<td>270</td>
<td></td>
<td></td>
<td>DIAL=YES</td>
<td></td>
</tr>
</tbody>
</table>

name LINE

<table>
<thead>
<tr>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANSWER</td>
<td></td>
<td>204</td>
<td>G</td>
<td>ON</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CALL</td>
<td></td>
<td>208</td>
<td>G</td>
<td>IN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISTATUS</td>
<td></td>
<td>234</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LIMRES</td>
<td></td>
<td>236</td>
<td>G</td>
<td>NO</td>
<td>LU 6.2</td>
<td>NetView</td>
</tr>
<tr>
<td>SPAN</td>
<td></td>
<td>255</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWNORDER</td>
<td></td>
<td>257</td>
<td>G</td>
<td>CPNAME, FIRST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VTAMTOPO</td>
<td></td>
<td>270</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

name PU

<table>
<thead>
<tr>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTATUS</td>
<td></td>
<td>234</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>SPAN</td>
<td></td>
<td>255</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

- **R** Required operand.
- **G** Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.
- **L** Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code it on this definition statement.

Transmission groups (TGs) to connection networks, DIAL=YES
To define a TG to a connection network, code a GROUP definition statement, followed by LINE and PU definition statements for the connections you expect to have using the connection network TG through the port defined by this XCA major node. The GROUP definition statement, particularly through the connection network name (vnname) of the DLCADDR operand, defines the signaling information used to establish switched connections with other remote nodes in the same connection network. The LINE definition statements serve as placeholders for SVCs to remote nodes that are defined either dynamically, or in a switched major node, and are used when those nodes are activated.

GROUP definition statement:
Code a GROUP definition statement to define a TG to a connection network through the port defined by this XCA major node.

**LINE definition statement:**
Code LINE definition statements for the connections you expect to have over the connection network TG through the port defined by this XCA major node. The total number of LINE definition statements should equal the maximum number of simultaneous connections you can have using the connection network TG. The LINE definition statement defines to VTAM certain procedural options to be used for a connection network TG assigned to this line.

**PU definition statement:**
Code a PU definition statement for a remote node with which VTAM can communicate over the connection network TG through the port defined by this XCA major node.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>GROUP</td>
<td>DIAL=YES</td>
<td>217</td>
<td>R</td>
<td>NO</td>
<td>switched lines</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AUTGEN</td>
<td></td>
<td>206</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CAPACITY</td>
<td></td>
<td>209</td>
<td></td>
<td>25M</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>COSTBYTE</td>
<td></td>
<td>212</td>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>COSTTIME</td>
<td></td>
<td>213</td>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DLCADDR</td>
<td></td>
<td>217</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DYNPU</td>
<td></td>
<td>223</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DYNPUFX</td>
<td></td>
<td>224</td>
<td></td>
<td>CN</td>
<td></td>
<td>DYNPU=YES</td>
</tr>
<tr>
<td></td>
<td>DYNVNPFX</td>
<td></td>
<td>225</td>
<td></td>
<td>DYNVNPFX</td>
<td></td>
<td>VNGROUP start option value</td>
</tr>
<tr>
<td></td>
<td>PDELAY</td>
<td></td>
<td>248</td>
<td></td>
<td>NEGLIGLB</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SECURITY</td>
<td></td>
<td>254</td>
<td></td>
<td>UNSECURE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SPAN</td>
<td></td>
<td>255</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td></td>
<td>TGP</td>
<td></td>
<td>259</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>UPARM1</td>
<td></td>
<td>263</td>
<td>128</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>UPARM2</td>
<td></td>
<td>264</td>
<td>128</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>UPARM3</td>
<td></td>
<td>265</td>
<td>128</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VTAMTOPO</td>
<td></td>
<td>270</td>
<td></td>
<td></td>
<td></td>
<td>DIAL=YES</td>
</tr>
<tr>
<td>name</td>
<td>LINE</td>
<td>ANSWER</td>
<td>204</td>
<td>G</td>
<td>ON</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALL</td>
<td></td>
<td>206</td>
<td>G</td>
<td>IN</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ISTATUS</td>
<td></td>
<td>234</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LIMRES</td>
<td></td>
<td>236</td>
<td>G</td>
<td>NO</td>
<td></td>
<td>LU 6.2</td>
</tr>
<tr>
<td></td>
<td>SPAN</td>
<td></td>
<td>255</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td></td>
<td>SWNORDER</td>
<td></td>
<td>257</td>
<td>G</td>
<td>CPNAME, FIRST</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VTAMTOPO</td>
<td></td>
<td>270</td>
<td></td>
<td></td>
<td></td>
<td>DIAL=YES</td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td>ISTATUS</td>
<td>234</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td></td>
<td>SPAN</td>
<td></td>
<td>255</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td>NetView</td>
</tr>
</tbody>
</table>
Table 22. External communication adapter major node–ATM native connection, TGs to connection networks (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
</table>

Legend:
- **R** Required operand.
- **G** Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.
- **L** Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code it on this definition statement.

### External communication adapter (XCA) major node coding notes

This information contains coding notes for the external communication adapter major node.

#### VBUILD definition statement

Code a VBUILD definition statement for each XCA major node.

#### PORT definition statement

Code one PORT definition statement for each VBUILD TYPE=XCA definition statement. The PORT definition statement defines the VTAM connection to a shared access transport facility, such as:

- A CSMA/CD 802.3 LAN, a token-ring network, or a FDDI LAN attached through an IBM 3172 Nways Interconnect Controller, an Open Systems Adapter, or through an IP network for Enterprise Extender.
- An ATM network attached through an Open Systems Adapter through LAN emulation access.

**Note:** ATM networks attached through an Open Systems Adapter through LAN emulation access are defined to VTAM as though they are CSMA/CD 802.3 LANs or token-ring networks.

- An ATM network attached through an Open Systems Adapter through native ATM access.

The PORT definition statement also defines the actual adapter on the IBM 3172 Nways Interconnect Controller, Open Systems Adapter, or through an IP network for Enterprise Extender through which the connection is made.

### Dynamic change of the operands for an XCA major node for EE

You can dynamically change the following Enterprise Extender XCA major node operands by editing the VTAMLST member and then issuing the VARY ACT,UPDATE=ALL command. You can specify the new operand value on a higher-level definition statement for sifting, if applicable:

- ANSWER
- AUTOGEN
- CALL
- CAPACITY
To dynamically change an operand on the LINE statement, the resource to which it applies must be inactive. To dynamically change an operand on the GROUP statement, the resource to which it applies must be inactive and all LINE subnodes must be inactive.

**External communication adapter (XCA) major node full syntax**

The full syntax for the external communication adapter (XCA) major node follows. The syntax for each operand is repeated in "External communication adapter (XCA) major node operand descriptions" on page 203.

**LAN and ATM LAN emulation connections**

VBUILD:
name—VBUILD—TYPE—XCA

 CONFIGDS = configds_name
 CONFIGPW = configds_password

 VTAMTOPO = REPORT
 VTAMTOPO = NOREPORT
 VTAMTOPO = NOLLINES
 VTAMTOPO = NOSWPUS
 VTAMTOPO = INCLUDE
 VTAMTOPO = IGNORE

 PORT:

 name—PORT—ADAPNO = adapter_number—CUADDR = device_address

 CAPACITY = 4M
 CAPACITY = number_of_bits
 COSTBYTE = 0
 COSTBYTE = cost_per_byte
 COSTTIME = 0
 COSTTIME = cost_per_unit_time
 DELAY = 0
 DELAY = timeout_value
 MAXSLOW = (0,180)
 MAXSLOW = (value1, value2)
 MEDIUM = BOXMGR
 MEDIUM = CSMACD
 MEDIUM = FDDI
 MEDIUM = RING
 PDELAY = TERRESTR
 PDELAY = LONG
 PDELAY = NEGIGIBLE
 PDELAY = PACKET
 ROUTEDIR = INBOUND
 ROUTEDIR = OUTBOUND
 SAPADDR = 4
 SAPADDR = address
 SECURITY = UNSECURE
 SECURITY = PUBLIC
 SECURITY = UNDERGRO
 SECURITY = SECURED
 SECURITY = GUARDED
 SECURITY = ENCRYPT
 SECURITY = SHIELD

 TGP = tgp_name

 Chapter 2. Major nodes 187
GROUP:

- name

  AUTOGEN
  - num_stmts
  - (num_stmts, line_seed_char, pu_seed_char)

  DIAL
  - NO
  - YES

  DYNPU
  - NO
  - YES

  DYNPUPFX
  - "CN"
  - "string"

  DYNVNPFX
  - "value_of_dynvnpfx_start_option"
  - "string"

  ISTATUS
  - "ACTIVE"
  - "INACTIVE"

  SPAN
  - span_name

  VTAMTOPO
  - "REPORT"
  - "NOREPORT"
  - "NOLLINES"
  - "NOSWPUS"

LINE:
Notes:
1  LIMRES=NO is the only valid value if the TRLE operand is coded.

PU:
ATM native connections

VBUILD:
Notes:
1. LIMRES=NO is the only valid value if the TRLE operand is coded.

PU:
- AUTHLEN = YES
- CONNTYPE = APPN

- COSTBYTE = 0
- COSTTIME = 0

- CPCP = YES
  - value_of_cpcp_start_option
    - CPCP = YES
      - value_of_cpcp_start_option
        - CPCP = NO
        - CPCP = YES

- CPNAME = control_point_name

- DYNLU = YES
  - value_of_dynlu_start_option
    - DYNLU = NO
    - DYNLU = YES

- DYNADJCP = YES
  - value_of_dynadjcp_start_option
    - DYNADJCP = NO
    - DYNADJCP = YES
    - HPR = YES
      - HPR = YES
Notes:
1. CPCP defaults to YES for the model RTP PU.
2. CPCP defaults to the value of the CPCP start options for all PUs except the model RTP PU.
3. DYNADJCP=YES is the only valid value if the TRLE operand is coded.
4. VERALSID=NO is the only valid value if the TRLE operand is coded.

**Enterprise Extender predefined connections**

**VBUILD:**
Notes:

1. HOSTNAME does not take a default if IPADDR is coded explicitly on the GROUP statement and HOSTNAME was not explicitly coded on the GROUP statement. If both IPADDR and HOSTNAME are coded explicitly, IPADDR is ignored.

2. IPADDR does not take a default if HOSTNAME is coded explicitly, but IPADDR is not coded explicitly, on the GROUP statement. If both IPADDR and HOSTNAME are coded explicitly, IPADDR is ignored.
Enterprise Extender connection network connections

VBUILD:
Notes:
1. Code VNGROUP if you code VNNAME (or VNTYPE=GLOBAL if you default the name to IP:IP)
2. VNNAME and VNTYPE can not be specified on the GROUP definition statement when that GROUP is specified on the VNGROUP operand on the PORT definition statement.

GROUP:

AUTOGEN = num_stmts, line_seed_char, pu_seed_char

CUADDR = device_address

COSTBYTE = cost_per_byte

COSTTIME = cost_per_unit_time

DIAL = YES

DYNPU = NO

DYNVNPFX = string

EEVERIFY = value_of_eeverify_start_option

HOSTNAME = fully_qualified_hostname

HPREEARB = PROGRESS
Notes:

1 HOSTNAME does not take a default if IPADDR is coded explicitly on the GROUP statement and HOSTNAME was not explicitly coded on the GROUP statement. If both IPADDR and HOSTNAME are coded explicitly, IPADDR is ignored.

2 IPADDR does not take a default if HOSTNAME is coded explicitly, but IPADDR is not coded explicitly, on the GROUP statement. If both IPADDR and HOSTNAME are coded explicitly, IPADDR is ignored.
External communication adapter (XCA) major node operand descriptions

The following information describes the operands you can code for the external communication adapter (XCA) major node.

ADAPNO
ADAPNO=adapter_number

statements: PORT
dependencies: LAN and ATM LAN emulation connections only, MEDIUM
range: 0–127

Specifies a relative adapter number assigned to each adapter or port by an IBM 3172 Nways Interconnect Controller or an Open Systems Adapter.

adapter_number is a 1–3 digit decimal integer in the range 0–127. This operand is used in conjunction with the MEDIUM operand to uniquely define an adapter or port within an IBM 3172 Nways Interconnect Controller or an Open Systems Adapter.

This operand is required for LAN and ATM LAN emulation connections.

ALLOWACT

ALLOWACT=NO
This physical unit does not allow activation of an NCP major node for an NCP running on the Communication Controller for Linux on System z (CCL) product.

ALLOWACT=YES
This physical unit allows activation of an NCP major node for an NCP running on the CCL product.

ANSWER
For LAN, ATM LAN emulation, or Enterprise Extender connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>GROUP, LINE</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies:</td>
<td>Peripheral node CALL=IN or INOUT</td>
</tr>
</tbody>
</table>

For ATM native connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>GROUP, LINE</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies:</td>
<td>TGs over SVCs TGs to connection networks DIAL=YES, CALL=IN or INOUT</td>
</tr>
</tbody>
</table>

Specifies whether physical units can dial in to VTAM.

**ANSWER=OFF**

Physical units cannot dial in to VTAM, regardless of whether the line is active or inactive.

**ANSWER=ON**

Physical units can dial in to VTAM when the line is activated.

The operator can issue a VARY ANS command after activation to change the answer state of the line. See [z/OS Communications Server: SNA Operation](https://www.ibm.com/support/docview.ws/docuid/ibmtr400-4888) for information about using the VARY ANS command.

**Note:** When CALL=OUT, the ANSWER operand is ignored, and the operator receives a message that this operand is not valid.

**AUTHLEN**
AUTHLEN = YES
VTAM will pass the specified transmission priority.

AUTHLEN = NO
VTAM will not pass the specified transmission priority; the default priority of medium will be passed.

AUTOGEN

For LAN, ATM LAN emulation, or Enterprise Extender connections:
statements: GROUP
dependencies: Peripheral node

For ATM native connections:
statements: GROUP
dependencies: TGs over SVCs
TGs to connection networks
DIAL=YES

Defines LINE and PU definition statements that VTAM automatically generates.

num_stmts
Specifies the number of VTAM-generated LINE and PU statements. Code a decimal value in the range 1–65536. Values between 4097 and 65536 may be entered only when MEDIUM=HPRIP is specified for the group. This value is required.

Note: For Interconnect Controller, the range is 1–255. If OF/2 is being used, the range is 1–245.

For ATM Native Connections: num_stmts should be the number of concurrent SVCs that you expect to have activated.

line_seed_char
Defines the first 1–5 characters of the name that VTAM creates for each LINE statement VTAM automatically generates. If num_stmts is greater than 4096, line_seed_char, if coded, must be 1–4 characters long.

pu_seed_char
Defines the first 1–5 characters of the name that VTAM creates for each PU
statement VTAM automatically generates. If \textit{num\_stmts} is greater than 4096, \textit{pu\_seed\_char}, if coded, must be 1-4 characters long.

\textbf{Note:} If you do not want to specify \textit{line\_seed\_char}, code a comma in its place. If you do not want to specify \textit{pu\_seed\_char}, code a comma in its place.

VTAM uses the following naming convention to create a LINE definition statement name and a PU definition statement name:

\begin{verbatim}
sssssnmm
\end{verbatim}

\begin{itemize}
  \item \texttt{ssss}
    \begin{itemize}
      \item Is the value specified by \textit{line\_seed\_char} or \textit{pu\_seed\_char}.
      \begin{itemize}
        \item If you specify a single-character seed value, VTAM adds the device number coded on the CUADDR operand on the PORT definition statement in the XCA major node to the single character seed value, but only if \textit{num\_stmts} is less than 4097. VTAM then adds a sequential number to the end of the name.
        \begin{itemize}
          \item \textbf{Note:} For ATM Native and EE Connections, VTAM uses 0000 for the device number.
        \end{itemize}
        \item If you specify a seed value greater than a single character, VTAM uses the multiple-character seed value you specify, then adds a sequential number to the end of the name.
        \item If you do not specify a seed character, VTAM adds the device number coded on the CUADDR operand on the PORT definitions statement in the XCA major node to the default seed value of O for LINEs and Q for PUs, but only if \textit{num\_stmts} is less than 4097. VTAM then adds a sequential number to the end of the name.
        \begin{itemize}
          \item \textbf{Note:} For ATM Native and EE Connections, VTAM uses 0000 for the device number.
        \end{itemize}
      \end{itemize}
    \end{itemize}
  \item \texttt{nnn}
    \begin{itemize}
      \item Is a sequential hexadecimal number created by VTAM (X'0'–X'FFF').
    \end{itemize}
\end{itemize}

\textbf{Examples:}
The following tables show the LINE and PU definition statements and names that VTAM generates based on the AUTOGEN operand.

\textit{AUTOGEN=(3,,) and CUA=505}:

\textbf{Note:} Seed characters are not specified. Therefore, the default character O is used for LINE statements and the default character Q is used for PU statements.

\begin{table}[h]
\centering
\begin{tabular}{|l|l|l|}
\hline
Statement & XCA AUTOGEN & Statement & XCA AUTOGEN \\
\hline
LINE & O0505000 & PU & Q0505000 \\
LINE & O0505001 & PU & Q0505001 \\
LINE & O0505002 & PU & Q0505002 \\
\hline
\end{tabular}
\caption{XCA AUTOGEN without user-supplied seeds}
\end{table}
**AUTOGEN=(2, K, J) and CUA=505:**

*Table 24. XCA AUTOGEN with single-character user-supplied seeds*

<table>
<thead>
<tr>
<th>Statement</th>
<th>XCA AUTOGEN</th>
<th>Statement</th>
<th>XCA AUTOGEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINE</td>
<td>K0505000</td>
<td>PU</td>
<td>J0505000</td>
</tr>
<tr>
<td>LINE</td>
<td>K0505001</td>
<td>PU</td>
<td>J0505001</td>
</tr>
</tbody>
</table>

**AUTOGEN=(2, KLM, HIJ) and CUA=505:**

*Note:* Device number is not included in the LINE and PU definition statement names because multiple-character seed values are specified for each.

*Table 25. XCA AUTOGEN with 3-character user-supplied seeds*

<table>
<thead>
<tr>
<th>Statement</th>
<th>XCA AUTOGEN</th>
<th>Statement</th>
<th>XCA AUTOGEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINE</td>
<td>KLM00000</td>
<td>PU</td>
<td>HIJ00000</td>
</tr>
<tr>
<td>LINE</td>
<td>KLM00001</td>
<td>PU</td>
<td>HIJ000001</td>
</tr>
</tbody>
</table>

**AUTOGEN=(4097,K,J) and CUA=505:**

*Note:* Device number is not included in the LINE and PU definition statement names because the specified number of statements exceeds 4096 and therefore only 4 characters are available for the seed value.

*Table 26. XCA AUTOGEN with number of statements greater than 4096*

<table>
<thead>
<tr>
<th>Statement</th>
<th>XCA AUTOGEN</th>
<th>Statement</th>
<th>XCA AUTOGEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINE</td>
<td>K0000000</td>
<td>PU</td>
<td>J0000000</td>
</tr>
<tr>
<td>LINE</td>
<td>K0000001</td>
<td>PU</td>
<td>J0000001</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>LINE</td>
<td>K0000FFF</td>
<td>PU</td>
<td>J0000FFF</td>
</tr>
<tr>
<td>LINE</td>
<td>K0001000</td>
<td>PU</td>
<td>J0001000</td>
</tr>
</tbody>
</table>

**CALL**
CALL = IN

Specifies that the line is to be used only for incoming calls (the physical units are to call VTAM).

CALL = INOUT

Specifies that the line is to be used for both incoming and outgoing calls.

CALL = OUT

Specifies that the line is to be used only for outgoing calls (VTAM is to call the physical units).

CAPACITY

Specifies whether physical units, or VTAM, or both, can initiate calls over the line represented by this LINE definition statement.

CALL = IN

For LAN, ATM LAN emulation, or Enterprise Extender connections:

For ATM native connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>GROUP, LINE</th>
<th>statements:</th>
<th>GROUP, LINE</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies:</td>
<td>Peripheral node</td>
<td>dependencies:</td>
<td>TGs over SVCs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TGs to connection networks</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DIAL=YES</td>
</tr>
</tbody>
</table>

CAPACITY = number_of_bits

For LAN, ATM LAN emulation, or Enterprise Extender connections:
For LAN, ATM LAN emulation, or Enterprise Extender connections:

<table>
<thead>
<tr>
<th>statements</th>
<th>PORT, GROUP (TGs to Enterprise Extender connection networks)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependency</td>
<td>TGs to connection networks</td>
</tr>
<tr>
<td></td>
<td>For GROUP; MEDIUM=HPRIP</td>
</tr>
<tr>
<td>range:</td>
<td>1 - 1000Kb</td>
</tr>
<tr>
<td></td>
<td>1 - 1000Mb</td>
</tr>
<tr>
<td></td>
<td>1 - 100Gb</td>
</tr>
</tbody>
</table>

For ATM native connections:

<table>
<thead>
<tr>
<th>statements</th>
<th>GROUP (TGs to connection networks)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PU (TGs over PVCs)</td>
</tr>
<tr>
<td>dependency</td>
<td>DIAL=YES (TGs to connection networks)</td>
</tr>
<tr>
<td></td>
<td>DIAL=NO (TGs over PVCs)</td>
</tr>
<tr>
<td>range:</td>
<td>1 - 1000Kb</td>
</tr>
<tr>
<td></td>
<td>1 - 100Mb</td>
</tr>
<tr>
<td></td>
<td>1 - 100Gb</td>
</tr>
</tbody>
</table>

Specifies the effective capacity of the link that comprises the transmission group (TG). Specify the value in Kb per second (for example, 100K), Mb per second (for example, 100M), or Gb per second (for example, 10G). This number approximates the bits per second that the link can transmit (the transmission rate of the link, times the maximum load factor expressed as a percentage).

**Note:**

1. Because the value for capacity is represented as a single byte, the precision of the specified number of bits might be lost. For example, numbers that are close (such as 100K and 101K) can be interpreted by VTAM, and displayed, as the same value. See Table 45 on page 722 for a list of CAPACITY values you can specify and their corresponding values when displayed.

2. The capacity specified for ATM native connections defined as reserved bandwidth should be related to the forward sustainable cell rate (DLCADDR subfield 7). To calculate the data throughput required for a reserved bandwidth connection, multiply the sustainable cell rate by the payload per cell (48 bytes). If the defined CAPACITY is greater than ten times the ATM reserved bandwidth connection capacity, excess storage might be used and the weighting of this TG with others might cause undesirable results.

**Recommendations:**

- When specified on the PORT statement, VNGROUP and VNNAME (or VNTYPE=GLOBAL) must also be specified on the PORT statement and the
value applies to the TG to the connection network specified in VNNAME (or IPIP if VNNAME is defaulted for VNTYPE=GLOBAL).

- When specified on the GROUP statement, VNNAME or VNTYPE=GLOBAL must also be specified on the GROUP statement. If the GROUP is not defined as a connection network, the TG characteristics cannot be specified on the GROUP definition statement. They must be specified on the PU definition statement in the switched major node (or for dynamically created dial-in EE PUs on the DYNTYPE=EE model major node) for the PU being dialed to establish the connection.

**Guideline:** Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT definition statement, you should define your Enterprise Extender connection network on the corresponding GROUP definition statement instead. By coding all connection network and TG characteristics operands on the GROUP statement, the VNGROUP operand is no longer needed. Also, because there can be multiple connection networks defined in an Enterprise Extender XCA major node with multiple GROUP definition statements, defining all of the operands on the GROUP statement to which they apply provides more clarity.

**CONFIGDS**

```plaintext
cfgds_name
```

*statements:* VBUILD
*dependencies:* none

Specifies a 1–8 character data definition name that identifies the configuration restart data set defined by the user for this major node. Include a DD statement that has this data definition name in the VTAM start procedure.

**CONFGPW**
**CONFIGPW**

statements:  
 dependencies:  

Specifies the 1–7 character alphanumeric password used by VTAM to gain access to the configuration restart data set. If you do not code CONFIGPW, but VSAM requires it, VSAM prompts the VTAM operator for the correct password when VTAM attempts to open the data set.

Code CONFIGPW only if you also code CONFGDS.

**CONNTYPE**

statements:  
 dependencies:  

Specifies that the connection is to be established as an APPN connection that can support parallel TGs, CP-CP sessions, and CP name change support. CONNTYPE=APPN is the only valid value for ATM native connections and is the default value.

**COSTBYTE**
COSTBYTE = 0

cost_per_byte

For LAN, ATM LAN emulation, or Enterprise Extender connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td>PORT, GROUP</td>
<td>GROUP (TGs to connection</td>
</tr>
<tr>
<td>(TGs to Enterprise</td>
<td>networks)</td>
</tr>
<tr>
<td>Extender connection</td>
<td>PU (TGs over PVCs)</td>
</tr>
<tr>
<td>networks)</td>
<td></td>
</tr>
</tbody>
</table>

dependency:

- TGs to connection networks
- For GROUP; MEDIUM=HPRIP

<table>
<thead>
<tr>
<th>range:</th>
<th>range:</th>
</tr>
</thead>
<tbody>
<tr>
<td>0–255</td>
<td>0–255</td>
</tr>
</tbody>
</table>

Specifies a cost-per-byte-transmitted to be associated with the transmission group (TG). A value of 0 is the least expensive cost per byte and 255 is the most expensive.

**Recommendations:**

- When specified on the PORT statement, VNGROUP and VNNAME (or VNTYPE=GLOBAL) must also be specified on the PORT statement and the value applies to the TG to the connection network specified in VNNAME (or IP/IP if VNNAME is defaulted for VNTYPE=GLOBAL).

- When specified on the GROUP statement, VNNAME or VNTYPE=GLOBAL must also be specified on the GROUP statement. If the GROUP is not defined as a connection network, the TG characteristics cannot be specified on the GROUP definition statement. They must be specified on the PU definition statement in the switched major node (or for dynamically created dial-in EE PUs on the DYNTYPE=EE model major node) for the PU being dialed to establish the connection.

**Guideline:** Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT definition statement, you should define your Enterprise Extender connection network on the corresponding GROUP definition statement instead. By coding all connection network and TG characteristics operands on the GROUP statement, the VNGROUP operand is no longer needed. Also, because there can be multiple connection networks defined in an Enterprise Extender XCA major node with multiple GROUP definition statements, defining all of the operands on the GROUP statement to which they apply provides more clarity.

**COSTTIME**
COSTTIME = \text{cost per unit time}

<table>
<thead>
<tr>
<th>For LAN, ATM LAN emulation, or Enterprise Extender connections:</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>statements:</strong> PORT, GROUP (TGs to Enterprise Extender connection networks)</td>
<td><strong>statements:</strong> GROUP (TGs to connection networks)</td>
</tr>
<tr>
<td><strong>dependency:</strong> TGs to connection networks</td>
<td><strong>dependency:</strong> DIAL=YES (TGs to connection networks)</td>
</tr>
<tr>
<td>For GROUP; MEDIUM=HPRIP</td>
<td>DIAL=NO (TGs over PVCs)</td>
</tr>
<tr>
<td><strong>range:</strong> 0–255</td>
<td><strong>range:</strong> 0–255</td>
</tr>
</tbody>
</table>

Specifies a cost-per-unit-of-time to be associated with the transmission group (TG). A value of 0 is the least expensive cost per unit of time and 255 is the most expensive.

**Recommendations:**
- When specified on the PORT statement, VNGROUP and VNNAME (or VNTYPE=GLOBAL) must also be specified on the PORT statement and the value applies to the TG to the connection network specified in VNNAME (or IP/IP if VNNAME is defaulted for VNTYPE=GLOBAL).
- When specified on the GROUP statement, VNNAME or VNTYPE=GLOBAL must also be specified on the GROUP statement. If the GROUP is not defined as a connection network, the TG characteristics cannot be specified on the GROUP definition statement. They must be specified on the PU definition statement in the switched major node (or for dynamically created dial-in EE PUs on the DYNTYPE=EE model major node) for the PU being dialed to establish the connection.

**Guideline:** Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT definition statement, you should define your Enterprise Extender connection network on the corresponding GROUP definition statement instead. By coding all connection network and TG characteristics operands on the GROUP statement, the VNGROUP operand is no longer needed. Also, because there can be multiple connection networks defined in an Enterprise Extender XCA major node with multiple GROUP definition statements, defining all of the operands on the GROUP statement to which they apply provides more clarity.

**CPCP**
### Notes:

1. CPCP defaults to YES for the model RTP PU.
2. CPCP defaults to the value of the CPCP start options for all PUs except the model RTP PU.

**statements:**

**dependencies:**

ATM native connections only (TGs over PVCs), DIAL=NO

Specifies whether CP-CP sessions are supported on this connection. If you do not code this operand, the value defaults to the value specified on the CPCP start option.

**CPCP=NO**

Indicates CP-CP sessions are not supported on this connection.

**CPCP=YES**

Indicates CP-CP sessions are supported on this connection.

### CPNAME
Identifies the resource's owning control point name. If it is later learned that the owning control point name is different than what is specified here, the resource will then be associated with the newly learned control point name.

**CUADDR**

Defines a 3-digit or 4-digit hexadecimal device number used for the attached LAN. The device number should be within the address range defined to the operating system.

The address must not be enclosed in quotation marks or apostrophes. For example, a channel device address of X'080' should be coded CUADDR=080.

This operand is required for LAN and ATM LAN emulation connections.
DELAY = timeout_value

**statements:** PORT
**dependencies:** LAN and ATM LAN emulation connections only (subarea node)
**range:** 0–9.999

Specifies the maximum amount of time, in seconds and thousandths of seconds, that VTAM should wait before transmitting low-priority data (that is, PIU transmission priority 0 or 1) to the physical unit.

For more efficient operation, code this value as high as possible. For maximum speed across the interface, do not code delay. See z/OS Communications Server: SNA Network Implementation Guide for more information.

**DIAL**

**DIAL = NO**

**DIAL = YES**

**statements:** GROUP
**dependencies:** none

Specifies whether the lines under the GROUP statement require switched line control protocols.

**Note:** You must code DIAL=YES for peripheral nodes and DIAL=YES when MEDIUM=HPRIP on PORT for Enterprise Extender.

**DIAL = NO**

Specifies that the lines in the group do not require switched line control protocols.

**DIAL = YES**

Specifies that the lines in the group require switched line control protocols.

**DLCADDR**
Specifies the signaling information used to establish a switched connection with a remote node through an ATM native connection network.

Multiple DLCADDR operands can be coded on a single GROUP definition statement. For example, the following information can be coded on a single GROUP statement:

```
GROUP DLCADDR=(1,TYPE1,STRING1),
   DLCADDR=(2,TYPE2,STRING2)
```

**subfield_id**
Indicates the subfield. A value between 1–96 can be specified.

The first DLCADDR operand must have a `subfield_id` of 1. Also, each operand must have a unique `subfield_id`. If two DLCADDR operands—after a DLCADDR operand with a `subfield_id` of 1—have the same `subfield_id`, VTAM treats the two operands as duplicates. VTAM ignores the second operand and issues an error message.

**Note:** If you receive an error message referring to the first DLCADDR operand coded on a GROUP statement, limited syntax checking is done on the subsequent DLCADDR operands coded.

**type**
Indicates the data format of the signaling information and is one of the following formats:

<table>
<thead>
<tr>
<th>Type</th>
<th>Data Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Compressed alphanumeric</td>
</tr>
<tr>
<td>BCD</td>
<td>Binary coded decimal</td>
</tr>
<tr>
<td>C</td>
<td>EBCDIC</td>
</tr>
<tr>
<td>D</td>
<td>Decimal</td>
</tr>
<tr>
<td>X</td>
<td>Hexadecimal</td>
</tr>
</tbody>
</table>

Type C (EBCDIC) is the default.

**string**
Defines the signaling information in the data format defined by `type`. Multiple `string` values can be coded for a `subfield_id` `type` pair. For example:

```
DLCADDR=(1,TYPE1,STRING1,STRING2,STRING3)
```

You can specify up to 250 bytes of signaling information on a single DLCADDR operand. If coding multiple `string`s on a single operand, the total number of bytes available for signaling information is reduced by 2 for each additional DLCADDR `string`.

The first DLCADDR operand defined for each GROUP definition statement identifies the DLC type, which for ATM native connection-network connections is
ATMSVC. The remaining DLCADDR operands define the DLC signaling information for the ATM native connection-network connection. You can code only one set of DLCADDR operands on the GROUP definition statement for an ATM native connection-network connection.

Code the following DLCADDR operands for an ATM native connection-network connection. Fields in italics are user-defined fields.

DLCADDR=(1,C,ATMSVC,vnname,tg_use)
DLCADDR=(7,BCD,format,best_effort_indicator,a,b,c,
forward_traffic_management_options)
DLCADDR=(8,X,format,outbound_QOS_class)
DLCADDR=(9,BCD,tns_identifier)
DLCADDR=(21,BCD,type,ATM_address)
DLCADDR=(51,X,format,a,b,c,d,e)
DLCADDR=(61,BCD,format,05,a,b)

The following list explains the information provided by each subfield:

<table>
<thead>
<tr>
<th>Subfield</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>C indicates EBCDIC format. ATMSVC indicates that the connection provides native access to an ATM network. Code vnname to specify a 1–17 character network-qualified CP name for the connection network. vnname is reported to the network topology as a virtual node and is treated as an adjacent CP to this node. If the name is not network-qualified, the network ID of the SSCP that owns the connection network (network ID of the host) is used. For sessions that are routed through the connection network, a dynamic PU is created to represent the destination node. In effect, DYNPU=YES is enforced automatically when a session is established through the connection network. Code tg_use to specify one of the following items: EXCLUSIVE Indicates that SVCs and TGs can be established that will handle sessions only over connection network connections. Sessions through a connection network cannot be routed over an SVC already established to the same remote node if that SVC was not established through a connection network. Shared sessions can be shared between connection network connections and non-connection networks over an SVC. SHARE Indicates that HPR sessions routed through connection networks can use an SVC that is already established to the same remote node, even if that SVC was not established through a connection network.</td>
</tr>
<tr>
<td>7</td>
<td>BCD indicates binary coded decimal format. Code format to specify one of the following items: 00 Indicates ITU-T defined 03 Indicates ATM network defined Code best_effort_indicator to specify one of the following items: 00 Indicates that guaranteed bandwidth is wanted</td>
</tr>
</tbody>
</table>
01 Indicates that best effort is acceptable

Code \(a\), \(b\), and \(c\) as 4-byte decimal values to specify the following items:

\(a\) Forward peak cell rate (cells per second) for cell loss priority=0+1 (CLP=0+1)
\(b\) Forward sustainable cell rate (cells per second) for (CLP=0)
\(c\) Forward maximum burst size (cells) for (CLP=0)

Code `forward_traffic_management_options` to specify one of the following items:

00 Indicates that tagging is not requested in the forward direction
01 Indicates that tagging is requested in the forward direction

**Note:** If `best_effort_indicator` is 00, code `format`, \(a\), \(b\), \(c\), and `forward_traffic_management_options`.

If `best_effort_indicator` is 01, code `format` and \(a\).

8 \(X\) indicates hexadecimal format.

Code `format` to specify one of the following items:

00 Indicates ITU-T defined
03 Indicates ATM network defined

Code `outbound_QOS_class` to specify the QOS class for outbound (forward) data traffic.

9 BCD indicates binary coded decimal format.

Code `tns_identifier` to specify the carrier identification code for the carrier that will be used for this connection.

21 BCD indicates binary coded decimal format.

Code `type` to specify the following items:

0101 Indicates public E164 address, which means the address is in a public ATM network. The address is variable in length.

Code `ATM_address` only if remote nodes that use the same connection network are attached directly to public networks. Code `ATM_address` to specify the public gateway E164 address through which this port can be reached.

51 \(X\) indicates hexadecimal format.

Code `format` to specify the following items:

00 Indicates ITU-T defined

Code \(a\), \(b\), \(c\), \(d\), and \(e\) to specify the following bearer capabilities:

\(a\) Bearer class.

\(a\) can be one of the following items:

01 Indicates class A
03 Indicates class C
10 Indicates class X

\(b\) Susceptibility to clipping.
$b$ can be one of the following items:

- 00 Not susceptible
- 01 Susceptible

$c$ User plane connection configuration.

$c$ must be:

- 00 Point-to-point

$d$ If bearer class is 10, indicates the traffic type.

$d$ can be one of the following items:

- 00 No indication
- 02 Variable bit rate

$e$ If bearer class is 10, indicates timing requirements.

$e$ can be one of the following items:

- 00 No indication
- 01 End-to-end timing required

61 BC0 indicates binary coded decimal format.

Code format to specify the following:

- 00 Indicates ITU-T defined

Code 05 to indicate ATM adaptation layer (AAL) type 5.

Code $a$ and $b$ as 4-byte decimal values to specify the following AAL parameters:

- $a$ Forward CPCS-SDU maximum size
- $b$ Backward CPCS-SDU maximum size

Note: If an application using an ATM reserved bandwidth SVC connection sends RUs smaller than the forward CPCS-SDU size, the throughput for the session may be reduced. For example, if the forward CPCS-SDU size was defined or allowed to default to 8448 (Maximum RU size 8K + 256), and the data being transmitted over this ATM connection is in 1K RUs, the throughput (as defined by the connection's forward cell rate) is greatly reduced. If this ATM SVC connection is used for 1K RU data transmission, the forward CPCS-SDU size should be set to 1280 bytes (Maximum RU size of 1K + 256).

**DYNADJCP**
Notes:

1. DYNADJCP=YES is the only valid value if the TRLE operand is coded.

Statements:

PU

Dependencies: ATM native connections only (TGs over PVCs), DIAL=NO; ignored if CPNAME coded

Specifies whether an ADJCP can be created dynamically for an adjacent node.

Note: This operand is ignored if CPNAME is coded on the physical unit.

**DYNADJCP=NO**

Indicates that the ADJCP cannot be created dynamically for an adjacent node.

**DYNADJCP=YES**

Indicates that the ADJCP can be created dynamically for an adjacent node.

**DYNLU**
Specifies whether dynamic allocation of CDRSC definitions is supported. If you do not code this operand, the default is the value you coded on the DYNLU start option.

If dynamic allocation of CDRSC definitions is not supported for a particular PU, a resource that was dynamically defined as a CDRSC cannot use that PU to connect to a session.

**DYNLU=NO**

Specifies that dynamic allocation of CDRSC definitions is not supported. If you code DYNLU=NO and plan to connect to other nodes using CP-CP sessions, then redefine each node in the CDRSC major node. If you do not, CP-CP session setup fails.

**DYNLU=YES**

Specifies that dynamic allocation of CDRSC definitions is supported.

For APPN-only nodes (HOSTSA is not specified), DYNLU=YES is meaningful even though a host CDRM cannot be activated.

**Note:** If you have coded a DYNLU value in the adjacent CP major node for the adjacent CP to which this PU is linked, that value overrides the DYNLU value you code on this PU definition statement. For additional DYNLU considerations, see "DYNLU" on page 23 under the adjacent control point major node.

**DYNPU**
Specifies whether a PU is to be dynamically allocated when the calling PU cannot be identified during a switched call-in operation. DYNPU applies to APPN and subarea PUs.

For non-connection-network Enterprise Extender connections only, if a model PU with DYNTYPE=EE specified has been defined in a model major node and that node has been activated, the dynamic PU created because of the DYNPU operand will use the characteristics coded or defaulted for that model. (See the DYNTYPE operand description in the model major node information for details.) Otherwise, a PU created by the DYNPU operand will use the switched major node PU operand defaults, except for the following operands which will use the values noted:

- ANS=CONT
- DISCNT=(YES,F)
- DYNADJCP=YES
- MAXOUT=8

A model PU definition can be created to customize the characteristics of dynamically created connection network PUs. Use the DYNTYPE=VN operand for the model PU definition in the model major node.

**DYNPU=NO**

Specifies that dynamic PUs are not to be used.

**DYNPU=YES**

Specifies that dynamic PUs can be used.

**Rule:** When more than one non-connection-network Enterprise Extender call-in capable group is defined for a given static VIPA (IPADDR), the DYNPU value that is specified on each of these groups must be consistent. The first line that is activated from one of these groups sets the DYNPU value for non-connection-network dial-in operations. Any subsequent line activation that is associated with a group for the same static VIPA, which has a conflicting DYNPU value, will fail activation with the sense code X'08490000'.

### For LAN, ATM LAN emulation, or Enterprise Extender connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>GROUP (peripheral node)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies:</td>
<td>None</td>
</tr>
</tbody>
</table>

### For ATM native connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>GROUP (TGs over SVCs, TGs to connection networks)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies:</td>
<td>DIAL=YES</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DYNPU=NO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies that dynamic PUs are not to be used.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DYNPU=YES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies that dynamic PUs can be used.</td>
</tr>
</tbody>
</table>
Specifies the first two characters of the PU name assigned when a dynamically generated PU is created for the link station. VTAM concatenates the characters specified to form a VTAM-generated character sequence.

**Note:** This operand is valid only if DYNPU is specified as YES.

**DYNUPFX=CN**

Indicates that CN is the first two characters of the name assigned to the dynamically created PU.

**Note:** VTAM will add a third character “V” for a dynamic PU created for a connection network.

**DYNUPFX=string**

Allows you to specify the first two characters of the name assigned to the dynamically created PU. If you create your own identifier instead of using the default, you must follow VTAM naming conventions. See "Restrictions on names" on page 14 for naming conventions.

**DYNVPFX**
DYNVNPFX = value_of_dynvnpfx_start_option

DYNVNPFX = string

Statements: GROUP (Peripheral node)
Dependencies: VNGROUP on PORT must identify this GROUP name

Specifies the first two characters of the PU name assigned when a dynamic connection network PU is created. VTAM concatenates the characters specified to form a VTAM-generated character sequence.

Note: This operand is valid only if this GROUP was identified by the VNGROUP operand on a PORT definition statement in the same External Communication Adapter Major Node.

DYNVNPFX=string

Allows you to specify the first two characters of the name assigned to dynamic connection network PUs. If you specify your own identifier (using this operand) instead of using the default, you must follow VTAM naming conventions. See “Restrictions on names” on page 14 for naming conventions.

EEVERIFY

EEVERIFY = value_of_eeverify_start_option

EEVERIFY = ACTIVATE

EEVERIFY = NEVER

EEVERIFY = Number_of_minutes

Statements: GROUP
Dependencies: Enterprise Extender connection network connections only
Range: 1 - 1440 (in minutes)

Specifies whether VTAM should automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the Enterprise Extender (EE) connection. The EEVERIFY option might also specify the time interval during which VTAM should send the probe on active EE connections.

The EEVERIFY operand is valid only for the EE connection network defined by this group. If you do not code this operand, VTAM uses the value that you coded on the EEVERIFY start option.

EEVERIFY=ACTIVATE

Specifies that VTAM should automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the EE connection. If one or more ports are not reachable or are blocked, then VTAM issues an error message and does not activate the EE connection.

EEVERIFY=NEVER

Specifies that VTAM should never automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the EE connection.

EEVERIFY=Number_of_minutes

Specifies that VTAM should automatically send an LDLC probe during the
activation of the EE connection and that VTAM should continue to periodically send LDLC probes at intervals specified by the `Number_of_minutes` value while the connection is active. Valid values for `Number_of_minutes` are in the range 1 - 1440 minutes (1440 minutes specifies 1 day). If one or more ports are not reachable or are blocked at activation, then VTAM issues an error message and does not activate the EE connection. If one or more ports are not reachable after the connection is active, VTAM issues an error message, but the connection remains active.

For predefined EE connections, define this parameter on the PU definition in the switched major node. For dial-in EE connections that have their associated PUs dynamically created, define this parameter on the model major node (DYNTYPE=EE) PU definition statement.

**HOSTNAME**

```
HOSTNAME=value_of_hostname_start_option
HOSTNAME=fully_qualified_hostname
```

*statements: GROUP
*dependencies: Enterprise Extender Connections only

Specifies the host name associated with the local TCP/IP stack to be used for Enterprise Extender connections utilizing this GROUP definition. The HOSTNAME you code should be a fully qualified TCP/IP domain name, up to 64 characters long. The HOSTNAME you code will be used to acquire, through DNS name-to-address resolution, the static VIPA address which other Enterprise Extender nodes will use in order to communicate with this host using the Enterprise Extender connection.

HOSTNAME can be used as an alternative to specifying an address explicitly using IPADDR. The benefit to using HOSTNAME, versus IPADDR, is that remote nodes attempting to establish EE connections to this host can perform name-to-address resolution on `hostname`, which they receive in the APPN session setup flows, and acquire either the static VIPA address (if no address translation is required) or a network address translation (NAT) address that will ultimately get them to the correct static VIPA address (if network address translation is employed). The downside of using HOSTNAME is that additional DNS definition and coordination might be required to get the correct resolution in all domains. The value of `hostname` can be resolved to either an IPv4 address or an IPv6 address, but it is strongly advised that a `hostname` resolve to a single static VIPA address, to avoid complications during EE connection establishment.

Specifying the HOSTNAME or IPADDR operand on the Enterprise Extender GROUP definition statement is necessary when your Enterprise Extender configuration requires the use of multiple static VIPAs. The HOSTNAME may also be defined as a VTAM start option. If the HOSTNAME operand is not specified on the Enterprise Extender GROUP definition statement, it will default to the value of the HOSTNAME start option, unless IPADDR is coded explicitly on the GROUP statement. When the IPADDR operand is present on the GROUP definition statement, the sift-down effect of the HOSTNAME start option is ignored.

Note:
1. If IPADDR or HOSTNAME is specified on any GROUP definition statement within the Enterprise Extender (XCA) definition, then one of those two operands must be specified on all Enterprise Extender GROUP definition statements (or defaulted from the VTAM start options).

2. The static VIPA addresses acquired by the way of resolving hostname are validated by TCP/IP during port activation. All static VIPA addresses defined must pass TCP/IP validation prior to the Enterprise Extender connection being available for use.

3. The fullyQualifiedHostname may be up to 64 characters in length. See z/OS Communications Server: SNA Network Implementation Guide for HOSTNAME length considerations.

4. The TCPNAME start option is required in a CINET environment if any of the following conditions exist:
   • The IPADDR parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
   • The HOSTNAME parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
   • The HOSTNAME keyword is specified as a VTAM start option.

   If OMVS is not initialized at Enterprise Extender line activation, a Common INET environment is assumed.

5. See z/OS Communications Server: IP Configuration Guide for information about UNIX Systems Services security considerations and required security definitions.
HPR = YES

Specifies that High-Performance Routing (HPR) support is enabled. HPR=YES is the only valid value for native ATM connections and is the default value.

**HPREEARB**

This operand specifies that the progressive-mode adaptive-rate-based (ARB) flow control algorithm is preferred for the High-Performance Routing (HPR) pipes that traverse the Enterprise Extender (EE) connections that are associated with the connection network that is defined by this group. Progressive mode is used only when the HPR pipe is a single-hop pipe (which includes a single physical hop across a 2-hop EE virtual routing node [VRN]) and the RTP partner supports progressive mode.

For dial-in EE connections that have their associated PUs dynamically created, define this parameter on the model major node (DYNTYPE=EE) PU definition statement. For predefined EE connections, define this parameter on the GROUP or PU definition that is in the switched major node.

**Result:** If a predefined EE connection is established between two EE endpoints before the VRN connection is established (using the same VIPA addresses), the VRN connection uses the established predefined EE connection. In this case, to control the HPREEARB value associated with the EE VRN, you must specify the HPREEARB operand on the EE PU definition statement (for example, on the switched PU definition).

**HPREELIV**

HPREELIV=YES
HPR Liveness Reduction for EE is enabled. When an RTP pipe is comprised of a single EE connection (which includes a single physical hop across a 2-hop EE virtual routing node [VRN]), the HPR ALIVE timer will not be used, thereby relying on the EE LDLC layer for inactivity monitoring and saving the timer maintenance and keep-alive status request/reply overhead. RTP pipes comprised of multi-hop or single non-EE hops will continue to use ALIVE timer processing.

HPREELIV=NO
HPR Liveness Reduction for EE is disabled. All RTP pipes will use ALIVE timer processing.

Tip: Use the default of HPREELIV=YES.

HPRPSDLY

Specifies the time that elapses before RTP pipes enter the path switch state.

The ARB flow control algorithm is sensitive to minor variations in the round-trip time and to unpredictable response times from the partner. This sensitivity might cause RTP endpoints to prematurely enter the path switch state. Although this situation does not affect availability, it does consume CPU cycles and it also causes a significant number of path switch messages to be written to the console log.

The HPRPSDLY operand is valid only for RTP pipes that are directly attached to the Enterprise Extender (EE) connection network that is defined by this group. This operand affects only path switches that are the result of an unresponsive partner (path switches that generate the message IST1818I PATH SWITCH REASON: SHORT REQUEST RETRY LIMIT EXHAUSTED). The HPRPSDLY operand does not control path switches that are initiated as a result of a TG INOP, FRTP commands, or the PSRETRY function. If you do not code this operand, VTAM uses the value that you coded on the HPRPSDLY start option.

HPRPSDLY=ps_delay
Specifies the number of seconds that RTP pipes must delay prior to entering the path switch state. During this time, the RTP endpoint periodically tries to contact the partner in an effort to avoid switching paths. The value 0 indicates...
that the RTP nodes enter the path switch state when a predetermined number of retry attempts have been unsuccessful.

**HPRPSDLY=EEDELAY**

Specifies that VTAM calculates the number of seconds that RTP pipes must delay prior to entering the path switch state. The value that is calculated allows enough time for the Enterprise Extender (EE) keep-alive mechanism to cause the EE connection to become inoperative if connectivity to the partner is lost. Unnecessary path switches are avoided while EE determines whether there is a loss of connectivity to the partner.

For predefined EE connections, define this parameter on the PU definition in the switched major node. For dial-in EE connections with associated PUs that are dynamically created, define this parameter on the model major node (DYNTYPE=EE) PU definition statement.

**Results:**

- When the EEDELAY value is specified, the value that is calculated for the HPR path-switch delay might be as long as 253 seconds.
- If a predefined EE connection is established between two EE endpoints before the VRN connection is established (using the same VIPA addresses), the VRN connection uses the established predefined EE connection. In this case, to control the HPRPSDLY value associated with the EE VRN, you must specify the HPRPSDLY operand on the EE PU definition statement (for example, on the switched PU definition). Optionally, you can specify the HPRPSDLY value as a VTAM start option.

**Tips:**

- This operand can be useful when the RTP partner is on a node that is CPU-constrained or that is running in a virtualized environment. In both of these situations, allowing additional time for the RTP partner to respond might avoid unnecessary processing associated with a path switch.
- If alternate routes exist, specifying a long delay time might cause unnecessary delays for the sessions that are using this RTP pipe.

**IPADDR**
IPADDR — value_of_ipaddr_start_option
IPADDR — IP_address

Specifies the IP address at the local TCP/IP stack. The static VIPA specified here will be the address which other Enterprise Extender nodes will use in order to communicate with this host using the Enterprise Extender connection.

Specifying the IPADDR or HOSTNAME operand on the Enterprise Extender GROUP definition statement is necessary when your Enterprise Extender configuration requires the use of multiple static VIPAs. The IPADDR may also be defined as a VTAM start option. If the IPADDR operand is not specified on the Enterprise Extender GROUP definition statement, it will default to the value of the IPADDR start option, unless HOSTNAME is coded explicitly on the GROUP statement. When the HOSTNAME operand is coded, there is no sift-down effect from the IPADDR start option.

The address must be specified in dotted decimal format for an IPv4 address (for example, IPADDR=4.6.41.7) or in colon-hexadecimal format for an IPv6 address (for example, IPADDR=2000::67:1:2).

Note:
1. If IPADDR or HOSTNAME is specified on any GROUP definition statement within the Enterprise Extender (XCA) definition, then one of those two operands must be specified on all Enterprise Extender GROUP definition statements (or defaulted from the VTAM start options).
2. The TCPNAME start option is required in a CINET environment if any of the following conditions exist:
   • The IPADDR parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
   • The HOSTNAME parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
   • The HOSTNAME keyword is specified as a VTAM start option.

If OMVS is not initialized at Enterprise Extender line activation, a Common INET environment is assumed.

IPPORT
IPPORT = number

Specifies the first number of an ascending sequence of five consecutive numbers that describe the ports used to transmit signal, network, high, medium, and low priority data, respectively. Use of the default value of 12000 is recommended, because all other partner nodes must use the same set of UDP port numbers to connect to this node using Enterprise Extender. Other Enterprise Extender platforms might not offer the capability of using port numbers other than 12000–12004.

IPRESOLV

Specifies the number of seconds VTAM waits for name-to-address resolution requests to complete before canceling the request. The default is 0 seconds, meaning VTAM will wait indefinitely for resolution to be performed. Any value from 0 to 255 seconds can be specified.

If a nonzero value is specified for IPRESOLV, the value should be chosen after giving consideration to the values defined for resolver timeouts on the TCP/IP stack (RESOLVERTIMEOUT directive in the TCPIP:DATA configuration file), the number of retries attempted (RESOLVERUDPRETRIES directive in the TCPIP:DATA configuration file), and the number of domain names that can be appended to the host name while performing name-to-address resolution (SEARCH or DOMAINORIGIN directive in the TCPIP:DATA configuration file). If a large timeout is specified for RESOLVERTIMEOUT, a number of retries is specified for RESOLVERUDPRETRIES, and/or a large number of domains is specified on SEARCH or DOMAINORIGIN, the resolver might take some time to complete the name-to-address resolution attempt, so IPRESOLV should be set to value large enough to allow the resolver to complete its processing. See z/OS Communications Server: IP Configuration Reference for details on the resolver directives.

Alternatively, you might want to ensure that the hostname being resolved (the HOSTNAME operand on the GROUP statement, or the HOSTNAME start option value) is defined in the local hosts files on this system, and then specify LOOKUP LOCAL DNS in the TCPIP:DATA configuration file to ensure that local hosts files are searched first (see z/OS Communications Server: IP Configuration Reference for details about the LOOKUP resolver directive). Coding LOOKUP LOCAL DNS
affects all resolver searches, not just Enterprise Extender searches initiated by VTAM, so this option should be chosen with caution.

Note:
1. The IPRESOLV keyword has been set to a default value of zero because in most cases it is desirable to establish the amount of time allowed before hostname resolution timeout by using the value in the TCP/IP configuration statements. Reset this value only when a VTAM hostname resolution time limit is required.
2. The RESOLVERUDPENTRIES option is applicable only when the resolver is communicating with the name server using the UDP protocol. (This is the default communication method and is specified with the RESOLVEVIA TCPIP.DATA statement.)

**IPTOS**

IPTOS = (20,40,80,C0,C0)

statements: PORT
dependencies: MEDIUM=HPRIP
range: 00–FF

Specifies the Type of Service use for TCP/IP communication. Default values will be used for each Type of Service value which is not entered.

The first value (20) is represented as low, the second value (40) as medium, the third value (80) as high, the fourth value (C0) as network, and the last value (C0) as signal for Type of Service.

**Note:** The IP precedence values (within the TOS byte) need to be consistent throughout the network. Enterprise Extender allows you to specify the IP TOS byte which contains the IP precedence field. You can specify one set of IPTOS values for Enterprise Extender. If more granular control is required for IP precedence values, see Policy Agent in the z/OS Communications Server: IP Configuration Reference.

For more information about Type of Service and Class of Service, see z/OS Communications Server: SNA Network Implementation Guide.

**ISTATUS**
ISTATUS=ACTIVE
Specifies that the line or PU becomes active when its major node becomes active.

ISTATUS=INACTIVE
Specifies that the line or PU remains inactive until the operator activates it.

The operator can override this value by specifying the SCOPE operand on the VARY ACT command when the major node is activated. See z/OS Communications Server: SNA Operation for information about using the VARY ACT command.

KEEPACT

KEEPACT=NO
Specifies that the line is not to be automatically reactivated after a link INOP.

KEEPACT=YES
Specifies that the line is to be automatically reactivated after a link INOP.
Coding KEEPACT=YES, or allowing KEEPACT to default to YES, can reduce the need for operator intervention in reestablishing Enterprise Extender connections.

LANTEST
**LANTEST=LOCAL**: Specifies that a TEST frame will first be sent out on the LOCAL token ring. If no response is received within the PORT TIMER interval, a GLOBAL TEST frame will be sent. Each time the GLOBAL TEST receives no response within the PORT TIMER interval, another GLOBAL TEST will be sent until the TEST receives a response, or the connection is deactivated.

**LANTEST=GLOBAL**: Specifies that only GLOBAL TEST frames will be used to locate the device. Each time the GLOBAL TEST receives no response within the PORT TIMER interval, another GLOBAL TEST will be sent until the TEST receives a response, or the connection is deactivated.

**Restriction**: If you specify MEDIUM=CSMACD (Ethernet) on the associated XCA PORT definition statement, the parameter has no meaning and is ignored.

This parameter allows the user to define how subarea nodes connected on the token ring will be located. There are two types of TEST commands sent: LOCAL and GLOBAL. The LOCAL will only flow on the local token ring. The GLOBAL will flow with a routing information field. This routing information field has the broadcast bit on, meaning to route the test into the entire network.

**Note**: For switched XCA PU nodes, this parameter might be defined on the PU definition in the switched major node. For XCA connection network PUs, this parameter exists under the model major node (DYNTYPE=VN).
Notes:

1. LIMRES=NO is the only valid value if the TRLE operand is coded.

<table>
<thead>
<tr>
<th>For LAN and ATM LAN emulation connections:</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td>statements:</td>
<td>statements:</td>
</tr>
<tr>
<td>GROUP, LINE (peripheral node)</td>
<td>GROUP, LINE (TGs over SVCs, TGs to connection networks)</td>
</tr>
<tr>
<td>dependencies:</td>
<td>dependencies:</td>
</tr>
<tr>
<td>LIMQSINT</td>
<td>LIMQSINT</td>
</tr>
<tr>
<td>LU 6.2 only</td>
<td>LU 6.2 only</td>
</tr>
<tr>
<td>PUTYPE=1 or 2</td>
<td>DIAL=YES (TGs over SVCs, TGs to connection networks)</td>
</tr>
</tbody>
</table>

Specifies whether a line or group of lines is to be treated as a limited resource. A limited resource allows you to limit the use of some network connections. It also causes any sessions that traverse the resource to be deactivated if no conversation is active. A session also can be deactivated if the time limit you specify for the LIMQSINT operand on the APPL definition statement expires.

This operand applies only to LU 6.2 conversations and PU types 1 and 2.

If you do not code a value for LIMQSINT, resources are not considered to be limited resources. Furthermore, sessions will remain active even after all conversations have ended.

You can also use LIMRES in conjunction with the LIMINTCP start option to specify the interval to retain a free session using a limited resource for some network management flows (excluding flows using the APPN control point sessions or dependent LU server sessions). If you do not code a value for the LIMINTCP start option, resources are not considered to be limited resources, and the sessions will remain active even after all conversations have ended.

Note: This is not a meaningful operand for Enterprise Extender.

LIMRES=NO

Specifies that this line or group of lines is not to be treated as a limited resource.

LIMRES=YES

Specifies that this line or group of lines is to be treated as a limited resource.

LIMRES=YES must be coded for the LIMINTCP start option to be effective.

LIVTIME
LIVTIME = (init_value, max_value)

**Statements:** PORT, GROUP

**Dependencies:** MEDIUM=HPRIP

**Range:**
- init_value 5-25
- max_value 0 or init_value-3600

Specifies the Enterprise Extender logical data link control liveness timer interval range, in seconds.

There are two timer functions that affect Enterprise Extender connections: the logical data link control (LDLC) timer and the disconnect timer. The operands that control the LDLC timer that are specified on the PORT and GROUP macros are LIVTIME, SRQTIME, and SRQRETRY. The disconnect timer is controlled by the DISCNT keyword on the PU macro.

LDLC uses the LDLC timer interval to determine how long to wait before considering an Enterprise Extender connection inoperative. Two values can be specified on the LDLC liveness timer (LIVTIME). These values are optional with the first being the initial LIVTIME value (init_value) and the second the maximum LIVTIME value (max_value). If the max_value is specified or defaulted to 0, or the max_value is specified as equal to the init_value, the LDLC timer facility uses the initial value for the liveness timer.

During periods of inactivity on the Enterprise Extender connection, when the liveness timer expires, LDLC polls the partner with an LDLC TEST request. This verifies that the EE partner is still available. In Enterprise Extender configurations that experience long periods of inactivity, the LDLC TEST flows or HPR status flows might be the only flows occurring. In configurations involving large numbers of EE connections, reducing these flows can significantly reduce EE overhead (CPU consumption).

Specifying a max_value larger than the init_value enables the EE LDLC Keep-Alive Reduction Function. This function enables the current LIVTIME window to expand and contract based on current network conditions. Expanding the current LIVTIME window reduces the number of LDLC test flows that occur during periods of inactivity.

The following information shows the basic set of rules of the EE LDLC Keep-Alive Reduction Function:

- The current LIVTIME interval is always initialized to the init_value when the connection is established.
- During a specific LIVTIME interval, if the only activity over the EE connection are LDLC TEST flows, the current LIVTIME value is doubled in size. The current LIVTIME value can double in size up to the size specified by max_value. This guarantees that VTAM will poll the EE partner at least once every number of seconds specified by the max_value value.
When HPR traffic starts flowing over an inactive EE connection and the current LIVTIME window has expanded to be larger than the initial value, VTAM immediately resets the LIVTIME value to the initial value and begins testing the EE connection for connectivity.

**Result:** When the EE LDLC Keep-Alive Reduction Function is enabled, there is a trade-off between CPU reduction and Enterprise Extender's responsiveness to TCP/IP connectivity problems. EE will verify the TCP/IP connection to the partner when HPR traffic resumes. However, if the EE partner is no longer reachable, there is an LDLC disconnect timeout period before the EE connection is disconnected. These trade-offs must be taken into account when selecting a `max_value` value for the LIVTIME facility.

The disconnect timer works independently of the LDLC timer, and determines the time interval during which Enterprise Extender monitors for data transfer before disconnecting the link (if no data transfer occurs).

An Enterprise Extender link might be disconnected under the following conditions:

- The interval determined by LIVTIME + (SRTIME*(SRRRETRY+1)) expires if required signal responses are not received.
- The DELAY value specified by DISCNT expires if no data transfer occurs.

**Note:** The Enterprise Extender link disconnection occurs if the DISCNT value defined specifies the connection is to be broken (see the DISCNT operand in “Switched major node” on page 551). In addition, if using DISCNT, the DELAY value in effect will be the minimum of the value on the dial-out PU and the dial-in PU. In order to accurately specify the DELAY value on the dial-in PU, you must be able to control which dial-in PU being is selected. See z/OS Communications Server: SNA Network Implementation Guide

MACADDR
MACADDR

Specifies the 12-digit hexadecimal medium access control (MAC) address for a peer processor on the local area network that this PU definition statement defines.

This operand is required for LAN and ATM LAN emulation connections. X'000000000000' is not a valid MACADDR value.

MAXSLOW

MAXSLOW=value1

Specifies the number of seconds that VTAM will allow a specific downstream PU to be in slowdown mode. When the PU is in slowdown mode, VTAM monitors when inbound data arrives from the PU, or when data is scheduled to be sent to the PU. If the PU is in slowdown longer than the MAXSLOW value1, the link station (PU) will be INOPed and the connection is cleaned up. If no data is being sent or received over this connection, the PU will not be INOPed for exceeding the MAXSLOW value1. This function requires the flow of inbound or outbound data to initiate the slowdown checks. If the PU is in slowdown and no data is being sent or received, the PU will remain hung. However, no data will be accumulating on this PU.

The allowed values are 0 or 10–255 seconds. When the value defaults to 0 or is coded as 0, the function is disabled.

Result: When an XCA controller initiates slowdown for a specific PU (Flow Off), only this specific PU and its associated sessions are affected.

Tip: When multiple XCA major nodes define and share the same XCA subchannel (CUADDR), the MAXSLOW value1 of the first XCA major node to allocate the XCA subchannel will be used.

MAXSLOW=(value1,value2)

Specifies a threshold value, in seconds, that an XCA subchannel has been in slowdown. When inbound or outbound data for an XCA port is being processed and the subchannel has been in slowdown longer than this threshold, message IST1886I will be issued for this condition. The allowed values are 0 or 10–255 seconds. When the value is coded as 0, the function is disabled. The default is 180 seconds.

Result: When an XCA controller initiates slowdown at the subchannel level, all downstream PUs and their associated sessions are affected by the slowdown mode.
Tip: When multiple XCA major nodes define and share the same XCA subchannel (CUADDR), the MAXSLOW value of the first XCA major node to allocate the XCA subchannel will be used.

**MEDIUM**

For LAN and ATM LAN emulation, connections:

- MEDIUM
  - BOXMGR
  - CSMACD
  - FDDI
  - RING

<table>
<thead>
<tr>
<th>statements:</th>
<th>PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies:</td>
<td>ADAPNO, CUADDR</td>
</tr>
</tbody>
</table>

For ATM native connections:

- MEDIUM
  - ATM

<table>
<thead>
<tr>
<th>statements:</th>
<th>PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependency:</td>
<td>PORTNAME</td>
</tr>
</tbody>
</table>

For Enterprise Extender connections:

- MEDIUM
  - HPRIP
For Enterprise Extender connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependency:</td>
<td>DIAL=NO</td>
</tr>
</tbody>
</table>

Identifies the type of shared access transport facility (SATF) being defined.

For LAN and ATM LAN emulation connections, this operand is required and the valid values are RING, CSMACD, FDDI, and BOXMGR.

For Enterprise Extender connections, the only valid option is HPRIP.

For ATM native connections, ATM is the only valid value and is the default value.

**MEDIUM=ATM**
- Defines a connection that provides native access to an ATM network through an Open Systems Adapter.

**MEDIUM=BOXMGR**
- Defines a box manager. A box manager allows an IBM 3172 Nways Interconnect Controller or an Open Systems Adapter to have minimal network management. It also allows generic alerts to flow from an IBM 3172 Nways Interconnect Controller or an Open Systems Adapter to the NetView program.

The IBM 3172 Nways interconnect controller tries to maintain continuous availability of the SSCP-PU session when there is an external communication adapter major node configured for network management (MEDIUM=BOXMGR). The BOXMGR session ends only when either VTAM terminates or the last external communication major node associated with the IBM 3172 Nways interconnect controller comes down. If the connection is terminated for any other reason, the IBM 3172 Nways interconnect controller attempts to reestablish the connection.

MEDIUM=BOXMGR sets MEDIUM=RING and ADAPNO=255 on the PORT definition statement. It also sets DIAL=YES on the GROUP definition statement and sets CUADDR to the value you specify on this PORT definition statement.

When MEDIUM=BOXMGR, only one GROUP, LINE, and PU definition statement is allowed. Also, the only operands that can be coded for this value are ISTATUS and SPAN. ISTATUS and SPAN are optional.

The MACADDR value for this option is specified on the DIALNO operand of the PATH definition statement of the switched major node.

**Note:** Because there are no LUs in the external communication adapter major node, there should be no LU definition statements in the switched major node. If there are LU definition statements under the PATH definition statement, errors occur when you attempt to activate the LUs.

See z/OS Communications Server: SNA Network Implementation Guide for an example of coding a box manager definition.

**MEDIUM=CSMACD**
- Defines a connection to a CSMA/CD 802.3 LAN

**MEDIUM=FDDI**
- Defines a connection to a FDDI LAN
MEDIUM=HPRIP
Defines a connection that provides access to resources located through an IP network through Enterprise Extender.

Note: On the PORT statement, only one XCA major node with MEDIUM=HPRIP may be active at any time.

MEDIUM=RING
Defines a connection to a token-ring network

MTU

\[ \text{MTU} = \text{mtu\_size} \]

statements: GROUP
dependencies: Enterprise Extender connection network connections only
range: 1024 - 32767

Defines the maximum transmission unit (MTU), in bytes, that is used for Enterprise Extender connections associated with the connection network (CN) that is defined by this group. This MTU size is reduced by VTAM to account for header lengths such as the IP, UDP, and LLC headers that are necessary for EE traffic. If MTU is not specified, VTAM uses the value learned from the TCP/IP stack. This learned value is the minimum MTU size (minus IP and UDP header sizes) associated with all the defined TCP/IP interfaces that are allowed for this local static VIPA. If an MTU size with a value is specified, VTAM uses the lesser of this \( \text{mtu\_size} \) value and the value learned from TCP/IP. If the TCP/IP stack presents a value less than 768 bytes, VTAM sets the MTU to 768 bytes because this is the smallest packet size allowed by the HPR architecture. Typically, the MTU size for an EE connection is constant after the EE connection is established. However, if the MTU size of the TCP/IP stack changes, RTP pipes with endpoints in the same node as the TCP/IP stack dynamically detect these changes when their outbound packets are being transmitted. If the learned value is less than the \( \text{mtu\_size} \) value, VTAM uses the learned value. Reasons for MTU size changes include the following information:
• New IP routes with different local MTU sizes become available
• Existing IP routes become unavailable
• Path MTU discovery is enabled for IPv4 or IPv6 EE connections (see "PMTUD start option" on page 911 for more information), and path MTU changes are discovered in the IP network

You can enable path MTU discovery for IPv4 and IPv6 EE connections (see "PMTUD start option" on page 911 for more information). If path MTU discovery is enabled, VTAM can dynamically learn about any MTU size changes that occur in the IP network. RTP pipes traversing Enterprise Extender connections can reduce their packet size to avoid IP fragmentation. In some environments, you might not be able to use path MTU discovery because firewalls must allow ICMP messages to be forwarded back to the host. In these cases, specify the MTU operand to control the maximum packet size that VTAM sends out on an Enterprise Extender connection.

Result: If a predefined EE connection is established between two EE endpoints before the VRN connection is established (using the same VIPA addresses), the
VRN connection uses the established predefined EE connection. In this case, to control the MTU size associated with the EE VRN, you must specify the MTU operand for the predefined EE connection (for example, on the switched PU definition).

**Rule:** When the MTU operand is specified on the GROUP statement, VNNAME and VNTYPE must also be specified on the GROUP statement. If the group is not defined as a connection network, the MTU operand cannot be specified on the GROUP definition statement.

**Tips:**
- If the `mtu_size` value is specified and IPv4 Enterprise Extender packets are being fragmented in the IP network, you might need to reduce the `mtu_size` value to account for optional headers that might be present.
- For dial-in Enterprise Extender connections with associated PUs that are dynamically created, you can define the MTU parameter in the model major node (DYNTYPE=EE) PU definition statement.
- For predefined Enterprise Extender connections, you can define the MTU parameter on the PU definition in the switched major node.

**MULTGSUP**

Specifies whether multiple-link TGs are supported on this connection. MULTGSUP=YES are for most configurations.

**MULTGSUP=Yes**

Specifies the XIDs received from the connecting node must have the XID2 Multiple-link TG support bit turned ON, meaning Multiple-link TGs are supported. In addition, VR TG sequence number checking on inbound PIUs will be performed. MULTGSUP=YES is the default.

**MULTGSUP=No**

Specifies that the XIDs received from the connecting node will be allowed to have the XID2 Multiple-link TG support bit OFF, meaning Multiple-link TGs are not supported. In addition, VR TG sequence number checking on inbound PIUs will not be performed.

**Note:** If not coded correctly, Subarea XCA connections could fail at activation. This parameter is introduced to support Subarea XCA connections which traverse mediums that do not support Multiple-link TGs. The specific configuration which required this parameter follows:

```
VTAM A/OSA ===> tkr=== 2210 ---X.25 PSDN--- NPSI/NCP === VTAM B
```

In this case, the NPSI/NCP X.25 connection does not support Multiple-link TGs.
name

statements: VBUILD, PORT, GROUP, LINE, PU
dependencies: none

On VBUILD:
name provides the required and unique name for this major node.

On PORT:
name provides the required name associated with a particular adapter in an IBM 3172 Nways interconnect controller with an OSA card or virtual representation of an Enterprise Extender connection.

On GROUP:
name provides the required minor node name of the line group represented by this definition statement.

On LINE:
name provides the required minor node name of the link represented by this definition statement.

On PU:
name provides the required minor node name of the physical unit represented by this definition statement.

NATIVE
Specifies whether this link station represents a connection to a native node.

If NATIVE is not specified on the ADJCP statement or on the PU definition statement, the two nodes negotiate their subnet affiliation during connection establishment.

- If the NETIDs match, the connection defaults to a native connection.
- If the NETIDs are different, the connection defaults to a nonnative connection.

No subnet negotiations will occur if NATIVE has been specified; in this case the value between the two nodes must match for connection to occur.

If you code NATIVE on both the ADJCP and PU definition statements, the values must match.

For transmission groups (TGs) between nodes, the value specified or negotiated for NATIVE must match. If a TG is already active, subsequent TG activations must not have a specified or negotiated value for NATIVE that conflicts with the value of NATIVE for the active TG. If there is a conflict, the TG attempting activation is not allowed to become active. If the activation is for a virtual-route-based TG, then the SSCP-SSCP session will also fail.

**NETID**
Specifies a 1–8 character network identifier. This operand is used in conjunction with the XNETALS start option in support of nonnative network connection. See "XNETALS start option" on page 978 for information about XNETALS.

**Note:** Use of the national character # is discouraged because this character might not be on keyboards produced worldwide. Because # might not be on all keyboards, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

```
NETID=network_id
```

If you specify NETID on the PU definition statement, VTAM ensures that when the physical unit is active, the connecting resource is within the network specified by NETID. The network ID specified by NETID is the CP NETID of the NETID where the PU resides. If you omit NETID, VTAM dynamically acknowledges the network ID during connection establishment and this network ID becomes the CP NETID. If you code NETID, and nonnative network connection is not supported, the definition is rejected unless the operand specifies VTAM’s network ID.

```
NETID=(,NOXNETLS)
```

Specifies that VTAM does not allow nonnative network attachment for LU-LU sessions for this PU. When NETID=(,NOXNETLS), VTAM indicates to the NCP that this PU will use VTAM’s native network ID.

LU-LU sessions might fail if the network identifier sent in the bind from the LU is different than the network identifier specified in VTAM, even though PU-PU connections can occur. If no network identifier is included in the bind, or if the network identifier is the same as that specified by VTAM, the LU-LU session should activate.
NN = NO

Specifies that the adjacent node is expected to be an end node.

NN = YES

Specifies that the adjacent node is expected to be a network node.

Restriction: In a peripheral subnetwork boundary configuration between a border node and a network node without border node function, the border node appears to the non-border node as an end node. In this case, coding NN=YES on the definition statement that represents the border node, a connection failure will occur. Avoid the failure by not coding the NN operand or by coding NN=NO.

**PDELAY**

<table>
<thead>
<tr>
<th>For LAN, ATM LAN emulation, or Enterprise Extender connections:</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDELAY = TERRESTR</td>
<td>PDELAY = NEGLIGIB</td>
</tr>
<tr>
<td>PDELAY = LONG</td>
<td>PDELAY = NEGLIGIB</td>
</tr>
<tr>
<td>PDELAY = NEGLIGIB</td>
<td>PDELAY = NEGLIGIB</td>
</tr>
<tr>
<td>PDELAY = PACKET</td>
<td>PDELAY = TERRESTR</td>
</tr>
<tr>
<td>PDELAY = TERRESTR</td>
<td></td>
</tr>
</tbody>
</table>
For LAN, ATM LAN emulation, or Enterprise Extender connections:

<table>
<thead>
<tr>
<th>PORT, GROUP (TGs to Enterprise Extender connection networks)</th>
<th>GROUP (TGs to connection networks)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TGs to connection networks</td>
<td>PU (TGs over PVCs)</td>
</tr>
<tr>
<td>For GROUP; MEDIUM=HPRIP</td>
<td>DIAL=YES (TGs to connection networks)</td>
</tr>
</tbody>
</table>

For ATM native connections:

<table>
<thead>
<tr>
<th>GROUP (TGs to connection networks)</th>
<th>dependency: DIAL=NO (TGs over PVCs)</th>
</tr>
</thead>
</table>

specifies the maximum propagation delay of the link for the TG. Propagation delay represents the time needed for a signal to travel from one end of the link to the other.

**PDELAY=NEGLIGIB**
Indicates a local area network delay (less than .48 milliseconds).

**PDELAY=TERRESTR**
Indicates telephone network delay (between .48 and 49.152 milliseconds).

**PDELAY=PACKET**
Indicates a packet-switched network delay (between 49.152 and 245.76 milliseconds).

**PDELAY=LONG**
Indicates a satellite delay (greater than 245.76 milliseconds).

**Recommendations:**

- When specified on the PORT statement, VNGROUP and VNNAME (or VNTYPE=GLOBAL) must also be specified on the PORT statement and the value applies to the TG to the connection network specified in VNNAME (or IP.IP if VNNAME is defaulted for VNTYPE=GLOBAL).

- When specified on the GROUP statement, VNNAME or VNTYPE=GLOBAL must also be specified on the GROUP statement. If the GROUP is not defined as a connection network, the TG characteristics cannot be specified on the GROUP definition statement. They must be specified on the PU definition statement in the switched major node (or for dynamically created dial-in EE PUs on the DYNTYPE=EE model major node) for the PU being dialed to establish the connection.

**Guideline:** Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT definition statement, you should define your Enterprise Extender connection network on the corresponding GROUP definition statement instead. By coding all connection network and TG characteristics operands on the GROUP statement, the VNGROUP operand is no longer needed. Also, because there can be multiple connection networks defined in an Enterprise Extender XCA major node with multiple GROUP definition statements, defining all of the operands on the GROUP statement to which they apply provides more clarity.

**PORTNAME**
PORTNAME—*(port_name)*

- statements: PORT
- dependencies: ATM native connections only (MEDIUM=ATM)

Specifies the name of a port on an Open Systems Adapter through which an ATM native connection is made.

The maximum length for PORTNAME is 8 characters.

This name must match the portname specified at OSA customization through OSA/SF. This name must also match the portname defined on a TRLE statement in a Transport resource list (TRL) major node for this OSA ATM native connection.

**Note:** PORTNAME is mutually exclusive with the CUADDR and ADAPNO operands for any single PORT definition statement.

**PUTYPE**

<table>
<thead>
<tr>
<th>For LAN and ATM LAN emulation connections:</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUTYPE = 5</td>
<td>PUTYPE = 2</td>
</tr>
<tr>
<td>PUTYPE = 4</td>
<td></td>
</tr>
</tbody>
</table>

For LAN and ATM LAN emulation connections, PUTYPE can be 5 or 4. For ATM native connections, PUTYPE must be 2.
For LAN and ATM LAN emulation connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP, LINE, PU (subarea node) USER=VCNS</td>
<td>PU statements:</td>
</tr>
<tr>
<td>dependency:</td>
<td>dependency:</td>
</tr>
<tr>
<td>USER=VCNS</td>
<td>TGs over PVCs, DIAL=NO</td>
</tr>
</tbody>
</table>

LAN and ATM LAN Emulation Connections:

Specifies the PU type.

Note:

1. If USER=VCNS is specified on the LINE definition statement, the PUTYPE operand is not valid.
2. If you have a virtual-route-based transmission group connection established between two hosts and HPR=YES is coded on the CDRM statements for the hosts, HPR may be used over this connection.

PUTYPE=4

Specifies a connection to an IBM 37xx Communication Controller running NCP/Token-Ring interconnection (NTRI) or a connection through a CSMA/CD 802.3 LAN to a 37xx Communication Controller running NCP. Specify PUTYPE=4 only if MEDIUM=RING or MEDIUM=CSMACD on the PORT definition statement.

PUTYPE=5

Specifies a connection to a host

ATM Native Connections:

Specifies the PU type.

Note: If USER=VCNS is specified on the LINE definition statement, the PUTYPE operand is not valid.

PUTYPE=2

Specifies a connection to a remote ATM PU through an Open Systems Adapter. PUTYPE=2 is the only valid value for ATM native connections and is the default value.

PVCNAME
**PVCNAME** = pvc_name

**LINE**

**statements:**

**dependencies:**

ATM native connections only (TGs over PVCs), DIAL=NO

Specifies the name of a permanent virtual channel (PVC) through which a remote node is connected.

PVCNAME is required for ATM native connections and must match the name of the PVC as it is defined during Open Systems Adapter configuration in the ATM Native Settings notebook for the ATM OSA-2 physical port. The ATM Native Settings notebook is part of the OSA/SF OS/2 interface.

**ROUTEDIR**

**INBOUND**

Indicates the direction of the route is from the device inbound to VTAM. The default is to set the RIF direction bit as an INBOUND direction.

**OUTBOUND**

Indicates it will flip the RIF direction bit prior to presenting the information to the Config Services XID exit. This will allow the XID exit to interpret the RIF field in an INBOUND or OUTBOUND direction.

This parameter is relevant to users of the VTAM Config Services XID exit, ISTEXCCS.

For LAN problem analysis, VTAM XCA (FID2) support provides the VTAM Config Services XID exit with the TokenRing Routing Information Field (RIF) of the received XID. The RIF is available in Subfield X’03’ within the CV57 (REQCONT RU).

**SAPADDR**
SAPADDR = 4

Connection type: | LAN and ATM LAN emulation | ATM native | Enterprise Extender (HPRIP) |
---|---|---|---|
statements: | PORT | PORT | PORT |
dependency: | PU (subarea node) | PU (TGs over PVCs) | PU |
range: | 4–252 | 4–252 | 4–252 |

For Enterprise Extender, there must be different SAP values for host connections. In other words, if two hosts are issuing dials, both in and out, they must be using different SAP values to represent the connection. For example, if host 1 issues a dial to host 2 with
local SAP = 04
remote SAP = 08

and the connection comes active, host 2 cannot issue a dial back to host 1 with
local SAP = 08
remote SAP = 04

because the connection represented by the SAP pair 0408 is already active. If host 2 issues the dial with the above information, the dial request will be failed and the operator will receive a sense code of FFC80004.

Note: See z/OS Communications Server: SNA Network Implementation Guide for more information about SAP pairs.

On PORT:
Specifies the service access point (SAP) address for the connection to a LAN, ATM or TCP/IP network through the port that this major node defines.

Note: Enterprise Extender is a connection identifier (there are no CUADDR and ADAPNO operands).

The SAP values for Enterprise Extender are multiples of 2 and for all other mediums a multiple of 4 applies. A null SAPADDR value has the same meaning as not coding the SAPADDR operand.

Note: Use the default for SAPADDR on the PORT definition statement. If there are multiple hosts sharing the XCA device, the SAPADDR must be unique for each VTAM that is sharing the XCA port.

Code the SAPADDR operand, in addition to the CUADDR and ADAPNO operands, to route information between the LAN and VTAM.

On PU:

Specifies the service access point (SAP) address for the remote node that this PU definition statement defines.
Note: This is not applicable for MEDIUM=HPRIP.

**SECURITY**

For LAN, ATM LAN emulation, or Enterprise Extender connections:

- **PORT, GROUP (TGs to Enterprise Extender connection networks)**
- **GROUP (TGs to connection networks)**
- **PU (TGs over PVCs)**
- **DIAL=NO (TGs over PVCs)**

For ATM native connections:

- **GROUP (TGs to connection networks)**
- **DIAL=YES (TGs to connection networks)**

<table>
<thead>
<tr>
<th>SECURITY</th>
<th>UNSECURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SECURITY</td>
<td>PUBLIC</td>
</tr>
<tr>
<td>SECURITY</td>
<td>UNDERGRO</td>
</tr>
<tr>
<td>SECURITY</td>
<td>SECURE</td>
</tr>
<tr>
<td>SECURITY</td>
<td>GUARDED</td>
</tr>
<tr>
<td>SECURITY</td>
<td>ENCRYPT</td>
</tr>
<tr>
<td>SECURITY</td>
<td>SHIELDED</td>
</tr>
</tbody>
</table>

Specifies the security level of the transmission group. The following options range from the least secure level to the most secure level.

- **SECURITY=UNSECURE**
  - Specifies no security level.

- **SECURITY=PUBLIC**
  - Specifies a public switched network.

- **SECURITY=UNDERGRO**
  - Specifies an underground cable, not guarded.

- **SECURITY=SECURE**
  - Specifies a secure conduit, not guarded.

- **SECURITY=GUARDED**
  - Specifies a guarded conduit, physical only.

- **SECURITY=ENCRYPT**
  - Specifies link encryption.

- **SECURITY=SHIELDED**
  - Specifies a guarded conduit, physical and radiation shielded.

**Recommendations:**

- When specified on the PORT statement, VNGROUP and VNNAME (or VNTYPE=GLOBAL) must also be specified on the PORT statement and the value applies to the TG to the connection network specified in VNNAME (or IPIP if VNNAME is defaulted for VNTYPE=GLOBAL).
- When specified on the GROUP statement, VNNAME or VNTYPE=GLOBAL must also be specified on the GROUP statement. If the GROUP is not defined as
a connection network, the TG characteristics cannot be specified on the GROUP
definition statement. They must be specified on the PU definition statement in
the switched major node (or for dynamically created dial-in EE PUs on the
DYNTYPE=EE model major node) for the PU being dialed to establish the
connection.

**Guideline:** Although an Enterprise Extender connection network, including the
virtual node name, type, and all TG characteristics, can be defined on the PORT
definition statement, you should define your Enterprise Extender connection
network on the corresponding GROUP definition statement instead. By coding all
connection network and TG characteristics operands on the GROUP statement, the
VNGROUP operand is no longer needed. Also, because there can be multiple
connection networks defined in an Enterprise Extender XCA major node with
multiple GROUP definition statements, defining all of the operands on the GROUP
statement to which they apply provides more clarity.

### SPAN

```
<SPAN> span_name
```

*statements:* GROUP, LINE, PU
*dependencies:* NetView

Defines a span of control for VTAM minor node resources. Code this operand if
you are using the NetView program. For a full description, see the *IBM Tivoli
NetView for z/OS Installation: Configuring Additional Components.*

The NetView program checks the SPAN value, but VTAM ignores it.

### SRQRETRY
SRQRETRY

```
SRQRETRY = number_of_retries
```

**statements:** PORT, GROUP  
**dependencies:** MEDIUM=HPRIP  
**range:** 3-9

Specifies the number of times the short request timer is retried before the port becomes inoperative. For an explanation about using SRQRETRY externally, see “LIVTIME” on page 237.

SRQTIME

```
SRQTIME = time
```

**statements:** PORT, GROUP  
**dependencies:** MEDIUM=HPRIP  
**range:** 3-20

Specifies the Enterprise Extender logical link control short request timer interval in seconds. For an explanation about using SRQTIME externally, see “LIVTIME” on page 237.

SUBAREA
Specifies the subarea address assigned to the adjacent domain's NCP or VTAM, whichever is represented by this PU definition statement. If you code SUBAREA, VTAM compares the subarea number on the XID frame with the value you code on this operand. If the values do not match, the connection ends.

The SUBAREA value must be the same as the subarea address defined for this physical unit in the adjacent domain.

**SWNORDER**

For LAN, ATM LAN emulation, or Enterprise Extender connections:

<table>
<thead>
<tr>
<th>statements</th>
<th>GROUP, LINE, Peripheral node</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies</td>
<td>Statements: GROUP, LINE, Peripheral node</td>
</tr>
</tbody>
</table>

For ATM native connections:

<table>
<thead>
<tr>
<th>statements</th>
<th>GROUP, LINE</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies</td>
<td>GROUP, LINE, TGs over SVCs, TGs to connection networks, DIAL=NO</td>
</tr>
</tbody>
</table>

Specifies the way VTAM locates a switched PU. If you do not code SWNORDER, the default is the value of the SWNORDER start option.

If VTAM does not find the PU using the SWNORDER parameter 1 value, and the second parameter of SWNORDER is FIRST, VTAM searches by the other value. For example, if you code SWNORDER=CPNAME and VTAM cannot find a PU with a matching CPNAME, VTAM searches by the station identifier (IDBLK and IDNUM operands on the PU definition statement for the switched major nodes).

If VTAM does not find the PU using the SWNORDER value you specify and the second parameter of SWNORDER is ONLY, the search for the PU terminates. For
example, if you code SWNORDER=(CPNAME,ONLY) and VTAM cannot find a PU with a matching CPNAME, no attempt will be made to find the PU using the station identifier.

If VTAM fails to find a predefined PU as described above and dynamic PUs are allowed, a dynamic PU will be created and PU connection processing will continue. If dynamic PUs are not allowed, the PU connection process will fail.

These operands override the SWNORDER start option values for SWNORDER=value_1,\_value_2:

\textit{value}_1=\textbf{CPNAME}

Specifies that VTAM searches for a switched PU by the CPNAME.

\textit{value}_1=\textbf{STATNID}

Specifies that VTAM searches for a switched PU by the station identifier.

\textit{value}_2=\textbf{FIRST}

If a switched PU is not found using the value specified by \textit{value}_1, the alternate value of \textit{value}_1 will be used to search for a switched PU.

\textit{value}_2=\textbf{ONLY}

If a switched PU is not found using the value specified by \textit{value}_1, a search using the alternate value of \textit{value}_1 will not be performed.

If either \textit{value}_1 or \textit{value}_2 are omitted from the SWNORDER definition, the corresponding start option SWNORDER value will be the default for the omitted value.

\textbf{TGN}

\begin{center}
\begin{tabular}{c|c}
\hline
For LAN and ATM LAN emulation connections: & For ATM native connections: \\
\hline
\includegraphics[width=0.4\textwidth]{tgn LAN ATM LAN.png} & \includegraphics[width=0.4\textwidth]{tgn ATM native.png} \\
\hline
\end{tabular}
\end{center}
For LAN and ATM LAN emulation connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>PU (subarea node)</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependency:</td>
<td>none</td>
<td>PUTYPE=2,</td>
</tr>
<tr>
<td>range:</td>
<td>1–255</td>
<td>CPNAME, NETID,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DIAL=NO</td>
</tr>
</tbody>
</table>

LAN and ATM LAN Emulation Connections:
Specifies the TG number that represents the connection between the host to which this PU definition statement applies and either a communication controller or another host.

Coding TGN=ANY allows the adjacent subarea node to accept whatever valid TG number it receives on the XID.

You can specify TGN=ANY only for the PU at one end of a transmission group. The PU at the other end must have a valid transmission group number or default to 1. An error occurs if you specify TGN=ANY for both PUs.

ATM Native Connections:
Specifies the TG number that represents the connection between this node and the destination node. This operand is valid when PUTYPE=2.

By specifying a tg_number, you indicate a preferred TG that might be used depending on how the partner nodes negotiate the number.

The CPNAME and NETID operands are required with the TGN operand. Note that you can specify the CPNAME operand without the TGN operand, and the TG number will be negotiated. However, if a value for TGN is specified, a value for CPNAME and NETID must also be specified.

TGN=tg_number
Specifies a decimal integer assigned to the transmission group.

For APPN, coding TGN=0 means that the PU does not support parallel TGs.

TGN=ANY
For subarea, allows the adjacent node to accept whatever valid transmission group number it receives on the XID. You can specify TGN=ANY for only one end of the transmission group. For the other end, specify a valid TGN number or let it default to 1. If you specify TGN=ANY for both ends, an error occurs.

For APPN, coding TGN=ANY, or not coding TGN, allows the transmission group number to be negotiated.

TGP
For LAN, ATM LAN emulation, or Enterprise Extender connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td>PORT, GROUP (TGs to Enterprise Extender connection networks)</td>
<td>GROUP (TGs to connection networks)</td>
</tr>
<tr>
<td>dependency:</td>
<td>DIAL=YES (TGs to connection networks)</td>
</tr>
<tr>
<td>TGs to connection networks</td>
<td>DIAL=NO (TGs over PVCs)</td>
</tr>
<tr>
<td>For GROUP; MEDIUM=HPRIP</td>
<td>PU (TGs over PVCs)</td>
</tr>
</tbody>
</table>

Specifies the name of a TG profile definition. The characteristics of the TG profile (along with any modifiers) become the characteristics of the PU. If TGP is not specified or has not been activated when the PU becomes active, default TG characteristics are assigned.

See “APPN transmission group profile” on page 720 for more information on transmission group profiles.

Recommendations:

- When specified on the PORT statement, VNGROUP and VNNAME (or VNTYPE=GLOBAL) must also be specified on the PORT statement and the value applies to the TG to the connection network specified in VNNAME (or IP/IP if VNNAME is defaulted for VNTYPE=GLOBAL).
- When specified on the GROUP statement, VNNAME or VNTYPE=GLOBAL must also be specified on the GROUP statement. If the GROUP is not defined as a connection network, the TG characteristics cannot be specified on the GROUP definition statement. They must be specified on the PU definition statement in the switched major node (or for dynamically created dial-in EE PUs on the DYNTYPE=EE model major node) for the PU being dialed to establish the connection.

Guideline: Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT definition statement, you should define your Enterprise Extender connection network on the corresponding GROUP definition statement instead. By coding all connection network and TG characteristics operands on the GROUP statement, the VNGROUP operand is no longer needed. Also, because there can be multiple connection networks defined in an Enterprise Extender XCA major node with multiple GROUP definition statements, defining all of the operands on the GROUP statement to which they apply provides more clarity.

**TIMER**
Specifies the number of seconds that VTAM will wait for a response after the channel is activated.

The TIMER operand is used for control-type commands. Data commands are timed by the link protocol. However, some control commands are also timed at the link level. As a result, a restriction exists: the calculation of the reply timer multiplied by one plus the number of retransmissions should be less than the value coded for TIMER.

**TOPO**

Controls the way APPN connections using this link station will be reported to APPN topology and routing services for inclusion into APPN functions.

**Note:** Except in the switched major node, the only value for TOPO is CONNECT. If you specify APPN on the CONNTYPE start option, it is not necessary to code TOPO, as you automatically get a value of CONNECT.

**TOPO=CONNECT**

APPN topology and routing services receives information about the connection and its characteristics when the connection is activated. When the connection is deactivated, APPN topology and routing services receives notification that the TG is inactive.

**TYPE**
Indicates an external communication adapter major node. This operand is required.

**UNRCHTIM**

\[ \text{UNRCHTIM} = \text{value of \_unrchtim\_start\_option\_value1} \]

\[ \text{UNRCHTIM} = \text{unreachable\_time} \]

**statements:** PORT, GROUP

**dependencies:** Enterprise Extender connections; VNNAME or VNTYPE

**range:** 0, or 10–65535

Specifies the default number of seconds that a partner node for a session path through an Enterprise Extender connection network is considered unreachable after certain Enterprise Extender connection network failures. During this period, the path through the Enterprise Extender connection network to this partner node will not be considered for new sessions or HPR path switches. A partner node is determined to be unreachable when a dial failure occurs across an Enterprise Extender connection network or when an INOP of an existing connection through an Enterprise Extender connection network is detected. UNRCHTIM is meaningful only if the NODETYPE start option is also used.

**Guidelines:**

- Coding UNRCHTIM for an Enterprise Extender connection network provides a way to route new sessions and HPR path switches around a problem with a specific connection network path, as long as an alternate route exists, until the underlying problem can be resolved. Specifying UNRCHTIM=0, or allowing it to default, will cause Enterprise Extender connection network processing to work the same as it does in z/OS V1R5 Communications Server and prior releases, but the topology and routing services component will not be informed of Enterprise Extender connection network connectivity problems and will continue to use the connection network path for new sessions and HPR path switches even if alternate paths are available. These new sessions and HPR path switches will continue to fail until the underlying problem is resolved. See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/docs/en/zos) for additional information about the implications of turning on UNRCHTIM for an Enterprise Extender connection network.

- Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT definition statement, you should define your Enterprise Extender connection network on the corresponding GROUP definition statement instead. By coding all connection network and TG characteristics operands on the GROUP statement, the VNGROUP operand is no longer needed. Also, because there can be multiple connection networks defined in an Enterprise Extender XCA major node with multiple GROUP definition statements, defining all of the operands on the GROUP statement to which they apply provides more clarity.
Tips:

- The *unreachable_time* value should be the same for all PORT or GROUP statements that define the same connection network. You can code a PORT statement and multiple GROUP statements with the same VNNAME if different IP addresses are specified. This allows you to have multiple TGs to that connection network. If the *unreachable_time* values are different on these statements, the *unreachable_time* value used will be the value associated with the line on which a dial failure or connection INOP occurs, causing the unreachable time to be unpredictable.

- An end node that has detected an unreachable partner path notifies its network node server (NNS) of the unreachability so that the NNS can use that information when calculating routes for the end node. Beginning with z/OS V1R8 Communications Server, the NNS overrides the *unreachable_time* value sent by the EN using its unreachable time value for the associated connection network, if configured, or the value specified in its UNRCHTIM VTAM start option. With this enhancement, changing the unreachable time value for an Enterprise Extender connection network no longer requires you to update every attached end node. Only the attached network nodes and those attached end nodes with NNS running z/OS V1R6 or V1R7 Communications Server need to be updated with the new value. You must still configure an unreachable time value on every end node to activate the EE connection network reachability awareness function.

**UPARM1**

For LAN, ATM LAN emulation, or Enterprise Extender connections:

| statements: | PORT, GROUP (TGs to Enterprise Extender connection networks) |
| dependency: | TGs to connection networks |
| range: | 0–255 |

For ATM native connections:

| statements: | GROUP (TGs to connection networks) |
| dependency: | PU (TGs over PVCs) |
| range: | 0–255 |

Specifies a user-defined operand value.

See z/OS Communications Server: SNA Network Implementation Guide for an example of using the UPARM operands.

**Recommendations:**

- When specified on the PORT statement, VNGROUP and VNNAME (or VNTYPE=GLOBAL) must also be specified on the PORT statement and the value applies to the TG to the connection network specified in VNNAME (or IPIP if VNNAME is defaulted for VNTYPE=GLOBAL).

- When specified on the GROUP statement, VNNAME or VNTYPE=GLOBAL must also be specified on the GROUP statement. If the GROUP is not defined as
a connection network, the TG characteristics cannot be specified on the GROUP
definition statement. They must be specified on the PU definition statement in
the switched major node (or for dynamically created dial-in EE PUs on the
DYNTYPE=EE model major node) for the PU being dialed to establish the
connection.

**Guideline:** Although an Enterprise Extender connection network, including the
virtual node name, type, and all TG characteristics, can be defined on the PORT
definition statement, you should define your Enterprise Extender connection
network on the corresponding GROUP definition statement instead. By coding all
connection network and TG characteristics operands on the GROUP statement, the
VNGROUP operand is no longer needed. Also, because there can be multiple
connection networks defined in an Enterprise Extender XCA major node with
multiple GROUP definition statements, defining all of the operands on the GROUP
statement to which they apply provides more clarity.

**UPARM2**

```
UPARM2 = 128
UPARM2 = user-defined_operand_value
```

<table>
<thead>
<tr>
<th>For LAN, ATM LAN emulation, or Enterprise Extender connections:</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>statements:</strong> PORT, GROUP (TGs to Enterprise Extender connection networks)</td>
<td><strong>statements:</strong> GROUP (TGs to connection networks)</td>
</tr>
<tr>
<td><strong>dependency:</strong> TGs to connection networks</td>
<td><strong>dependency:</strong> DIAL=YES (TGs to connection networks)</td>
</tr>
<tr>
<td>For GROUP; MEDIUM=HPRIP 0–255</td>
<td>DIAL=NO (TGs over PVCs) 0–255</td>
</tr>
</tbody>
</table>

Specifies a user-defined operand value.

**Recommendations:**

- When specified on the PORT statement, VNGROUP and VNNAME (or
  VNTYPE=GLOBAL) must also be specified on the PORT statement and the
  value applies to the TG to the connection network specified in VNNAME (or
  IP/IP if VNNAME is defaulted for VNTYPE=GLOBAL).
- When specified on the GROUP statement, VNNAME or VNTYPE=GLOBAL
  must also be specified on the GROUP statement. If the GROUP is not defined as
  a connection network, the TG characteristics cannot be specified on the GROUP
  definition statement. They must be specified on the PU definition statement in
  the switched major node (or for dynamically created dial-in EE PUs on the
  DYNTYPE=EE model major node) for the PU being dialed to establish the
  connection.

**Guideline:** Although an Enterprise Extender connection network, including the
virtual node name, type, and all TG characteristics, can be defined on the PORT
definition statement, you should define your Enterprise Extender connection
network on the corresponding GROUP definition statement instead. By coding all
connection network and TG characteristics operands on the GROUP statement, the VNGROUP operand is no longer needed. Also, because there can be multiple connection networks defined in an Enterprise Extender XCA major node with multiple GROUP definition statements, defining all of the operands on the GROUP statement to which they apply provides more clarity.

**UPARM3**

![UPARM3 Diagram]

<table>
<thead>
<tr>
<th>For LAN, ATM LAN emulation, or Enterprise Extender connections:</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>statements:</strong> PORT, GROUP (TGs to Enterprise Extender connection networks)</td>
<td><strong>statements:</strong> GROUP (TGs to connection networks)</td>
</tr>
<tr>
<td><strong>dependency:</strong> TGs to connection networks</td>
<td><strong>dependency:</strong> PU (TGs over PVCs) DIAL=YES (TGs to connection networks)</td>
</tr>
<tr>
<td><strong>range:</strong> For GROUP; MEDIUM=HPRIP 0–255</td>
<td><strong>range:</strong> DIAL=NO (TGs over PVCs) 0–255</td>
</tr>
</tbody>
</table>

Specifies a user-defined operand value.

**Recommendations:**
- When specified on the PORT statement, VNGROUP and VNNAME (or VNTYPE=GLOBAL) must also be specified on the PORT statement and the value applies to the TG to the connection network specified in VNNAME (or IP.IP if VNNAME is defaulted for VNTYPE=GLOBAL).
- When specified on the GROUP statement, VNNAME or VNTYPE=GLOBAL must also be specified on the GROUP statement. If the GROUP is not defined as a connection network, the TG characteristics cannot be specified on the GROUP definition statement. They must be specified on the PU definition statement in the switched major node (or for dynamically created dial-in EE PUs on the DYNTYPE=EE model major node) for the PU being dialed to establish the connection.

**Guideline:** Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT definition statement, you should define your Enterprise Extender connection network on the corresponding GROUP definition statement instead. By coding all connection network and TG characteristics operands on the GROUP statement, the VNGROUP operand is no longer needed. Also, because there can be multiple connection networks defined in an Enterprise Extender XCA major node with multiple GROUP definition statements, defining all of the operands on the GROUP statement to which they apply provides more clarity.

**USER**
For LAN and ATM LAN emulation connections:

<table>
<thead>
<tr>
<th>statements:</th>
<th>LINE</th>
<th>For ATM native connections:</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies:</td>
<td>Subarea node</td>
<td>statements: GROUP, LINE</td>
</tr>
</tbody>
</table>

Identifies the owner of the line. This operand is valid only when DIAL=NO is coded on the GROUP definition statement. This operand has two acceptable values:

**USER=SNA**

Specifies that the line uses SNA protocols to access the LAN.

**Note:** The MACADDR and SAPADDR operands on the accompanying PU definition statement provide the remote node’s address.

**USER=VCNS**

Specifies that VCNS uses the line to access the LAN.

When USER=VCNS, code only one LINE definition statement for each XCA major node.

**Note:**

1. If you code USER=VCNS on more than one definition statement, a warning message is issued and the value of USER defaults to SNA.
2. If you code USER=VCNS, do not code a PU definition statement for that LINE definition statement.

**VERALSID**
Notes:
1 VERALSID=NO is the only valid value if the TRLE operand is coded.

Statements: PU
Dependencies: ATM native connections only (TGs over PVCs), DIAL=NO

Specifies whether the adjacent link station name should be used to determine the link station. When specified as YES, the connection endpoints must supply the same name in the CV0E7 vector that is appended to XID3s. The name supplied in CV0E7 is the name of a switched PU or the name of a PU under a non-switched line.

**VNGROUP**

Notes:
1 Code VNGROUP if you code VNNAME (or VNTYPE=GLOBAL if you default the name to IP.IP)
2 VNNAME and VNTYPE can not be specified on the GROUP definition statement when that GROUP is specified on the VNGROUP operand on the PORT definition statement.

Statements: PORT, GROUP
Dependencies: LAN, ATM LAN emulation and Enterprise Extender connections only; VNNAME; DIAL=YES on GROUP statement

On GROUP, MEDIUM=HPRIP is required on the PORT statement

Specifies the name of the GROUP containing dial-out links available for use on the connection network named on the VNNAME operand.

For sessions that are routed through the connection network, a dynamic PU is created to represent the destination node. In effect, DYNPU=YES is enforced automatically when a session is established through the connection network. The name specified on the VNGROUP operand provides the name of the group for the dynamic PU that is allocated.

**Guideline:** Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT definition statement, you should define your Enterprise Extender connection network on the corresponding GROUP definition statement instead. By coding all
connection network and TG characteristics operands on the GROUP statement, the
VNGROUP operand is no longer needed. Also, because there can be multiple
collection networks defined in an Enterprise Extender XCA major node with
multiple GROUP definition statements, defining all of the operands on the GROUP
statement to which they apply provides more clarity.

**VNNAME**

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>VNNAME=connection_network_cpname</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Notes:**

1. On the PORT statement for all connections other than Enterprise Extender,
code VNNAME if you code VNGROUP.

2. On the PORT statement for Enterprise Extender connections, VNNAME is not
required if VNTYPE=GLOBAL, otherwise code VNNAME if you code
VNGROUP.

3. VNNAME is valid on the GROUP statement only if this is an Enterprise
Extender XCA major node (MEDIUM=HPRIP on the PORT statement).

**Statements:**

<table>
<thead>
<tr>
<th>PORT</th>
<th>GROUP</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAN, ATM LAN emulation, or Enterprise Extender connections only; VNGROUP; DIAL=YES on GROUP statement</td>
<td></td>
</tr>
<tr>
<td>Enterprise Extender connections; DIAL=YES on GROUP statement</td>
<td></td>
</tr>
</tbody>
</table>

**Dependencies:**

| On GROUP, MEDIUM=HPRIP is required on the PORT statement |

Specifies a 1–17 character network-qualified CPNAME for the connection network.
VNNAME is reported to the network topology as a virtual node and is treated as
an adjacent CP to this node. The NETID of the SSCP that owns the connection
network (the NETID of the host) is used.

For sessions that are routed through the connection network, a dynamic PU is
created to represent the destination node. In effect, DYNPU=YES is enforced
automatically when a session is established through the connection network.

VNNAME operand can be specified on the GROUP definition statement for
Enterprise Extender Connection Networks.

VNNAME operand will no longer be required for Enterprise Extender Connection
Networks when the VNTYPE is defined as GLOBAL. If no VNNAME is specified
for a Global Virtual Node (VNTYPE=GLOBAL), then a default name of IP.IP will
be used. If no Network ID (NETID) is specified for the VNNAME, then the NETID
of the host defining the Global Virtual Node is used. Therefore, if a user does not
want to use the default VNNAME, they should define their VNNAME as fully
qualified throughout their network (otherwise they risk non-matching VNNAMEs
and not being able to fully use their Global Virtual Routing Node support).

If the same VNNAME is defined more than once within the Enterprise Extender
XCA major node, the associated IPADDR value, HOSTNAME value, or IP address
resolved from the HOSTNAME value must be unique. A general rule is the
VNNAME/specified or resolved IP address pair must be unique.
If the same VNNAME is defined more than once within the Enterprise Extender XCA major node, the VNTYPE associated with this virtual routing node (VRN) must be the same.

**Note:** A user may define both Global and Local Virtual Nodes in the same XCA (MEDIUM=HPRIP) major node. This can be done by defining them on multiple GROUP definition statements. Alternatively, this can be done by coding one Virtual Node on the PORT and coding other Virtual Nodes on one or more GROUP statements. If a Virtual Node is being defined on the PORT statement, then it must specify a VNGROUP operand and the GROUP referenced cannot define another Virtual Node (either Local or Global). A user may define as many Local or Global Virtual Nodes as their configuration requires.

**Guideline:** Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT definition statement, you should define your Enterprise Extender connection network on the corresponding GROUP definition statement instead. By coding all connection network and TG characteristics operands on the GROUP statement, the VNGROUP operand is no longer needed. Also, because there can be multiple connection networks defined in an Enterprise Extender XCA major node with multiple GROUP definition statements, defining all of the operands on the GROUP statement to which they apply provides more clarity.

### VNTYPE

- **VNTYPE=LOCAL**
  - Connection network is not permitted to span multiple networks (default).

- **VNTYPE=GLOBAL**
  - Connection network is permitted to span multiple networks.

VNTYPE operand can be specified on the GROUP definition statement for Enterprise Extender Connection Networks. To define a Global VRN, the VNTYPE=GLOBAL must also be specified.

**Tip:** A VRN that is defined with VNTYPE=GLOBAL can also be used locally. Attempts to define the same VRN as GLOBAL in some nodes and LOCAL in others might not produce the results that you want. If a VRN is used across networks or subnetworks, it should always be defined with VNTYPE=GLOBAL in all nodes.

**Guideline:** Although an Enterprise Extender connection network, including the virtual node name, type, and all TG characteristics, can be defined on the PORT
definition statement, you should define your Enterprise Extender connection
network on the corresponding GROUP definition statement instead. By coding all
connection network and TG characteristics operands on the GROUP statement, the
VNGROUP operand is no longer needed. Also, because there can be multiple
connection networks defined in an Enterprise Extender XCA major node with
multiple GROUP definition statements, defining all of the operands on the GROUP
statement to which they apply provides more clarity.

VTAMTOPO

statements: VBUILD
dependencies: none

VTAMTOPO

statements: GROUP
dependencies: (switched logical lines only)
Specifies whether the VTAM topology agent reports the major nodes or subordinate minor nodes to the CMIP manager application programs for SNA local topology.

Specifying VTM.VTAMTOPO allows VTAMTOPO to be handled appropriately by the NCP Definition Facility for versions of NDF that do not recognize VTAMTOPO natively.

**VTAMTOPO=REPORT**

On **VBUILD**: Specifies that the VTAM topology agent reports this XCA major node and all subordinate minor nodes, including connected switched PUs under switched logical lines. This does not override any value specified on the GROUP or LINE.

On **GROUP**: Specifies that the VTAM topology agent reports all subordinate lines and connected switched PUs. This does not override any value specified on the LINE.

On **LINE**: Specifies that the VTAM topology agent reports this logical LINE.

**VTAMTOPO=NOREPORT**

On **VBUILD**: Specifies that the VTAM topology agent does not report the switched logical lines or connected switched PUs. The topology agent still reports the major node and the other subordinate resources. This does not override any value specified on the GROUP or LINE.

On **GROUP**: Specifies that the VTAM topology agent does not report any subordinate lines and connected switched PUs. This does not override any value specified on the LINE.

On **LINE**: Specifies that the VTAM topology agent does not report this logical LINE.

**VTAMTOPO=NOLLINES**

On **VBUILD**: Specifies that the VTAM topology agent does not report the switched logical lines. The topology agent still reports the major node and the other subordinate resources, including the connected switched PUs. This does not override any value specified on the GROUP or LINE.

On **GROUP**: Specifies that the VTAM topology agent does not report any subordinate switched logical lines. The topology agent still reports the connected switched PUs under the subordinate switched logical lines. This does not override any value specified on the LINE.

**VTAMTOPO=NOSWPUS**

On **VBUILD**: Specifies that the VTAM topology agent does not report the connected switched PUs. The topology agent still reports the major node and the other subordinate resources, including the switched logical lines. This does not override any value specified on the GROUP.
On **GROUP**: Specifies that the VTAM topology agent does not report any connected switched PUs under the subordinate switched logical lines. The topology agent still reports the subordinate switched logical lines.

**VTAMTOPO=INCLUDE**

On **VBUILD**: Specifies that the VTAM topology agent reports this XCA major node and all subordinate minor nodes. VTAMTOPO=INCLUDE overrides any other value specified on the GROUP or LINE.

**VTAMTOPO=IGNORE**

On **VBUILD**: Specifies that the VTAM topology agent does not report this XCA major node nor any subordinate minor nodes. VTAMTOPO=IGNORE overrides any other value specified on the GROUP or LINE.

**XID**

```
 XID—YES
```

*statements: PU*

*dependencies: ATM native connections only (TGs over PVCs), DIAL=NO*

Specifies that a PU type 2.1 channel contact procedure is to be used. XID=YES is the only valid value for ATM native connections and is the default value.

**XNETALS**
Determines whether the network ID of an attaching node or the VTAM network ID is used for an adjacent nonnative node. Used to control how a nonnative node can attach to VTAM.

**Note:** The XNETALS option has changed in the way VTAM treats PUs that send in non-native NETIDs in their XID. VTAM used to let the PU connect and treated it as if it sent in the same NETID that VTAM started with but that is no longer true. If the NETID that the PU sent in its XID is not the same as the VTAM ID and XNETALS=NO or is defaulted, IST6801 will be issued.

**XNETALS=NO**

Indicates that the VTAM network ID will be used to represent the connecting node’s network ID.

During connection:
- If the NETID operand is specified on the PU statement, then it must specify VTAM's network ID.
- If the NETID operand is not coded on the PU statement, then VTAM's network ID is assumed and the adjacent node's network ID is ignored.

During LU-LU session establishment:
- If the network ID for the originating LU is specified (during connection time) and is not identical to VTAM’s network ID, the session will not be established.
- If the originating LU is not using a network qualified name, VTAM’s network ID is assumed and the session will be established.

**Note:** If the PU is a dependent LU server or requester, nonnative network connection is not allowed if XNETALS=NO is specified and the NETID operand is not coded.

**XNETALS=YES**

Indicates that the network ID of the adjacent PU will be used for connection and session establishment.

During connection:
- If the NETID operand on the PU statement is coded, it will be used as the PU's network ID. In this case, the network ID specified on the PU statement must match the network ID specified by the attaching node during connection processing.
- If the NETID operand is not specified on the PU statement, then the network ID specified by the attaching node during connection processing will be assumed.

During LU-LU session establishment:
- If the network ID for the originating LU is specified and does not match the PU's network ID, the session will not be established.
• If the originating LU name is not a network qualified name, the session will be established. VTAM will qualify the LU's name with the attaching node's network ID.

**Note:** Providing that XNETALS=YES is coded, you can prohibit nonnative network attachment for LU-LU sessions for a particular PU by coding XNETALS=NO on the GROUP, LINE or PU statements.

**Local non-SNA major node**

To define a local non-SNA major node, code an LBUILD definition statement followed by a LOCAL definition statement. The definition statements define each channel-attached non-SNA terminal as part of a logical set (group) of channel-attached non-SNA terminals. The non-SNA cluster controller (3272 or compatible device) to which the terminal is attached is not defined to VTAM. All terminals defined within a local non-SNA major node do not need to be on the same controller. Different terminals on the same controller can be defined to VTAM in different local non-SNA major nodes.

**Table 27. Local non-SNA major node definition statements and operands**

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>LBUILD</td>
<td>CONFIGDS</td>
<td>277</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFGPW</td>
<td>278</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>LOCAL</td>
<td>CUADDR</td>
<td>278</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TERM</td>
<td>287</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASLENT</td>
<td>276</td>
<td></td>
<td></td>
<td>ASLTAB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASLTAB</td>
<td>276</td>
<td></td>
<td></td>
<td>ASLTAB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BUFNUM</td>
<td>277</td>
<td></td>
<td></td>
<td>ASLTAB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DLOGMOD</td>
<td>279</td>
<td></td>
<td></td>
<td>MODETAB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>FEATUR2</td>
<td>280</td>
<td></td>
<td></td>
<td>(NOEDATS, MODL1, NOPRINTR, NOSELPEN) MODETAB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISTATUS</td>
<td>281</td>
<td></td>
<td></td>
<td>ACTIVE</td>
<td>NetView or TSO</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOGAPPL</td>
<td>281</td>
<td></td>
<td></td>
<td></td>
<td>dependent LUs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOGTAB</td>
<td>282</td>
<td></td>
<td></td>
<td></td>
<td>dependent LUs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MDLENT</td>
<td>283</td>
<td></td>
<td></td>
<td></td>
<td>MDLTAB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MDLTAB</td>
<td>283</td>
<td></td>
<td></td>
<td></td>
<td>MDLTAB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODETAB</td>
<td>284</td>
<td></td>
<td></td>
<td>ISTINCLM</td>
<td>APPN only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>REGISTER</td>
<td>285</td>
<td></td>
<td></td>
<td>NETSRVR</td>
<td>NetView</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPAN</td>
<td>286</td>
<td></td>
<td></td>
<td></td>
<td>USS3270</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SSPFM</td>
<td>286</td>
<td></td>
<td></td>
<td></td>
<td>USS3270</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USSTAB</td>
<td>287</td>
<td></td>
<td></td>
<td>ISTINCDT</td>
<td>Dependent LUs</td>
</tr>
</tbody>
</table>

**Legend:**

| R | Required operand. |

**Local non-SNA major node coding notes**

This information contains coding notes for the local non-SNA major node.
**LBUILD definition statement**

Code one LBUILD definition statement for each logical group of channel-attached non-SNA terminals.

**LOCAL definition statement**

Code one LOCAL definition statement for each channel-attached non-SNA terminal in the VTAM domain.

**Local non-SNA major node full syntax**

The full syntax for the local non-SNA major node follows. The syntax for each operand is repeated in “Local non-SNA major node operand descriptions” on page 276.

**LBUILD**

```
LBUILD

name

CONFGS = configds_name
CONFGPW = configds_password
CONFGDS = configds_name

```

**LOCAL**

```
LOCAL

name

ASLENT = associated_lu_table_entry_name

ASLTAB = associated_lu_table_name
BUFNUM = number_IO_buffers
CUADDR = device_address
DLOGMOD = default_logon_mode_table_entry_name

FEATUR2 = ( MODEL1, NOEDATS, NOPRINTR, NOSELPEN )
FEATUR2 = ( )
FEATUR2 = ( MODEL2, EDATS, PRINTR, SELPEN )

ISTATUS = ACTIVE
ISTATUS = INACTIVE
LOGAPPL = controlling_primary_lu

LOGTAB = interpret_table_name
MDLENT = model_name_table_entry_name
MDLTAB = model_name_table_name
MODETAB = ISTINCLM
MODETAB = logon_mode_table_name
```
Local non-SNA major node operand descriptions

The following information describes the operands you can code for the local non-SNA major node.

**ASLENT**

```
statements: LOCAL
dependencies: ASLTAB
```

Specifies the name of the associated LU table entry to be used for this logical unit.

If you code the ASLENT operand, you must also code the ASLTAB operand. If you omit ASLENT and include ASLTAB, VTAM uses the first entry in the associated LU table by default.

**ASLTAB**
**ASLTAB**

Specifies the name of the associated LU table to be used for the logical unit. See "Associated LU table" on page 981 for a description of how to define an associated LU table.

If you omit the ASLTAB operand, VTAM will not provide the names of associated logical units to the application during LU-LU session initiation. However, you can provide the associated LU names in other ways. You can enter them from the terminal on the PRINTER1 and PRINTER2 operands of the LOGON command, or VTAM can send them to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the PRINTER1 and PRINTER2 operands of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace an associated LU table (ASLTAB). See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**BUFFNUM**

Specifies the number of IO buffers to be read from devices that are not properly emulating 3x74 type devices. BUFFNUM helps ensure that the screen is fully read when VTAM issues READ and READ modifies to the device.

BUFFNUM should be set so that BUFFNUM*IOBUF_size is large enough to get all of the data from a screen.

**CONFGDS**
CONFGDS

CONFGDS = _confgds_name_

statements: LBUILD
dependencies: none

Specifies a 1–8 character data definition name that identifies the configuration restart data set defined by the user for this major node. Include a DD statement that has this data definition name in the VTAM start procedure.

CONFGPW

CONFGPW = _confgds_password_

statements: LBUILD
dependencies: CONFGDS

Specifies the 1–7 character alphanumeric password that VTAM uses to access the configuration restart data set. If you do not code CONFGPW, but VSAM requires it, VSAM prompts the VTAM operator for the correct password when VTAM attempts to open the data set.

Code CONFGPW only if you also code CONFGDS.

CUADDR
defines the 3-digit or 4-digit hexadecimal device number used for this channel-attached terminal. The address must not be enclosed in quotation marks or apostrophes. For example, a channel device address of X'080' is coded as CUADDR=080.

Restriction: In addition to the resource definition with the name taken from the LOCAL statement's name field, VTAM creates an internal resource definition representing this local non-SNA terminal device when it is activated. This internal direct attachment node (DAN) definition uses the device address specified on the CUADDR operand as its name. No other definition in the same VTAM domain can use the value specified by the device_address value as its name, or activation failure will occur.

For the same reason, a local non-SNA terminal should not be defined to and activated by VTAM if its device number is defined as an MVS console and allocated to console services. Activating a local non-SNA terminal whose device number is in use by console services can cause VTAM, console services, or both to abend.

DLOGMOD
DLOGMOD = _default_logon_mode_table_entry_name_

Specifies the name of the default logon mode table entry used by default, if one is not otherwise provided. If you do not code this operand and the name of a logon mode table entry is not otherwise provided, VTAM uses the IBM-supplied session parameters for BSC 3270 terminals.

If you specify MODETAB, the entry must be in either the specified table or in ISTINCLM, an IBM-supplied logon mode table. If you do not specify MODETAB, the entry must be in ISTINCLM. For more information on logon mode entries, see "Logon mode table" on page 1026.

FEATUR2

Identifies the machine features for a channel-attached non-SNA terminal.

FEATUR2=DUALCSE
FEATUR2=LOWERCSE

Specifies how to send alphabetical characters coded with the TEXT operand on a USSMSG macroinstruction to a non-SNA terminal over the SSCP-LU session. This value does not affect non-alphabetical characters or any characters coded on the BUFFER operand of a USSMSG macroinstruction.

Code LOWERCSE to indicate that alphabetical characters are sent to the terminal over the SSCP-LU session in lowercase. Code DUALCSE to indicate that VTAM sends all characters as they are coded in the USSMSG macroinstruction.

FEATUR2=EDATS
FEATUR2=NOEDATS

Specifies whether this terminal has the extended data stream feature.

If you code EDATS, you must specify a logon mode entry that has a PSERVIC operand that indicates extended data streaming and that also contains other required device-specific and application-specific bit combinations. Refer to the device's component description manual and the application's programming manual for information on the specific bit setting to code.

FEATUR2=MODEL1
FEATUR2=MODEL2

Identifies the specific model number (Model 1 or 2) for this 3277, 3284, or 3286
terminal. Code MODEL1 for those devices that have a default screen or buffer size of 480 bytes. Code MODEL2 for those devices that have a default screen or buffer size of 1920 bytes.

These FEATUR2 values are available to an application program as part of the device characteristics pertaining to this channel-attached terminal. You can obtain these characteristics using the INQUIRE macroinstruction. For more information on the INQUIRE macroinstruction, see [z/OS Communications Server: SNA Programming](https://www.ibm.com/docs/en/zos/2.4.0?topic=features-supporting-communications-server).

**FEATUR2=NOPRINTR**
**FEATUR2=PRINTR**

Specifies whether the local device is a printer.

If you do not define a local non-SNA printer as a printer, raising the cover on the printer could end the session between the printer and the application. By defining it as a printer, you specify to VTAM that an asynchronous device-end from this device (caused by lifting the cover) is not to end the session.

See also the description of the VTAM start option ASYDE under [ASYDE start option](https://www.ibm.com/docs/en/zos/2.4.0?topic=start-asyde) on page 798.

**FEATUR2=NOSELPEN**
**FEATUR2=SELPEN**

Specifies whether this terminal supports the use of a selector pen.

**ISTATUS**

```
<table>
<thead>
<tr>
<th>ISTATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTIVE</td>
</tr>
<tr>
<td>INACTIVE</td>
</tr>
</tbody>
</table>
```

**ISTATUS**

Determines whether this terminal is to be activated after the first activation of the local non-SNA major node. The operator can override this value by specifying the SCOPE operand on the VARY ACT command when the major node is activated. See [z/OS Communications Server: SNA Operation](https://www.ibm.com/docs/en/zos/2.4.0?topic=features-supporting-communications-server) for information on using the VARY ACT command.

**ISTATUS=ACTIVE**

Specifies that the terminal becomes active when its local non-SNA major node becomes active.

**ISTATUS=INACTIVE**

Specifies that the terminal remains inactive until the operator activates it.

**LOGAPPL**
LOGAPPL = controlling_primary.lu

Identifies the name of the primary logical unit that this logical unit will automatically log on to as a secondary logical unit when this logical unit is activated. The name you specify can also be a USERVAR that will be resolved into its current value before being used to establish or redrive the session with the controlling primary logical unit.

The LOGAPPL operand cannot contain the name of an independent logical unit because independent logical units cannot be the controlling logical units for an automatic logon session.

To ensure retry of a pending automatic logon, code the AUTORTRY or AUTOTI start options.

You can specify a network-qualified name for LOGAPPL. You cannot specify an LU_ALIAS name for LOGAPPL.

**Note:** If you specify the NetView or TSO program as the application program on the LOGAPPL operand, you will get unpredictable results in recovery situations.

For more information on establishing sessions, see [z/OS Communications Server: SNA Network Implementation Guide](z/OS V2R1.0 Communications Server: SNA Resource Definition Reference).
LOGTAB = __interpret_table_name__

statements: LOCAL
dependencies: valid only for dependent logical units

Specifies the name of an interpret table that VTAM uses when processing logon requests originating from the logical unit defined by this LOCAL definition statement. VTAM uses the interpret table to interpret the name received from an initiation request. See “Interpret table” on page 1017 for a description of how to define an interpret table.

MDLENT

MDLENT = __model_name_table_entry_name__

statements: LOCAL
dependencies: MDLTAB

Specifies the name of the model name table entry to be used for this logical unit.

If you code the MDLENT operand, you must also code the MDLTAB operand. If you omit MDLENT and include MDLTAB, VTAM uses the first entry in the associated LU table by default.

MDLTAB
Specifies the name of the model name table to be used for the logical unit. See "Model name table" on page 1054 for a description of how to define a model name table.

The model name is passed only when VTAM initiates the session with the application program, for example, only when the LOGAPPL = parameter is also coded.

If you omit the LOGAPPL operand or the MDLTAB operand, VTAM will not provide the name of the model name table to the application during LU-LU session initiation. However, you can provide the model name in other ways. You can enter it from the terminal on the MODEL operand of the LOGON command, or VTAM can send it to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the MODEL operand of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace a model name table. See z/OS Communications Server: SNA Operation for more information on the MODIFY TABLE command.

MODETAB
Specifies the name of a logon mode table to be used for the logical unit. The name you code must be the name of a logon mode table created as described in "Logon mode table" on page 1026. If you do not supply a logon mode table for the logical unit on the MODETAB operand, an IBM-supplied default logon mode table (ISTINCLM) is used. If you specify a table, both the table you specify and the default table are used.

**name**

Assigns a name to the resource being defined by the definition statement.

**On LBUILD:**
name is optional. If a syntax or definition error is detected with this statement during activation of the major node, name appears in the message issued to identify the error.

The optional name you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node's status.

**On LOCAL:**
Provides the required minor node name assigned to the device whose address appears in the CUADDR operand in this LOCAL definition statement.

**REGISTER**
Specifies how a resource should be registered.

**REGISTER=CDSERVVR**
Indicates that an end node resource is registered to a network node server and directory resource registration is requested for it. A network node resource is registered at the central directory server. If the node is configured as a central directory server, this value has the same effect as NO.

**REGISTER=NETSRVR**
Indicates that the end node resource should be registered to its network node server, but that directory registration should not be requested for it.

**REGISTER=NO**
Indicates that the end node resource should not be registered.

**SPAN**

Defines a span of control for VTAM minor node resources. Code this operand if you are using the NetView program. For a full description, refer to the *IBM Tivoli NetView for z/OS Installation: Getting Started*.

The NetView program checks the SPAN value, but VTAM ignores it.
Determines the type of 3270 write control character (WCC) sent to the local non-SNA device for USS messages with TEXT= specified.

**SSCPFM=USS3270**

Sends a WCC with the start print bit on for USS messages with TEXT= specified on the USSMSG definition. If the device is a printer, these user-defined USS messages can be printed.

**SSCPFM=USS3275**

Sends a WCC with the start print bit off for USS messages with TEXT= specified on the USSMSG definition. If the device is a printer, these user-defined USS messages are not printed.

**TERM**

Specifies a specific, channel-attached non-SNA terminal (printer or display station).
USSTAB = ISTINCDT

Specifies the name of a USS table that VTAM uses to process character-coded input that it receives from the logical unit.

A terminal user can issue a USS command by coding the LANGTAB operand. This causes a second USS table to be associated with the logical unit, which overrides the table specified with USSTAB. If you do not code USSTAB and a LANGTAB USS table is not in use, the IBM-supplied USS table (ISTINCDT) is used. For more information on USS tables, see "Unformatted system services tables" on page 1069.

Local SNA major node

To define a local SNA major node, code a VBUILD definition statement for the major node and a separate PU and LU definition statement for each minor node.

See z/OS Communications Server: SNA Network Implementation Guide for a discussion about requirements for defining SNA and non-SNA devices.

Table 28. Local SNA major node definition statements and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=LOCAL</td>
<td>335</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFIGDS</td>
<td>302</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONFIGPW</td>
<td>302</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td>AUTHLEN</td>
<td>299</td>
<td></td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CAPACITY</td>
<td>299</td>
<td></td>
<td></td>
<td>APPN only; Type 2.1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONNTYPE</td>
<td>303</td>
<td></td>
<td></td>
<td>APPN only; NODETYPE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE</td>
<td>304</td>
<td></td>
<td>0</td>
<td>APPN only</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTTIME</td>
<td>304</td>
<td></td>
<td>0</td>
<td>APPN only</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CPCP</td>
<td>304</td>
<td></td>
<td></td>
<td>APPN only; CPCP start option</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CPNAME</td>
<td>305</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CUADDR</td>
<td>306</td>
<td></td>
<td>TRLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DELAY</td>
<td>306</td>
<td></td>
<td>0.200</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DISCNT</td>
<td>307</td>
<td></td>
<td>(NO)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DYNADJCP</td>
<td>308</td>
<td></td>
<td></td>
<td>APPN only; CPNAME</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DYNLU</td>
<td>309</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ENCR</td>
<td>311</td>
<td></td>
<td>P</td>
<td>NONE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ENCRTYPE</td>
<td>311</td>
<td></td>
<td>P</td>
<td>DES</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HPR</td>
<td>313</td>
<td></td>
<td>YES</td>
<td>APPN only</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Definition statement</td>
<td>Operands</td>
<td>Page</td>
<td>Req</td>
<td>Sift effect</td>
<td>Default</td>
<td>Restrictions, dependencies</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------</td>
<td>----------</td>
<td>------</td>
<td>-----</td>
<td>-------------</td>
<td>---------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>INCLUD0E</td>
<td></td>
<td></td>
<td>314</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISTATUS</td>
<td></td>
<td></td>
<td>314</td>
<td></td>
<td>ACTIVE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LIMRES</td>
<td></td>
<td></td>
<td>315</td>
<td></td>
<td>NO</td>
<td></td>
<td>LIMQSINT; LU 6.2; PUTYPE=1, 2</td>
</tr>
<tr>
<td>LUGROUP</td>
<td></td>
<td></td>
<td>319</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LUSEED</td>
<td></td>
<td></td>
<td>320</td>
<td></td>
<td></td>
<td></td>
<td>LUGROUP</td>
</tr>
<tr>
<td>MAXDATA</td>
<td></td>
<td></td>
<td>321</td>
<td></td>
<td></td>
<td></td>
<td>PU type 2</td>
</tr>
<tr>
<td>MAXBFRO</td>
<td></td>
<td></td>
<td>320</td>
<td></td>
<td></td>
<td>1 or 5</td>
<td>TRLE</td>
</tr>
<tr>
<td>NATIVE</td>
<td></td>
<td></td>
<td>324</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NETID</td>
<td></td>
<td></td>
<td>325</td>
<td></td>
<td></td>
<td></td>
<td>XNETALS</td>
</tr>
<tr>
<td>NN</td>
<td></td>
<td></td>
<td>326</td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>PDELAY</td>
<td></td>
<td></td>
<td>328</td>
<td></td>
<td>TERRESTR</td>
<td>2</td>
<td>APPN only</td>
</tr>
<tr>
<td>PUTYPE</td>
<td></td>
<td></td>
<td>328</td>
<td></td>
<td></td>
<td>2</td>
<td>APPN only</td>
</tr>
<tr>
<td>SECNET</td>
<td></td>
<td></td>
<td>330</td>
<td></td>
<td>NO</td>
<td>3174</td>
<td>3174 token ring</td>
</tr>
<tr>
<td>SECURITY</td>
<td></td>
<td></td>
<td>331</td>
<td></td>
<td>UNSECURE</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>SPAN</td>
<td></td>
<td></td>
<td>331</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>TGN</td>
<td></td>
<td></td>
<td>332</td>
<td></td>
<td>ANY</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>TGP</td>
<td></td>
<td></td>
<td>333</td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>TOPO</td>
<td></td>
<td></td>
<td>334</td>
<td></td>
<td>CONNECT</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>TRLE</td>
<td></td>
<td></td>
<td>334</td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CUADDR</td>
</tr>
<tr>
<td>UPARM1</td>
<td></td>
<td></td>
<td>335</td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>UPARM2</td>
<td></td>
<td></td>
<td>336</td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>UPARM3</td>
<td></td>
<td></td>
<td>336</td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>VERALSID</td>
<td></td>
<td></td>
<td>337</td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>XID</td>
<td></td>
<td></td>
<td>338</td>
<td></td>
<td></td>
<td>NO</td>
<td>APPN only</td>
</tr>
<tr>
<td>XNETALS</td>
<td></td>
<td></td>
<td>339</td>
<td></td>
<td></td>
<td></td>
<td>XNETALS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>start option value</td>
</tr>
<tr>
<td>name</td>
<td>LU</td>
<td>LOCADDR</td>
<td>316</td>
<td>R</td>
<td></td>
<td></td>
<td>see description</td>
</tr>
<tr>
<td>ASLENT</td>
<td></td>
<td>ASLTAB</td>
<td>298</td>
<td>P</td>
<td></td>
<td></td>
<td>ASLTAB</td>
</tr>
<tr>
<td>ASLTAB</td>
<td></td>
<td>ASLTAB</td>
<td>298</td>
<td>P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CERTIFY</td>
<td></td>
<td>CERTIFY</td>
<td>300</td>
<td>P</td>
<td>NO</td>
<td></td>
<td>ENCR</td>
</tr>
<tr>
<td>CKEYNAME</td>
<td></td>
<td>CKEYNAME</td>
<td>301</td>
<td>P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLRSESSQ</td>
<td></td>
<td>CLRSESSQ</td>
<td>301</td>
<td>P</td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLOGMOD</td>
<td></td>
<td>DLOGMOD</td>
<td>308</td>
<td>P</td>
<td></td>
<td></td>
<td>MODETAB</td>
</tr>
<tr>
<td>EAS</td>
<td></td>
<td>EAS</td>
<td>310</td>
<td>P</td>
<td>256</td>
<td>256</td>
<td>independent LUs only</td>
</tr>
<tr>
<td>ENCR</td>
<td></td>
<td>ENCR</td>
<td>311</td>
<td>P</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENCRTYPE</td>
<td></td>
<td>ENCRTYPE</td>
<td>311</td>
<td>P</td>
<td>DES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FASTPASS</td>
<td></td>
<td>FASTPASS</td>
<td>312</td>
<td>P</td>
<td>YES</td>
<td>TSO</td>
<td>TSO only</td>
</tr>
<tr>
<td>ISTATUS</td>
<td></td>
<td>ISTATUS</td>
<td>314</td>
<td></td>
<td></td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td>LOGAPPL</td>
<td></td>
<td>LOGAPPL</td>
<td>317</td>
<td>P</td>
<td></td>
<td></td>
<td>NetView or TSO</td>
</tr>
<tr>
<td>LOGTAB</td>
<td></td>
<td>LOGTAB</td>
<td>318</td>
<td>P</td>
<td></td>
<td></td>
<td>dependent LUs</td>
</tr>
<tr>
<td>LUDR</td>
<td></td>
<td>LUDR</td>
<td>319</td>
<td></td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDLENT</td>
<td></td>
<td>MDLENT</td>
<td>322</td>
<td>P</td>
<td></td>
<td></td>
<td>MDLTAB</td>
</tr>
</tbody>
</table>
### Table 28. Local SNA major node definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDLTAB</td>
<td>323</td>
<td>P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODETAB</td>
<td>323</td>
<td>P</td>
<td>ISTINCLM</td>
<td>ISTINCDT</td>
<td>dependent LUs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PACING</td>
<td>327</td>
<td>P</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGISTER</td>
<td>329</td>
<td>P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESETLU</td>
<td>330</td>
<td>YES</td>
<td>dependent LUs</td>
<td>dependent LUs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPAN</td>
<td>331</td>
<td></td>
<td>NetView</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSCPFM</td>
<td>332</td>
<td>P</td>
<td>FSS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USSTAB</td>
<td>337</td>
<td>P</td>
<td>ISTINCDT</td>
<td>dependent LUs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VPACING</td>
<td>338</td>
<td>P</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**
- **R** Required operand.
- **P** Code this operand on the higher-level PU definition statement to take advantage of the sift effect, or code it on this definition statement.

---

**Local SNA major node coding notes**

This information contains coding notes for the local SNA major node.

**VBUILD definition statement**

Code a VBUILD definition statement for each set of channel-attached SNA devices. One VBUILD definition statement, placed before the first PU definition statement, must be included for each major node.

**PU definition statement**

Code a PU definition statement for each physical unit, such as a cluster controller, in the local SNA major node. Note that a physical unit and all its logical units must be defined within a single major node.

**LU definition statement**

Code an LU definition statement for each logical unit associated with a physical unit within a local SNA major node. The LU definition statement must follow the PU definition statement that defines the physical unit with which this logical unit is associated.

**Dynamic change of operands for local SNA major node**

You can dynamically change the following local SNA major node operands by editing the VTAMLST and then issuing the VARY ACT,UPDATE=ALL command. You can specify the new operand value on a higher-level definition statement for sifting, if applicable.
Note: * You can dynamically change ISTATUS only on the LU definition statement.

To dynamically change the operand, the resource to which it applies must be inactive. For instance, an operand on an LU definition statement which receives its value from a PU definition statement can be dynamically changed only when the LU is inactive; the PU can remain active.

You can dynamically change the operands associated with tables (ASLTAB, LOGTAB, MDLTAB, MODETAB, and USSTAB) at any time without deactivating resources.

See z/OS Communications Server: SNA Network Implementation Guide and z/OS Communications Server: SNA Operation for more information about the VARY ACT,UPDATE=ALL command.

**NODETYPE start option**

Note that if you do not code the NODETYPE start option, any APPN operands you code in this major node are ignored.

The NODETYPE start option allows APPN function. The combination of NODETYPE and the HOSTSA start option determines the various configurations of subarea node, interchange node, migration data host, network node, or end node.

**Local SNA major node full syntax**

The full syntax for the local SNA major node follows. The syntax for each operand is repeated in “Local SNA major node operand descriptions” on page 298.

**VBUILD**
- \( \text{DELAY} = \text{timeout\_value} \)
- \( \text{DISCNT} = (\text{NO}) \)
- \( \text{DISCNT} \) values
- \( \text{DYNADJCP} = \text{value\_of\_dynadjcp\_start\_option} \)
- \( \text{DYNADJCP} = \text{YES} \) (4)
- \( \text{DYNLU} = \text{value\_of\_dynlu\_start\_option} \)
- \( \text{DYNLU} = \text{YES} \) (5)
- \( \text{ENCR} = \text{NONE} \)
- \( \text{ENCR} = \text{OPT} \)
- \( \text{ENCR} = \text{REQD} \)
- \( \text{ENCR} = \text{SEL} \)
- \( \text{ENCRTYPE} = \text{DES} \)
- \( \text{ENCRTYPE} = \text{TDES24} \)
- \( \text{HPR} = \text{YES} \)
- \( \text{HPR} = \text{NO} \)
- \( \text{INCLUD0E} = \text{NO} \)
- \( \text{ISTATUS} = \text{ACTIVE} \)
- \( \text{ISTATUS} = \text{INACTIVE} \)
- \( \text{LIMRES} = \text{NO} \)
- \( \text{LIMRES} = \text{YES} \) (6)
Notes:
1 CONNTYPE=APPN is the only valid value if the TRLE operand is coded.
2 CPCP defaults to YES for the model RTP PU.
3 CPCP defaults to the value of the CPCP start options for all PUs except the model RTP PU.
4 DYNADJCP=YES is the only valid value if the TRLE operand is coded.
5 DYNLU=YES is the only valid value if the TRLE operand is coded.
6 LIMRES=NO is the only valid value if the TRLE operand is coded.
7 MAXBFRU defaults to 1 when HPR=NO, XID=NO, CONNTYPE=LEN, or the VTAM node is not HPR capable.
8 MAXBFRU defaults to 5 when HPR=YES, XID=YES, CONNTYPE=APPN, and the VTAM node is HPR capable.
MAXDATA defaults to 261 when PUTYPE=1 is coded.
MAXDATA defaults to 265 when PUTYPE=2 is coded.
NEGLIGIB is the default if the TRLE operand is coded.
VERALSID=NO is the only valid value if the TRLE operand is coded.

LU

\[\text{name}\text{LU}\text{LOCADDR}\text{address}\]
\[\text{ASLENT}\text{associated\_lu\_table\_entry\_name}\]
\[\text{ASLTAB}\text{associated\_lu\_table\_name}\]
\[\begin{array}{c}
\text{CERTIFY}\text{NO} \\
\text{CERTIFY}\text{YES}
\end{array}\]
\[\begin{array}{c}
\text{CKEYNAME}\text{name} \\
\text{CKEYNAME}\text{crypto\_key\_name} \\
\text{CLRSESSQ}\text{NO} \\
\text{CLRSESSQ}\text{YES}
\end{array}\]
\[\text{DLOGMOD}\text{default\_logon\_mode\_table\_entry\_name}\]
\[\begin{array}{c}
\text{EAS}\text{256} \\
\text{EAS}\text{number\_of\_sessions} \\
\text{ENCR}\text{NONE} \\
\text{ENCR}\text{OPT} \\
\text{ENCR}\text{REQD} \\
\text{ENCR}\text{SEL}
\end{array}\]
Notes:
1  REGISTER defaults to NETSRVR for dependent LUs.
2  REGISTER defaults to NO for independent LUs.

Local SNA major node operand descriptions
The following information describes the operands you can code for the local SNA major node.

ASLENT

Specifies the name of the associated LU table entry to be used for this logical unit.

If you code the ASLENT operand, you must also code the ASLTAB operand. If you omit ASLENT and include ASLTAB, VTAM uses the first entry in the associated LU table by default.

ASLTAB
ASLTAB = associated_lu_table_name

**stages:** PU, LU

**dependencies:** none

Specifies the name of the associated LU table to be used for the logical unit. See "Associated LU table" on page 981 for a description of how to define an associated LU table.

If you omit the ASLTAB operand, VTAM will not provide the names of associated logical units to the application during LU-LU session initiation. However, you can provide the associated LU names in other ways. You can enter them from the terminal on the PRINTER1 and PRINTER2 operands of the LOGON command, or VTAM can send them to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the PRINTER1 and PRINTER2 operands of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace an associated LU table (ASLTAB). See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**AUTHLEN**

**stages:** PU

**dependencies:** Type 2.1; APPN only

Specifies whether VTAM should pass the transmission priority field specified by this PU to another PU.

**AUTHLEN=NO**

VTAM will not pass the specified transmission priority, instead the default priority of medium will be passed.

**AUTHLEN=YES**

VTAM will pass the specified transmission priority.

**CAPACITY**
CAPACITY = 32M

CAPACITY = number of bits

statements: PU
dependencies: APPN only
range:

1 - 1000Kb
1 - 1000Mb
1 - 100Gb

Specifies the effective capacity of the link that comprises the transmission group (TG). Specify the value in Kb per second (for example, 100K), Mb per second (for example, 100M), or Gb per second (for example, 10G). This number approximates the bits per second that the link can transmit (the transmission rate of the link, times the maximum load factor expressed as a percentage).

Note: Because the value for capacity is represented as a single byte, the precision of the specified number of bits might be lost. For example, numbers that are close (such as 100K and 101K) can be interpreted by VTAM, and displayed, as the same value. See Table 45 on page 722 for a list of CAPACITY values you can specify and their corresponding values when displayed.

CERTIFY
Specifies whether VTAM, on behalf of the application program, will verify authentication.

**CERTIFY=YES**
- Specifies SLU authentication (verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

**CERTIFY=NO**
- Specifies no SLU authentication (not verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

**CKEYNAME**

Specifies the name of the key-encrypting key (KEK) in the cryptographic key data set (CKDS) for the LU. Use CKEYNAME to encipher the session key that is sent in the BIND to the LU when a cryptographic session is established. CKEYNAME has the same format and restrictions as other VTAM names. For information on naming conventions, see “Format of definition statements” on page 8.

If you do not specify CKEYNAME, the cryptographic key name defaults to the name on the LU definition statement.

**Note:** Independent LUs will ignore the CKEYNAME keyword and use their default name.

**CLRSESSQ**
Specifies whether a dependent LU is forced into an inhibited session state when the LU is disabled. When the LU is enabled, it is returned to normal status. However, sessions are not connected.

**CLRSESSQ=NO**
- Specifies that the LU is not forced into inhibited session state.

**CLRSESSQ=YES**
- Specifies that the LU is forced into inhibited session state. Both active and pending sessions are removed from the session queue.

**CONFGDS**

Specifies a 1–8 character data definition name that identifies the configuration restart data set defined by the user for this major node. Include a DD statement that has this data definition name in the VTAM start procedure.
CSM590000/SM590000

CONFGPW = confgds_password

statements: VBUILD
dependencies: CONFGDS

Specifies the 1–7 character alphanumeric password that VTAM uses to access the configuration restart data set. If you do not code CONFGPW, but VSAM requires it, VSAM prompts the VTAM operator for the correct password when VTAM attempts to open the data set.

Code CONFGPW only if you also code CONFGDS.

CONNTYPE

CONNTYPE = value_of_conntype_start_option

Notes:
1 CONNTYPE=APPN is the only valid value if the TRLE operand is coded.

statements: PU
dependencies: NODETYPE; APPN only

Specifies for a type 2.1 whether the connection is to be established as a LEN connection or attempted as an APPN connection.

If you do not code the NODETYPE start option, the CONNTYPE operand is ignored, and the connection will be a LEN connection.

CONNTYPE=APPN

Specifies that this connection can support parallel TGs, CP-CP sessions, and CP name change support. When you code CONNTYPE=APPN, XID rules for parallel TGs are enforced. So, if you have PUs with duplicate CPNAMEs, you should make them unique before this node becomes an APPN node.

CONNTYPE=LEN

Specifies that this connection does not support CP-CP sessions and CP name change, and that VTAM does not check for duplicate CPNAMEs.

Because CONNTYPE=LEN indicates that the PU is not APPN-capable, do not code any APPN-only operands on the same PU statement. If an APPN-only operand is coded with CONNTYPE=LEN, VTAM will issue a message warning of a parameter conflict.

Although you need to code CONNTYPE=APPN for a PU to be an APPN resource, this does not necessarily mean the PU will be APPN; a PU’s characteristics are
determined at activation. However, coding CONNTYPE=LEN means that the PU can be a LEN resource only, even if, at activation, the XID indicates the PU can be an APPN resource.

**COSTBYTE**

```
| statements: | PU       |
| dependencies: | APPN only |
| range: | 0–255 |
```

Specifies a cost-per-byte-transmitted to be associated with the transmission group. A value of 0 is the least expensive cost per byte and 255 is the most expensive.

**COSTTIME**

```
| statements: | PU       |
| dependencies: | APPN only |
| range: | 0–255 |
```

Specifies a cost-per-unit-of-time to be associated with the transmission group. A value of 0 is the least expensive cost per unit of time and 255 is the most expensive.
Notes:
1 CPCP defaults to YES for the model RTP PU.
2 CPCP defaults to the value of the CPCP start options for all PUs except the model RTP PU.

Specifies whether CP-CP sessions are supported on this connection. If you do not code this operand, the value defaults to the value specified on the CPCP start option.

**CPCP=NO**
- Indicates CP-CP sessions are not supported on this connection.

**CPCP=YES**
- Indicates CP-CP sessions are supported on this connection.

**CPNAME**
Identifies the resource's owning control point name. To allow definition of casually
connected resources, this operand can be fully qualified with a network ID. If it is
later learned that the owning control point name is different from what is specified
here, the resource will be associated with the newly learned control point name.

**CUADDR**

Defines a 3-digit or 4-digit hexadecimal number that identifies the channel device
address used when activating the physical unit. The value must match a channel
device name supplied when the operating system is generated.

The address must not be enclosed in quotation marks or apostrophes. For example,
a channel device address of X'080' should be coded CUADDR=080.

**Note:** You cannot code CUADDR with the TRLE operand. However, if you omit
both the CUADDR and TRLE operands, you must code ISTATUS=INACTIVE and
include the address in the U operand in the VARY ACT command that activates
the physical unit.

**DELAY**
Specifies the maximum amount of time, in seconds and thousandths of seconds, that VTAM should wait before transmitting low-priority data (that is, PIU transmission priority 0 or 1) to the physical unit.

For more efficient operation, code this value as high as possible. For maximum speed across the interface, set this value to 0. See z/OS Communications Server: SNA Network Implementation Guide for more information.

**DISCNT**

- **DISCNT=(NO)**
  - If you code NO or use the default, VTAM terminates the SSCP-LU and SSCP-PU sessions when certain conditions are met. See z/OS Communications Server: SNA Network Implementation Guide for a discussion about conditions that can cause sessions to terminate.

- **DISCNT=(YES)**
  - If you code YES, VTAM automatically terminates the SSCP-LU and SSCP-PU sessions as soon as the last LU-LU session is terminated.

- **DISCNT=(YES,F)**
- **DISCNT=(YES,NF)**
  - Specifies whether VTAM indicates final-use status in the DACTPU request unit when it deactivates a physical unit as a result of DISCNT=YES. This operand does not apply when DISCNT=NO, nor does it have any effect on the VARY INACT command.
If you code F or use the default, final-use status is indicated and the physical connection can be ended. If you code NF, not-final-use status is indicated and the physical connection should not be broken.

Each device has its own requirements regarding final-use status. To determine whether to code F or NF for a given device, consult the appropriate installation publication for the device.

**DLOGMOD**

```plaintext
DLOGMOD — default_logon_mode_table_entry_name
```

*statements:*  
PU, LU

*dependencies:*  
MODETAB

Specifies the name of the default logon mode table entry used by default if one is not otherwise provided. If you do not code this operand and the name of a logon mode table entry is not otherwise provided, VTAM uses the first entry in the applicable logon mode table (specified on the MODETAB operand or used by default).

If you specify MODETAB, the entry must be in either the specified table or in ISTINCLM, an IBM-supplied logon mode table. If you do not specify MODETAB, the default entry name must be in ISTINCLM. For more information on logon mode entries see ["Logon mode table" on page 1026](#).

**DYNADJCP**
**Notes:**

1. **DYNADJCP=YES** is the only valid value if the TRLE operand is coded.

   **Statements:**
   
   **PU**

   **Dependencies:**
   
   APPN only; ignored if CPNAME coded

Specifies whether an ADJCP can be created dynamically for an adjacent node.

**Note:** This operand is ignored if CPNAME is coded on the physical unit.

**DYNADJCP=NO**

Indicates that the ADJCP cannot be created dynamically for an adjacent node.

**DYNADJCP=YES**

Indicates that the ADJCP can be created dynamically for an adjacent node.

**DYNLU**
Notes:

1. DYNLU=YES is the only valid value if the TRLE operand is coded.

Specifies whether dynamic allocation of CDRSC definitions is supported. If you do not code this operand, the default is the value you coded on the DYNLU start option.

If dynamic allocation of CDRSC definitions is not supported for a particular PU, a resource that was dynamically defined as a CDRSC cannot use that PU to connect to a session.

**DYNLU=NO**

Specifies that dynamic allocation of CDRSC definitions is not supported. If you code DYNLU=NO and plan to connect to other nodes using CP-CP sessions, then predefine each node in the CDRSC major node. If you do not, CP-CP session setup fails.

**DYNLU=YES**

Specifies that dynamic allocation of CDRSC definitions is supported.

For subarea nodes (HOSTSA is specified), DYNLU=YES is meaningful only when the host CDRM is activated and CDRDYN=YES is specified on the CDRM definition statement.

For APPN-only nodes (HOSTSA is not specified), DYNLU=YES is meaningful even though a host CDRM cannot be activated.

**Note:** If you have coded a DYNLU value in the adjacent CP major node for the adjacent CP to which this physical unit is linked, that value overrides the DYNLU value you code on this PU definition statement. For additional DYNLU considerations, see "DYNLU" on page 23 under the adjacent control point major node.

**EAS**
EAS = 256

EAS = number_of_sessions

Specifies the estimated number of sessions that will be active with this logical unit at any given time. This operand is valid only for an independent logical unit (LOCADDR=0). If you code EAS for other logical units, VTAM ignores it.

If your estimated value is greater than 256, specify the next higher multiple of 256 for better performance.

You can use this operand in conjunction with the BSBUF buffer pool start option to support peak session levels. See "Buffer Pool start option" on page 804 for information about BSBUF.

ENCR

ENCR = NONE
ENCR = OPT
ENCR = REQD
ENCR = SEL

Specifies whether this logical unit has any special requirements for enciphering and deciphering messages.

ENCR=NONE
    Specifies that this logical unit cannot engage in cryptographic sessions.

ENCR=OPT
    Specifies that this logical unit can engage in cryptographic sessions, but allows the application program to determine whether to use cryptography.

ENCR=REQD
    Specifies that VTAM must encipher all messages to and from this logical unit. If ENCR=REQD, no sessions can be established with this logical unit unless the host with which it is associated can handle cryptographic sessions.

ENCR=SEL
    For local SNA major nodes, SEL has no meaning for logical units. If you specify ENCR=SEL, ENCR=OPT is used instead.

ENCRTYPE
Specifies the minimum type of encryption that VTAM should use on behalf of the application when performing session level encryption.

**ENCRTYPE=DES**
- Specifies that VTAM must use a minimum of DES encryption with an 8-byte key when performing session level encryption. This is the default.

**ENCRTYPE=TDES24**
- Specifies that VTAM must use a minimum of Triple DES encryption with a 24-byte key when performing session level encryption.

**Note:** The ENCRTYPE value coded on the PU will be sifted down to the appropriate resource (LU).

The SYSDEFed ENCRTYPE of a LU can be changed at runtime through the MODIFY SECURITY command. See the [z/OS Communications Server: SNA Operation](https://zdocs.esm.com/ESM/Secure/ESM/ESM/CommServe/CommServe.html) for specific details. However, the ENCRTYPE value will not be able to be downgraded. In other words, if the application is currently operating at ENCRTYPE=TDES24, then the encryption level cannot be changed to ENCRTYPE=DES using the MODIFY SECURITY command.

**Note:** If downgrading the security level is preferred, the current major node being used must be deactivated and another major node value must be coded.

In addition, a user may use the LOGMODE TABLE to override the ENCRTYPE value of the SLU based on the logmode entry being used. The ENCRTYPE value will not be able to be downgraded from TDES24 to DES using the logmode table. Only TDES24 can be specified for the encryption level (ENCRTYP). See [“MODEENT macroinstruction” on page 1027](https://zdocs.esm.com/ESM/Secure/ESM/ESM/CommServe/CommServe.html) for more information about MODEENT macroinstructions.

**FASTPASS**
Determine how sessions are established for application programs that issue the \texttt{CLSDST} macroinstruction with the \texttt{PASS} option as part of their session establishment procedure. The application program acts as a secondary logical unit (SLU).

The PLU acknowledges the capability of the SLU during session establishment. If you have coded the logon performance enhancement in the PLU's application program, only a single \texttt{BIND} is issued to establish the session.

Some SLUs are incompatible with this enhancement. If your SLU is incompatible with this enhancement, code \texttt{FASTPASS=NO}.

Note: \texttt{FASTPASS} applies only to application programs (acting as SLUs) which establish sessions with TSO.

- \texttt{FASTPASS=NO} 
  Specifies that an unenhanced logon procedure requiring multiple binds is performed for application programs that issue the \texttt{CLSDST PASS} macroinstruction with the \texttt{PASS} option as part of their session establishment procedure.

- \texttt{FASTPASS=YES} 
  Specifies that the logon performance enhancement is performed for application programs that issue the \texttt{CLSDST macroinstruction with the PASS option as part of their session establishment procedure.}

\textbf{HPR}
Specifies whether High-Performance Routing (HPR) support is enabled for a PU.

**HPR=YES**
- Indicates that this PU provides HPR support.

**HPR=NO**
- Indicates that this PU does not provide HPR support.

**INCLUDE0E**

Specifies whether VTAM provides LU names to the logical units defined under this PU. The names are provided by passing a CV0E, Network Name control vector on the ACTLU requests.

**INCLUDE0E=NO**
- Indicates that VTAM will not provide LU names to the logical units defined under this PU.

**INCLUDE0E=YES**
- Indicates that VTAM will provide LU names to the logical units defined under this PU.

**ISTATUS**
Determines whether a physical unit or logical unit is to be activated after the first activation of the channel-attachment major node.

**ISTATUS=ACTIVE**
- Specifies that the physical unit or logical unit becomes active when its channel becomes active.

**ISTATUS=INACTIVE**
- Specifies that the physical unit or logical unit remains inactive until the operator activates it.

The operator can override this value by specifying the SCOPE operand on the VARY ACT command when the major node is activated. See the Communications Server: SNA Operation for information about using the VARY ACT command.

**Note:** You can dynamically change ISTATUS only on the LU definition statement.

**LIMRES**
Specifies whether this physical unit is to be treated as a limited resource. A limited resource allows you to limit the use of some network connections. It also causes any sessions that traverse the resource to be deactivated if no conversation is active. A session can also be deactivated if the time limit you specify for the LIMQSINT operand on the APPL definition statement expires.

This operand applies only to LU 6.2 conversations and PU types 1 and 2.

If you do not specify a value for LIMRES, resources are not considered to be limited resources. Furthermore, sessions will remain active even after all conversations have ended.

You can also use LIMRES in conjunction with the LIMINTCP start option to specify the interval to retain a free session using a limited resource for some network management flows (excluding flows using the APPN control point sessions or dependent LU server sessions). If you do not code a value for the LIMINTCP start option, resources are not considered to be limited resources, and the sessions will remain active even after all conversations have ended.

**LIMRES=NO**

Specifies that this physical unit is not to be treated as a limited resource.

**LIMRES=YES**

Specifies that this physical unit is to be treated as a limited resource.

**LOCADDR**
LOCADDR = address

**statements:**

LU

**dependencies:**

see notes

**range:**

0–63 (PU type 1)
0–255 (PU type 2)

Specifies the logical unit's local address at the physical unit.

For independent logical units, specify 0. For dependent logical units, specify a nonzero value. All dependent logical units under a given physical unit must have unique LOCADDR values.

An LU definition statement is not required for every possible local address, and LOCADDR values need not be consecutive. Unused local addresses smaller than the largest local address at a station are not assigned network resources.

**Note:**

1. The value of LOCADDR depends upon the requirements of the device being defined. Consult the appropriate component description manual for these restrictions.

2. Although you can use this operand to define independent LUs, it is recommended that you either use dynamic definition or redefine the independent LUs as CDRSCs.

**LOGAPPL**
LOGAPPL = controlling_primary_lu

statements: PU, LU
dependencies: NetView or TSO

Identifies the name of the primary logical unit that this logical unit will automatically log on to as a secondary logical unit when this logical unit is activated. The name you specify can also be a USERVAR that will be resolved into its current value before it is used to establish or redrive the session with the controlling primary logical unit.

For independent logical units, the LOGAPPL operand establishes a session with the primary logical unit designated on the LOGAPPL operand if a session does not currently exist with this logical unit. The session is established regardless of whether sessions with other logical units exist. The LOGAPPL operand cannot contain the name of an independent logical unit because independent logical units cannot be the controlling logical units for an automatic logon session.

To ensure retry of a pending automatic logon, code the AUTORTRY or AUTOTI start options.

You can specify a network-qualified name for LOGAPPL. You cannot specify an LUALIAS name for LOGAPPL.

Note: If you specify the NetView or TSO program as the application program on the LOGAPPL operand, you will get unpredictable results in recovery situations.

For more information about establishing sessions, see z/OS Communications Server: SNA Network Implementation Guide.

LOGTAB
**LOGTAB**

Specifies the name of an interpret table that VTAM uses when processing logon requests originating from the logical unit. VTAM uses the interpret table to interpret the name received from an initiation request. See "Interpret table" on page 1017 for a description of how to define an interpret table.

LOGTAB is valid only for dependent logical units.

### LUDR

**LUDR=NO**

Indicates that VTAM rejects any attempt to move or delete the LU.

LUDR=NO is not valid if PUDR=YES. However, by using DRDS definitions you can create an LU with LUDR=NO under a PU with PUDR=YES, and DR operations on the LU will fail, but DR operations on the PU ignore the LUDR=NO value.

**LUDR=YES**

Indicates that the LU can be dynamically deleted or moved. Valid only if dynamic reconfiguration is supported for this LU.
LUGROUP

```plaintext
LUGROUP = lu_group_name
```

**statements:** PU
**dependencies:** none

Specifies the name of the model LU group that VTAM uses to select a model LU definition when dynamically defining an LU. This operand also indicates that the device supports dynamic definition of LUs.

LUSEED

```plaintext
LUSEED = lu_pattern_name
```

**statements:** PU
**dependencies:** LUGROUP

Provides a 1–8 character pattern name that the SDDLU exit routine uses to generate an LU name when dynamically defining an LU.

If you use the IBM-supplied SDDLU exit routine, this operand is required and must have the following format:

- The first character must be either alphabetical or one of the national characters (@ or $).
- The remaining characters can be either alphabetical, numerical, or national.
- There must be either two or three contiguous national # characters in the name. These characters represent holders, in which the logical unit’s local address is substituted when an LU’s name is generated. If you specify three # characters, the LU’s logical address is substituted in decimal for the # characters. If you specify two # characters, the LU’s logical address is substituted in hexadecimal for the # characters.

When used with a user-written SDDLU exit routine, this operand is optional and there are no restrictions on its format. For more information on the SDDLU exit, see [z/OS Communications Server: SNA Customization](https://www.ibm.com/docs/en/zos/2.4.0?topic=communications-server-sna-customization).

This operand is valid only when you code the LUGROUP operand.

MAXBFRU
Notes:

1. MAXBFRU defaults to 1 when HPR=NO, XID=NO, CONNTYPE=LEN, or the VTAM node is not HPR capable.
2. MAXBFRU defaults to 5 when HPR=YES, XID=YES, CONNTYPE=APPN, and the VTAM node is HPR capable.

Specifies the number of buffer units that VTAM uses to receive data from the physical unit. These buffer units are elements of the IOBUF buffer pool.

*number_of_buffers* is a decimal integer. Always code MAXBFRU such that the bufsize value for the IOBUF buffer pool times MAXBFRU is equal to or greater than the maximum RU size used by the controller.

For more information about determining a value for MAXBFRU, see [z/OS Communications Server: SNA Network Implementation Guide](#).

Note: When the TRLE operand is coded on the PU definition statement, the MAXBFRU value coded on the PU definition statement (if coded) is ignored. VTAM uses the MAXBFRU value from the TRLE definition statement in the TRL major node.

**MAXDATA**

For PU type 1:

For PU type 1:
Specifies the maximum number of bytes that the physical unit can receive in one PIU or PIU segment, including the transmission header (TH) and request or response header (RH). To determine the maximum PIU (or PIU segment) size that the physical unit can receive, consult the component publications for the specific type of SDLC station represented by this PU definition statement.

The maximum amount of user data that VTAM sends to the physical unit in one PIU is:

- MAXDATA minus 2 bytes (for a PU type 1)
- MAXDATA minus 6 bytes (for a PU type 2)

**Note:** The RU part of the PIU contains the user data. When VTAM segments the PIU, the middle and last PIUs contain three more bytes of user data in the RU because there is no RH field. The value for the PIU varies with the device; for example, the 3276 requires MAXDATA=262.

MAXDATA is ignored for type 2.1 nodes attached through an NCP with peripheral node type 2.1 support. The type 2.1 node supplies the actual value used when the connection is established.

See NCP, SSP, and EP Resource Definition Reference for more information about physical units connected through an NCP.

**Note:** If the TRLE operand is coded, MAXDATA is ignored. See “TRLE” on page 414 for more information.

**Note:**
1. In a local area network, the value for the MAXDATA operand on the PU definition statement must be less than or equal to the value for the MAXDATA operand on the PORT definition statement for the LAN major node to which the switched physical unit is connected.
2. NCP V4R3 changed the MAXDATA calculation for a type 1 physical unit. See NCP, SSP, and EP Resource Definition Reference for information about calculating a MAXDATA value for physical units on non-switched links.

**MDLENT**
**MDLENT**

```
MDLENT=model_name_table_entry_name
```

**statements:** PU, LU  
**dependencies:** MDLTAB

Specifies the name of the model name table entry to be used for this logical unit.

If you code the MDLENT operand, you must also code the MDLTAB operand. If you omit MDLENT and include MDLTAB, VTAM uses the first entry in the associated LU table by default.

**MDLTAB**

```
MDLTAB=model_name_table_name
```

**statements:** PU, LU  
**dependencies:** none

Specifies the name of the model name table to be used for the logical unit. See "Model name table" on page 1054 for a description of how to define a model name table.

The model name is passed only when VTAM initiates the session with the application program, for example, only when the LOGAPPL = parameter is also coded.

If you omit the LOGAPPL operand or the MDLTAB operand, VTAM will not provide the name of the model name table to the application during LU-LU session initiation. However, you can provide the model name in other ways. You can enter it from the terminal on the MODEL operand of the LOGON command, or VTAM can send it to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the MODEL operand of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace a model name table. See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**MODETAB**
Specifies the name of a logon mode table to be used for the logical unit. The name you code must be the name of a logon mode table created as described in "Logon mode table" on page 1026. If you do not supply a logon mode table for the logical unit on the MODETAB operand, an IBM-supplied default logon mode table (ISTINCLM) is used. If you specify a table, both the table you specify and the default table are used.

**name**

*On VBUILD:*

name is optional. If a syntax or definition error is detected with this statement during activation of the major node, name appears in the message issued to identify the error.

The optional name you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node's status.

*On PU:*

Provides the required minor node name of the physical unit represented by this statement.

*On LU:*

Provides the required minor node name of the logical unit represented by this statement.
Specifies whether this link station represents a connection to a native node.

If NATIVE is not specified on the ADJCP statement or on the PU definition statement, the two nodes negotiate their subnet affiliation during connection establishment.

- If the NETIDs match, the connection defaults to a native connection.
- If the NETIDs are different, the connection defaults to a nonnative connection.

No subnet negotiations will occur if NATIVE has been specified; in this case the value between the two nodes must match for connection to occur.

If you code NATIVE on both the ADJCP and PU definition statements, the values must match.

For transmission groups (TGs) between nodes, the value specified or negotiated for NATIVE must match. If a TG is already active, subsequent TG activations must not have a specified or negotiated value for NATIVE that conflicts with the value of NATIVE for the active TG. If there is a conflict, the TG attempting activation is not allowed to become active. If the activation is for a virtual-route-based transmission group then the SSCP-SSCP session will also fail.

**NETID**
NETID=network_id

If you specify NETID on the PU definition statement, VTAM ensures that when the physical unit is active, the connecting resource is within the network specified by NETID. The network ID specified by NETID is the CP NETID of the NETID where the PU resides. If you omit NETID, VTAM dynamically acknowledges the network ID during connection establishment and this network ID becomes the CP NETID. If you code NETID, and nonnative network connection is not supported, the definition is rejected unless the operand specifies VTAM’s network ID.

NETID=(,NOXNETLS)

Specifies that VTAM does not allow nonnative network attachment for LU-LU sessions for this PU. When NETID=(,NOXNETLS), VTAM indicates to the NCP that this PU uses VTAM’s native network ID.

LU-LU sessions might fail if the NETID sent in the BIND from the LU is different than the NETID specified in VTAM (even though PU-PU connections can be established). If no NETID is included in the BIND, or if the NETID is identical to that specified by VTAM, the LU-LU session should activate.

Note: The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

NETID=network_id

Specifies a 1–8 character network identifier. This operand is used in conjunction with the XNETALS start option in support of nonnative network connection. See “XNETALS start option” on page 978 for information about XNETALS.

Note: The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.
Specifies whether the adjacent node is expected to be a network node.

If you specify a value for NN, it is validated when a connection is attempted to an adjacent node. If the adjacent CP is not the type of node that is expected, then connection setup fails. If NN is not specified, then the APPN capabilities of the adjacent node are identified and accepted when a connection is established.

NN=NO
    Specifies that the adjacent node is expected to be an end node.

NN=YES
    Specifies that the adjacent node is expected to be a network node.

Restriction: In a peripheral subnetwork boundary configuration between a border node and a network node without border node function, the border node appears to the non-border node as an end node. In this case, coding NN=YES on the definition statement that represents the border node, a connection failure will occur. Avoid the failure by not coding the NN operand or by coding NN=NO.

PACING
Specifies how VTAM paces the flow of data from the boundary node, which performs pacing for a channel-attached SNA device, to the SLU. The PACING value is not used in all configurations.

For more information about pacing, see z/OS Communications Server: SNA Network Implementation Guide.

**PDELAY**

1. NEGLIGIB is the default if the TRLE operand is coded.

**Notes:**

1. NEGLIGIB is the default if the TRLE operand is coded.

**PUTYPE**
PUTYPE = 2

Specifies the physical unit type. PUTYPE=2 is the only valid value.

REGISTER

For independent LUs:

REGISTER = NO

REGISTER = CDSERVR

REGISTER = NETSRVR

For dependent LUs:

REGISTER = NETSRVR

REGISTER = CDSERVR

REGISTER = NO

Specifies how a resource should be registered. The default depends on whether you code REGISTER for dependent or independent LUs.

REGISTER=CDSERV

Indicates that an end node resource is registered to a network node server and directory resource registration is requested for it. A network node resource is registered at the central directory server. If the node is configured as a central directory server, this value has the same effect as NO.

REGISTER=NETSRVR

Indicates that the end node resource should be registered to its network node server, but that directory registration should not be requested for it.

For dependent LUs, NETSRVR is the default, and LOCADDR must be specified as a nonzero value.

Dependent LUs are not usually the targets of LU-LU session setup attempts and are therefore not likely to be the targets of an APPN LOCATE search. However, because end nodes cannot be searched, you can register their dependent LUs with their network node server.
REGISTER=NO
    Indicates that the resource should not be registered.
    For independent LUs, NO is the default, and LOCADDR must be specified as 0.

RESETLU
    RESETLU=YES
    Specifies that during clean-up processing for dependent LUs, VTAM will issue an ACTLU request after the DACTLU request.
    RESETLU=NO
    Specifies that during clean-up processing for LU-LU sessions, RESETLU=NO prevents a DACTLU-ACTLU from flowing to an LU.

SECNET
SECNET

Specifies whether this physical unit is associated with a secondary network containing resources whose connectivity is not defined to VTAM.

When SECNET=YES, data that is received by VTAM from this physical unit is flagged as requiring special problem-determination considerations when routed to a communication network management (CNM) application.

SECURITY

Specifies the security level of the transmission group. The following options range from the least secure level to the most secure level.

SECURITY=UNSECURE
    Specifies no security level.

SECURITY=PUBLIC
    Specifies a public switched network.

SECURITY=UNDERGRO
    Specifies an underground cable, not guarded.

SECURITY=SECURE
    Specifies a secure conduit, not guarded.

SECURITY=GUARDED
    Specifies a guarded conduit, physical only.

SECURITY=CRYPT
    Specifies link encryption.

SECURITY=SHIELDED
    Specifies a guarded conduit, physical and radiation shielded.

SPAN
**SPAN**

```
[SPAN=span_name]
```

**statements:**  
PU, LU

**dependencies:**  
NetView

Defines a span of control for VTAM minor node resources. Code this operand if you are using the NetView program. For a full description, see *IBM Tivoli NetView for z/OS Installation: Getting Started*.

The NetView program checks the SPAN value, but VTAM ignores it.

**SSCPFM**

```
SSCPFM=FSS
SSCPFM=USSCS
```

**statements:**  
PU, LU

**dependencies:**  
none

Determines what type of RUs the logical unit can support in its communications with the SSCP.

**SSCPFM=FSS**

Specifies that the logical unit supports formatted commands in its communication with the SSCP.

**Note:** You might receive character-coded messages when SSCP=\text{FSS}.

**SSCPFM=USSCS**

Specifies that the logical unit supports character-coded messages in its communication with the SSCP.

Refer to the publications for each individual device to determine whether formatted or character-coded messages are supported by that device.

**TGN**
Specifies the transmission group (TG) number that represents the connection between this node and the destination APPN node.

By specifying a \textit{tg\_number}, you indicate a preferred TG that might be used depending on how the partner nodes negotiate the number.

Coding TGN=ANY, or not coding TGN, allows the transmission group number to be negotiated.

Coding TGN=0 means that the PU does not support parallel TGs. For connections to LEN nodes, or to any node that does not support parallel TGs, TGN=0 is always assumed.

For APPN-capable nodes, the CPNAME and NETID operands are required with the TGN operand. Note that you can specify the CPNAME operand without the TGN operand and the transmission group number will be negotiated. However, if a value for TGN is specified, then a value for CPNAME and NETID must also be specified.

\textbf{TGP}
**TGP**

```
TGP->tgp_name
```

- **statements:** PU
- **dependencies:** APPN only

Specifies the name of a transmission group (TG) profile. The characteristics of the TG profile (along with any modifiers specified on this statement) become the characteristics of the PU. If a `tgp_name` is not specified or has not been activated when the PU becomes active, the default transmission group operand values or the override values specified on GROUP, LINE, or PU statements are used.

See "APPN transmission group profile" on page 720 for more information on transmission group profiles.

**TOPO**

```
TOPO->CONNECT
```

- **statements:** PU
- **dependencies:** APPN only

Controls the way APPN connections using this link station will be reported to APPN topology and routing services for inclusion into APPN functions.

**Note:** Except in the switched major node, the only value for TOPO is CONNECT.

If you specify APPN on the CONNTYPE start option, it is not necessary to code TOPO, as you automatically get a value of CONNECT.

**TOPO=CONNECT**

APPN topology and routing services receives information about the connection and its characteristics when the connection is activated. When the connection is deactivated, APPN topology and routing services receives notification that the TG is inactive.

**TRLE**
**TRLE**

```
TRLE = trle_name
```

**statements:** PU

**dependencies:** CUADDR; APPN only

Specifies the 8-character name of the TRLE definition statement to be used for the physical unit being defined by this PU definition statement. The TRLE determines the transport characteristics of the PU.

If you specify a value on the TRLE operand that does not have a corresponding TRLE definition statement in the TRL major node, activation of the PU fails.

**Note:** You cannot code TRLE with the CUADDR operand. However, if you omit both TRLE and CUADDR, you must code ISTATUS=INACTIVE and include the address in the U operand in the VARY ACT command that activates the physical unit.

**TYPE**

```
TYPE = LOCAL
```

**statements:** VBUILD

**dependencies:** none

Specifies a local SNA major node. This operand is required.

**UPARM1**
UPARM1

- user-defined_operand_value

statements: PU
dependencies: APPN only
range: 0–255

Specifies a user-defined operand value.

See z/OS Communications Server: SNA Network Implementation Guide for an example about using the UPARM operands.

UPARM2

- user-defined_operand_value

statements: PU
dependencies: APPN only
range: 0–255

Specifies a user-defined operand value.

UPARM3
UPARM3 = 128

UPARM3 = user-defined_operand_value

**UPARM3**

- **statements:** PU
- **dependencies:** APPN only
- **range:** 0–255

Specifies a user-defined operand value.

**USSTAB**

- **statements:** PU, LU
- **dependencies:** dependent LUs

Specifies the name of a USS table that VTAM uses to process character-coded input that it receives from the logical unit.

A terminal user can issue a USS command with the LANGTAB operand. This causes a second USS table to be associated with the logical unit, which overrides the table specified with USSTAB. If you do not code USSTAB and a LANGTAB USS table is not in use, the IBM-supplied USS table (ISTINCDT) is used. For more information on USS tables, see "Unformatted system services tables" on page 1069.

USSTAB is valid for dependent logical units only.

**VERALSID**
Notes:

1. VERALSID=NO is the only valid value if the TRLE operand is coded.

   statements:         PU
   dependencies:      APPN only

Specifies whether the adjacent link station name should be used to determine the
link station. When specified as YES, the connection endpoints must supply the
same name in the CV0EF7 vector that is appended to XID3s. The name supplied in
CV0EF7 is the name of a switched PU or the name of a PU under a non-switched
line.

**VPACING**

   VPACING = 1
   VPACING = number_of_requests

   statements:       PU, LU
   dependencies: none
   range:            0–63

Specifies how VTAM paces the flow of data from the boundary node containing
the PLU to the boundary node that performs pacing for a channel-attached SNA
device.

For more information on pacing, see [z/OS Communications Server: SNA Network
Implementation Guide](#).

**XID**
XID = NO

Specifies that the local SNA PU type 2 channel contact procedure is to be used.

Note: If TRLE is coded, activation of the PU fails.

XID = YES

Specifies that a type 2.1 channel contact procedure is to be used.

XNETALS

Determines whether the network ID of an attaching node or VTAM's network ID is used for an adjacent nonnative node. Used to control how a nonnative node can attach to VTAM.

Note: The XNETALS option has changed in the way VTAM treats PUs that send in nonnative NETIDs in their XID. VTAM used to let the PU connect and treated it as if it sent in the same NETID that VTAM started with but that is no longer true. If the NETID that the PU sent in its XID is not the same as VTAM's and XNETALS=NO or is defaulted, IST680I will be issued.

XNETALS=NO

Indicates that VTAM's network ID will be used to represent the connecting node's network ID.

During connection:
- If the NETID operand is specified on the PU statement, then it must specify VTAM's network ID.
- If the NETID operand is not coded on the PU statement, then VTAM's network ID is assumed and the adjacent node's network ID is ignored.

During LU-LU session establishment:
- If the network ID for the originating LU is specified (during connection time) and is not identical to VTAM's network ID, the session will not be established.
- If the originating LU is not using a network qualified name, VTAM's network ID is assumed and the session will be established.
Note: If the PU is a dependent LU server or requester, nonnative network connection is not allowed if XNETALS=NO is specified and the NETID operand is not coded.

**XNETALS=**YES
Indicates that the network ID of the adjacent PU will be used for connection and session establishment.

During connection:
- If the NETID operand on the PU statement is coded, it will be used as the PU's network ID. In this case, the network ID specified on the PU statement must match the network ID specified by the attaching node during connection processing.
- If the NETID operand is not specified on the PU statement, then the network ID specified by the attaching node during connection processing will be assumed.

During LU-LU session establishment:
- If the network ID for the originating LU is specified and does not match the PU's network ID, the session will not be established.
- If the originating LU name is not a network qualified name, the session will be established. VTAM will qualify the LU's name with the attaching node's network ID.

Note: Providing that XNETALS=**YES** is coded, you can prohibit nonnative network attachment for LU-LU sessions for a particular PU by coding XNETALS=**NO** on the GROUP, LINE or PU statements.

---

**LU group major node**

To define an LU group major node, code a VBUILD definition statement for the major node, and an LUGROUP definition statement for each model LU group in the major node.

The LU group major node defines one or more model LU groups. Each model LU group contains a list of model LU definitions. VTAM uses the model LU definitions to dynamically define LUs attached to devices that support dynamic definition of dependent LUs.

Each model LU group consists of an LUGROUP definition statement followed by at least one, but no more than 255, LU definition statements. Each LU definition statement represents the LU definitions that are used to dynamically define LUs of a particular device type.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=LUGROUP</td>
<td>355</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>LUGROUP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>LU</td>
<td>ASLENT</td>
<td>343</td>
<td></td>
<td></td>
<td></td>
<td>ASLTAB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASLTAB</td>
<td>343</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CERTIFY</td>
<td>344</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CKEYNAME</td>
<td>344</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

*Table 29. LU group major node definition statements and operands*
### Table 29. LU group major node definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLRSESSQ</td>
<td></td>
<td></td>
<td>345</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLOGMOD</td>
<td></td>
<td></td>
<td>345</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENCR</td>
<td></td>
<td></td>
<td>346</td>
<td></td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENCRTYPE</td>
<td></td>
<td></td>
<td>346</td>
<td></td>
<td>DES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FASTPASS</td>
<td></td>
<td></td>
<td>347</td>
<td></td>
<td>YES</td>
<td>TSO only</td>
<td></td>
</tr>
<tr>
<td>LOGAPPL</td>
<td></td>
<td></td>
<td>348</td>
<td></td>
<td>NetView or TSO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOGTAB</td>
<td></td>
<td></td>
<td>349</td>
<td></td>
<td>dependent LUs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LUDR</td>
<td></td>
<td></td>
<td>350</td>
<td></td>
<td>YES</td>
<td>MDLTAB</td>
<td></td>
</tr>
<tr>
<td>MDLENENT</td>
<td></td>
<td></td>
<td>350</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDTAB</td>
<td></td>
<td></td>
<td>351</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODETAB</td>
<td></td>
<td></td>
<td>351</td>
<td></td>
<td>ISTINCLM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PACING</td>
<td></td>
<td></td>
<td>353</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGISTER</td>
<td></td>
<td></td>
<td>353</td>
<td></td>
<td>NETSRVR</td>
<td>APPN only</td>
<td></td>
</tr>
<tr>
<td>RESETLU</td>
<td></td>
<td></td>
<td>354</td>
<td>P</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSCFMM</td>
<td></td>
<td></td>
<td>355</td>
<td></td>
<td>FSS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>USSTAB</td>
<td></td>
<td></td>
<td>356</td>
<td></td>
<td>ISTINCDT</td>
<td>dependent LUs</td>
<td></td>
</tr>
<tr>
<td>VPACING</td>
<td></td>
<td></td>
<td>356</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

- **R** Required operand.

---

**LU group major node coding notes**

This information contains coding notes for the LU group major node.

**VBUILD definition statement**

Code a VBUILD definition statement for each LU group major node.

**LUGROUP definition statement**

Code an LUGROUP definition statement to specify the start of each model LU group in the LU group major node. All the LU definition statements that follow this definition represent model LUs in the model LU group. The model LU group ends when VTAM encounters either a VBUILD or an LUGROUP definition statement.

**LU definition statement**

Code an LU definition statement for each model LU in the LU group. On type 2.0 and type 2.1 devices that support dynamic definition of dependent LUs, the model LU definitions are used to dynamically define dependent LUs when the device powers on. The maximum number of model LUs that you can define for each LU group is 255.

You cannot take advantage of the sift-down effect in this major node.

**LU group major node full syntax**

The full syntax for the LU group major node follows. The syntax for each operand is repeated in **"LU group major node operand descriptions" on page 343.**
**LU group major node operand descriptions**

The following information describes the operands you can code for the LU group major node.

**ASLENT**

```
| ASLENT = associated_lu_table_entry_name |
```

*statements: LU*

*dependencies: ASLTAB*

Specifies the name of the associated LU table entry to be used for this logical unit.

If you code the ASLENT operand, you must also code the ASLTAB operand. If you omit ASLENT and include ASLTAB, VTAM uses the first entry in the table named in ASLTAB by default.

**ASLTAB**

```
| USTAB = uss_table_name |
```

*VPACING = number_of_requests*
ASLTAB = associated_lu_table_name

**Statements:** LU

**Dependencies:** none

Specifies the name of the associated LU table to be used for the logical unit. See "Associated LU table" on page 981 for a description of how to define an associated LU table.

If you omit the ASLTAB operand, VTAM will not provide the names of associated logical units to the application during LU-LU session initiation. However, you can provide the associated LU names in other ways. You can enter them from the terminal on the PRINTER1 and PRINTER2 operands of the LOGON command, or VTAM can send them to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the PRINTER1 and PRINTER2 operands of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace an associated LU table (ASLTAB). See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**CERTIFY**

**Statements:** LU

**Dependencies:** CERTIFY is ignored unless ENCR is also active. Before enabling the CERTIFY function, ensure that the device or software providing the partner’s LU support also supports the CERTIFY function.

Specifies whether VTAM on behalf of the application program will verify authentication.

**CERTIFY=**YES

Specifies SLU authentication (verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

**CERTIFY=**NO

Specifies no SLU authentication (not verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

**CKEYNAME**
Specifies the name of the key-encrypting key (KEK) in the cryptographic key data set (CKDS) for the LU. Use CKEYNAME to encipher the session key that is sent in the BIND to the LU when a cryptographic session is established. CKEYNAME has the same format and restrictions as other VTAM names. For information on naming conventions, see “Format of definition statements” on page 8.

If you do not specify CKEYNAME, the cryptographic key name defaults to the name on the LU definition statement.

**Note:** Independent LUs will ignore the CKEYNAME keyword and use their default name.

**CLRSESSQ**

Specifies whether a dependent LU is forced into an inhibited session state when the LU is disabled. When the LU is enabled, it is returned to normal status. Sessions, however, are not connected.

**CLRSESSQ=**YES

Specifies that the LU is forced into an inhibited session state. Both active and pending sessions are removed from the session queue.

**CLRSESSQ=**NO

Specifies that the LU is not forced into inhibited session state. This is the default.
DLOGMOD = default_logon_mode_table_entry_name

statements: LU
dependencies: MODETAB

Specifies the name of the default logon mode table entry used by default if one is not otherwise provided. If you do not code this operand, and the name of a logon mode table entry is not otherwise provided, VTAM uses the first entry in the applicable logon table (specified on the MODETAB operand or used by default).

If you specify MODETAB, the entry must be in either the specified table or in ISTINCLM, an IBM-supplied logon mode table. If you do not specify MODETAB, the entry must be in ISTINCLM. For more information on logon mode entries, see "Logon mode table" on page 1026.

ENCR

ENCR = NONE

Tells whether this logical unit has any special requirements for enciphering and deciphering messages.

ENCR=NONE
Specifies that this logical unit cannot engage in cryptographic sessions.

ENCR=OPT
Specifies that this logical unit can engage in cryptographic sessions, but allows the application program to determine whether to use cryptography.

ENCR=REQD
Specifies that VTAM must encipher all messages to and from this logical unit. If ENCR=REQD, no sessions can be established with this logical unit unless the host with which it is associated is able to handle cryptographic sessions.

ENCR=SEL
For an LU group major node, SEL has no meaning for logical units. If you specify ENCR=SEL, ENCR=OPT is used instead.

ENCRTYPE
ENCRTYPE=DES
   Specifies that VTAM must use a minimum of DES encryption with an 8-byte key when performing session level encryption. This is the default.

ENCRTYPE=TDES24
   Specifies that VTAM must use a minimum of Triple DES encryption with a 24-byte key when performing session level encryption.

The SYSDEFeD ENCRTYPE of a LU can be changed at runtime through the MODIFY SECURITY command. See z/OS Communications Server: SNA Operation for specific details. However, the ENCRTYPE value will not be able to be downgraded. In other words, if the application is currently operating at ENCRTYPE=TDES24, then the encryption level cannot be changed to ENCRTYPE=DES using the MODIFY SECURITY command.

Note: If downgrading is preferred, the current major node being used must be deactivated and another major node value must be coded.

In addition, a user may use the LOGMODE TABLE to override the ENCRTYPE value of the SLU based on the logmode entry being used. The ENCRTYPE value will not be able to be downgraded from TDES24 to DES using the logmode table. Only TDES24 can be specified for the encryption level (ENCRTYP). See "MODEENT macroinstruction" on page 1027 for more information about MODEENT macroinstructions.

FASTPASS
Determines how session establishment is performed for application programs that issue the CLSDST macroinstruction with the PASS option as part of their session establishment procedure. The application program acts as a secondary logical unit (SLU).

The PLU acknowledges the capability of the SLU during session establishment. If you have coded the logon performance enhancement in the PLU’s application program, only a single BIND is issued to establish the session.

Some SLUs are incompatible with this enhancement. If your SLU is incompatible with this enhancement, code FASTPASS=NO.

Note: FASTPASS applies only to application programs (acting as SLUs) that establish sessions with TSO.

**FASTPASS=NO**

Specifies that an unenhanced logon procedure requiring multiple binds is performed for application programs that issue the CLSDST PASS macroinstruction with the PASS option as part of their session establishment procedure.

**FASTPASS=YES**

Specifies that the logon performance enhancement is performed for application programs that issue the CLSDST macroinstruction with the PASS option as part of their session establishment procedure.

**LOGAPPL**
Identifies the name of the primary logical unit that this logical unit will automatically log on to as a secondary logical unit when this logical unit is activated. The name you specify can also be a USERVAR that will be resolved into its current value before being used to establish or redrive the session with the controlling primary logical unit.

For independent logical units, the LOGAPPL operand establishes a session with the primary logical unit designated on the LOGAPPL operand if a session does not currently exist with this logical unit. The session is established regardless of whether sessions with other logical units exist. The LOGAPPL operand cannot contain the name of an independent logical unit because independent logical units cannot be the controlling logical units for an automatic logon session.

To ensure retry of a pending automatic logon, code the AUTORTRY or AUTOTI start options.

You can specify a network-qualified name for LOGAPPL. You cannot specify an LUALIAS name for LOGAPPL.

Note:
1. If an error occurs for the switched connection (INOP), VTAM reallocates the LOGAPPL session and attempts to redial the logical unit to recover the connection. In other words, the session will be terminated by breaking the dial connection instead of by normal session termination using LOGOFF and TERMSELF.
2. If you specify the NetView or TSO program as the application program on the LOGAPPL operand, you will get unpredictable results in recovery situations.

For more information on establishing sessions, see z/OS Communications Server: SNA Network Implementation Guide.
LOGTAB = interpret_table_name

**statements:** LU
**dependencies:** valid only for dependent LUs

Specifies the name of an interpret table that VTAM uses when processing logon requests originating from the logical unit. VTAM uses the interpret table to interpret the name received from an initiation request. See "Interpret table" on page 1017 for a description of how to define an interpret table.

LOGTAB is valid for dependent logical units only.

**LUDR**

**LUDR=NO**

Specifies that an LU cannot be dynamically deleted or moved.

**LUDR=YES**

Specifies that an LU can be dynamically deleted or moved.

**MDLENT**
MDLENT

\[
\text{MDLENT} \rightarrow \text{model_name_table_entry_name}
\]

**statements:** LU  
**dependencies:** MDLTAB

Specifies the name of the model name table entry to be used for this logical unit.

If you code the MDLENT operand, you must also code the MDLTAB operand. If you omit MDLENT and include MDLTAB, VTAM uses the first entry in the table named in MDLTAB.

**MDLTAB**

\[
\text{MDLTAB} \rightarrow \text{model_name_table_name}
\]

**statements:** LU  
**dependencies:** none

Specifies the name of the model name table to be used for the logical unit. See "Model name table" on page 1054 for a description of how to define a model name table.

If you omit the MDLTAB operand, VTAM will not provide the name of the model name table to the application during LU-LU session initiation. However, you can provide the model name in other ways. You can enter it from the terminal on the MODEL operand of the LOGON command, or VTAM can send it to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the MODEL operand of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace a model name table. See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**MODETAB**
MODETAB = ISTINCLM

Specifies the name of a logon mode table to be used for the logical unit. The name you code must be the name of a logon mode table created as described in "Logon mode table" on page 1026. If you do not supply a logon mode table for the logical unit on the MODETAB operand, an IBM-supplied default logon mode table (ISTINCLM) is used. If you specify a table, both the table you specify and the default table are used.

**name**

Specifies the name of the resource being defined.

**On VBUILD:**

*name* is optional. If a syntax or definition error is detected with this statement during activation of the major node, *name* appears in the message issued to identify the error.

The optional *name* you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node’s status.

**On LUGROUP:**

Provides the required name of the model LU group.

**On LU:**

Provides the required name that VTAM uses when selecting the model LU group that it will use to build dynamic LU definitions. *name* is matched against the machine type and model number (model acronym). The control unit provides VTAM with the model acronym when the device powers on.

Each *name* within a model LU group must be unique. However, different model LU groups can contain model LU definitions with the same name.

For more information on model acronyms and how VTAM matches a model acronym to a model LU definition, see z/OS Communications Server: SNA Network Implementation Guide.

The following items are valid characters:

- Alphabetic (A—Z)
- Numerical (0—9)
• National (@, #, or $)

When matching an LU acronym to an LU definition, the @ character has the following special meanings:
• Within a name, @ represents a single unidentified character.
• If @ is the last character in a name, it represents any number of unknown characters.
• If @ is the only character used in a name, it will match any LU acronym.

**PACING**

Statements: LU
dependencies: none
range: 0–63

Specifies how VTAM paces the flow of data between the logical unit and the boundary node to which it is connected. The PACING value is not used in all configurations.

For more information on pacing, see [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/SSEPGG_2.2.0/com.ibm.zos.v2r2.doc/ct_vtamp.html).

**REGISTER**
Specifies how a resource should be registered. The default depends on whether you code REGISTER for dependent or independent LUs. Because the LU group major node is used to define only dependent LUs, the default is NETSRVR.

**REGISTER=CDSERV**
Indicates that an end node resource is registered to a network node server and directory resource registration is requested for it. A network node resource is registered at the central directory server. If the node is configured as a central directory server, this value has the same effect as NO.

**REGISTER=NETSRVR**
Indicates that the end node resource should be registered to its network node server, but that directory registration should not be requested for it.

For dependent LUs, NETSRVR is the default, and LOCADDR must be specified as a nonzero value.

Dependent LUs are not usually the targets of LU-LU session setup attempts and are therefore not likely to be the targets of an APPN LOCATE search. However, because end nodes cannot be searched, you can register their dependent LUs with their network node server.

**REGISTER=NO**
Indicates that the resource should not be registered.

For independent LUs, NO is the default, and LOCADDR must be specified as 0.

**RESETLU**
RESETLU = YES

Specifies that during clean-up processing for dependent LUs, VTAM will issue an ACTLU request after the DACTLU request.

RESETLU = NO

Specifies that during clean-up processing for LU-LU sessions, RESETLU = NO prevents a DACTLU-ACTLU from flowing to an LU.

SSCPFM

Determines what type of RUs the logical unit can support in its communications with the SSCP.

SSCPFM = FSS

Specifies that field-formatted messages are supported.

Note: You might receive character-coded messages when SSCPFM = FSS.

SSCPFM = USSSCS

Specifies that character-coded commands are supported.

SSCPFM = USS3270

Allows printing of a user-defined logon message (USSMSG10) on BSC printers.

SSCPFM = USS3275

Prevents printing of a user-defined logon message (USSMSG10) for a terminal defined as an SDLC 3275 (PU type 1) that might have an attached printer. Code SSCPFM = USS3275 to prevent the user-defined logon message from printing on BSC printers.
LUGROUP

Definitions:

VBUILD

dependencies: none

Defines an LU group major node to VTAM. VTAM uses the model LU groups to
dynamically define dependent LUs attached to type 2.0 and type 2.1 devices that
support dynamic definition of dependent LUs.

USSTAB

USSTAB = ISTINCDT

statements: LU

dependencies: dependent LUs only

Specifies the name of a USS table that VTAM uses to process character-coded input
that it receives from the logical unit.

A terminal user can issue a USS command with the LANGTAB operand. This
causes a second USS table to be associated with the logical unit, which overrides
the table specified with USSTAB. If you do not code USSTAB and a LANGTAB
USS table is not in use, the IBM-supplied USS table (ISTINCDT) is used. For more
information on USS tables, see "Unformatted system services tables" on page 1069.

USSTAB is valid for dependent logical units only.

VPACING
VPACING = number_of_requests

Determines how VTAM paces the flow of data from VTAM to the boundary node to which the logical unit is connected.

For more information about pacing, see z/OS Communications Server: SNA Network Implementation Guide

**Model major node**

To define dynamic switched PU and LU definitions in a model major node, code a VBUILD definition statement, followed by PU and LU definition statements. The PU and LU definitions contained within this node are used to build dynamic switched definitions for previously undefined devices.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>VBUILD GROUP</td>
<td>TYPE=MODEL</td>
<td></td>
<td>416</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See the LU definition statement for operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>PU</td>
<td>ADDR</td>
<td></td>
<td>368</td>
<td>R</td>
<td></td>
<td></td>
<td>non-switched connections</td>
</tr>
<tr>
<td></td>
<td>ANS</td>
<td></td>
<td>369</td>
<td></td>
<td>STOP</td>
<td></td>
<td>APPN only; Type 2.1</td>
</tr>
<tr>
<td></td>
<td>AUTHLEN</td>
<td></td>
<td>370</td>
<td></td>
<td>YES</td>
<td></td>
<td>APPN only; not valid with DYNTYPE=VN</td>
</tr>
<tr>
<td></td>
<td>CAPACITY</td>
<td></td>
<td>371</td>
<td></td>
<td>8K</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td></td>
<td>CONNTYPE</td>
<td></td>
<td>374</td>
<td></td>
<td>CONNTYPE start option</td>
<td></td>
<td>APPN only; not valid with DYNTYPE=VN</td>
</tr>
<tr>
<td></td>
<td>COSTBYTE</td>
<td></td>
<td>375</td>
<td></td>
<td>0</td>
<td></td>
<td>APPN only; not valid with DYNTYPE=VN</td>
</tr>
<tr>
<td></td>
<td>COSTTIME</td>
<td></td>
<td>375</td>
<td></td>
<td>0</td>
<td></td>
<td>APPN only; DYNTYPE=EE only</td>
</tr>
<tr>
<td></td>
<td>CPCP</td>
<td></td>
<td>376</td>
<td></td>
<td>CPCP start option or YES</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td></td>
<td>DATMODE</td>
<td></td>
<td>376</td>
<td></td>
<td>HALF</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DISCNT</td>
<td></td>
<td>377</td>
<td></td>
<td>(NO)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DWINOP</td>
<td></td>
<td>379</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DYNAJCP</td>
<td></td>
<td>379</td>
<td></td>
<td>DYNAJCP start option</td>
<td></td>
<td>APPN only</td>
</tr>
</tbody>
</table>
Table 30. Model major node definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYNLU</td>
<td></td>
<td></td>
<td>380</td>
<td></td>
<td></td>
<td>DYNLU start option</td>
<td></td>
</tr>
<tr>
<td>DYNTYPE</td>
<td></td>
<td></td>
<td>381</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EEVERIFY</td>
<td></td>
<td></td>
<td>384</td>
<td></td>
<td></td>
<td>EEVERIFY</td>
<td>DYNTYPE=EE only</td>
</tr>
<tr>
<td>HPR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>APPN only</td>
</tr>
<tr>
<td>HPREEARB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HPRPSDLY start option value</td>
<td>DYNTYPE=EE only</td>
</tr>
<tr>
<td>HPRPSDLY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INCLUD0E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IRETRY</td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LANTEST</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>see description</td>
</tr>
<tr>
<td>LIMRES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>LIMQSINT; LU 6.2; PUTYPE=1, 2</td>
</tr>
<tr>
<td>LUGROUP</td>
<td></td>
<td></td>
<td>395</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LUSEED</td>
<td></td>
<td></td>
<td>395</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXDATA</td>
<td></td>
<td></td>
<td>396</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXOUT</td>
<td></td>
<td></td>
<td>397</td>
<td></td>
<td></td>
<td>261 or 265</td>
<td></td>
</tr>
<tr>
<td>MAXSESS</td>
<td></td>
<td></td>
<td>398</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>MTU</td>
<td></td>
<td></td>
<td>400</td>
<td></td>
<td></td>
<td>The value learned from the TCP/IP stack</td>
<td>DYNTYPE=EE only</td>
</tr>
<tr>
<td>NATIVE</td>
<td></td>
<td></td>
<td>402</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NETID</td>
<td></td>
<td></td>
<td>402</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NN</td>
<td></td>
<td></td>
<td>403</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PACING</td>
<td></td>
<td></td>
<td>404</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PASSLIM</td>
<td></td>
<td></td>
<td>405</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PDELAY</td>
<td></td>
<td></td>
<td>406</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PUTYPE</td>
<td></td>
<td></td>
<td>407</td>
<td></td>
<td></td>
<td>2</td>
<td>APPN only; not valid with DYNTYPE=VN</td>
</tr>
<tr>
<td>REDDELAY</td>
<td></td>
<td></td>
<td>407</td>
<td></td>
<td></td>
<td>30</td>
<td>DYNTYPE=EE and DWINOP=YES only</td>
</tr>
<tr>
<td>REDIAL</td>
<td></td>
<td></td>
<td>408</td>
<td></td>
<td></td>
<td>3</td>
<td>DYNTYPE=EE and DWINOP=YES only</td>
</tr>
<tr>
<td>SECNET</td>
<td></td>
<td></td>
<td>410</td>
<td></td>
<td></td>
<td>NO</td>
<td>APPN only; not valid with DYNTYPE=VN</td>
</tr>
<tr>
<td>SECURITY</td>
<td></td>
<td></td>
<td>411</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPAN</td>
<td></td>
<td></td>
<td>411</td>
<td></td>
<td></td>
<td>ANY</td>
<td>NetView</td>
</tr>
<tr>
<td>TGN</td>
<td></td>
<td></td>
<td>413</td>
<td></td>
<td></td>
<td></td>
<td>DYNTYPE=EE only</td>
</tr>
<tr>
<td>TGP</td>
<td></td>
<td></td>
<td>414</td>
<td></td>
<td></td>
<td></td>
<td>APPN only; not valid with DYNTYPE=VN</td>
</tr>
<tr>
<td>TOPO</td>
<td></td>
<td></td>
<td>414</td>
<td></td>
<td></td>
<td>CONNECT</td>
<td>APPN only</td>
</tr>
<tr>
<td>TRLE</td>
<td></td>
<td></td>
<td>414</td>
<td></td>
<td></td>
<td></td>
<td>XCF connectivity only</td>
</tr>
</tbody>
</table>
Table 30. Model major node definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>UPARM1</td>
<td>416</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
<td></td>
<td>APPN only; not valid with DYNTYPE=VN</td>
</tr>
<tr>
<td>UPARM2</td>
<td>417</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
<td></td>
<td>APPN only; not valid with DYNTYPE=VN</td>
</tr>
<tr>
<td>UPARM3</td>
<td>417</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
<td></td>
<td>APPN only; not valid with DYNTYPE=VN</td>
</tr>
<tr>
<td>VPACING</td>
<td>419</td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>VERALSID</td>
<td>418</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>VTAMTOPO</td>
<td>420</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>XNETALS start option value</td>
</tr>
</tbody>
</table>

Legend:

R    Required operand.

G    Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.
Model major node coding notes

This information contains coding notes for the model major node.

VBUILD definition statement

Code a VBUILD definition statement for each model major node.

GROUP definition statement

Code a GROUP definition statement to group LU definition statements. All operands that are valid on the LU definition statement can be coded instead on the GROUP statement for sifting to the LU statements. There is no sifting to the PU statements. Sifting from the GROUP statement to the LU statement remains in effect for the remainder of the processing of this major node, or until the next GROUP statement is encountered.

PU definition statement

Code a PU definition statement for each physical unit in the model major node. The model PU definition statement is used to create a PU when the installation exit routine returns the name of the model and the name of the previously undefined PU to VTAM.

When defining a model major node, see z/OS Communications Server: SNA Customization for information about using the NIDDEF file. The exit routine is required.

Restrictions:
- The XCF, RTP, VN and EE model PUs do not use the configuration services exit.
- There is no sift-down from the GROUP definition statement to the PU definition statement. Nor is there any sift-down from the PU definition statement to the LU definition statement.

LU definition statement

Code an LU definition statement for each logical unit in the model major node. The model LU definition statement is used to create LUs when the installation exit routine returns the name of the model and the name of the previously undefined LU to VTAM.

Tip: There is no sift-down from the PU definition statements. A GROUP definition statement can be coded to enable sifting of LU keywords to LU definition statements.

NODETYPE start option

Note that if you do not code the NODETYPE start option, any APPN operands you code in this major node are ignored.

The NODETYPE start option allows APPN function. The combination of NODETYPE and the HOSTSA start option determines the various configurations of subarea node, interchange node, migration data host, network node, or end node.

Model major node full syntax

The full syntax for the model major node follows. The syntax for each operand is repeated in "Model major node operand descriptions" on page 368.
See the LU definition statement for allowed operands.

PU

AUTHLEN = YES
AUTHLEN = NO
CAPACITY = 8K
CAPACITY = number_of_bits

CONNTYPE = APPN
CONNTYPE = APPN (2)

COSTBYTE = 0
COSTBYTE = cost_per_byte
COSTTIME = 0
COSTTIME = cost_per_unit_time
DISCNT values:

```
(YES - F, -NF - )
```

DISCNT DELAY values:

```
(DELAY - F, -NF - delay_time - )
```

Notes:
1. TG characteristics are not valid with DYNTYPE=VN.
2. CONNTYPE=APPN is the only valid value if the TRLE operand is coded.
3. TG characteristics are not valid with DYNTYPE=VN.
4. TG characteristics are not valid with DYNTYPE=VN.
5. CPCP defaults to YES for the model RTP PU.
6. CPCP defaults to the value of the CPCP start options for all PUs except the model RTP PU.
7. DYNADJCP=YES is the only valid value if the TRLE operand is coded.
8. DYNLU=YES is the only valid value if the TRLE operand is coded.
9. LIMRES=NO is the only valid value if the TRLE operand is coded.
10. MAXDATA defaults to 261 when PUTYPE=1 is coded.
11. MAXDATA defaults to 265 when PUTYPE=2 is coded.
12. NEGLIGIB is the default if the TRLE operand is coded.
13. TG characteristics are not valid with DYNTYPE=VN.
14. PUTYPE=2 is the only valid value if the TRLE operand is coded.
15. TG characteristics are not valid with DYNTYPE=VN.
16. TG characteristics are not valid with DYNTYPE=VN.
17. TG characteristics are not valid with DYNTYPE=VN.
18. TG characteristics are not valid with DYNTYPE=VN.
19. TG characteristics are not valid with DYNTYPE=VN.
20. VERALSID=NO is the only valid value if the TRLE operand is coded.

LU
Notes:
1  REGISTER defaults to NETSRVR for dependent LUs.
2  REGISTER defaults to NO for independent LUs.

**Model major node operand descriptions**
The following information describes the operands you can code for the model major node.

**ADDR**
**ADDR** - *link_station_address*

**statements:** PU
**dependencies:** non-switched connections

Specifies the hexadecimal SDLC station address for the physical unit. You can specify one or two hexadecimal digits for ADDR. For example, if the address is X'03', you can code ADDR=03 or ADDR=3.

This address is required for all physical units attached through SDLC lines and X.25 networks.

**Note:**
1. This operand is ignored for switched subarea connections.
2. If the TRLE operand is coded, ANS is ignored. See “TRLE” on page 414 for more information.

**ANS**

**statements:** PU
**dependencies:** none

Determines whether the physical unit continues to operate when the NCP enters automatic network shutdown. The value coded on this definition statement overrides whatever value is coded on the ANS definition statement for the NCP.

For switched subarea connections, this operand specifies whether a switched subarea connection over lines attached to this physical unit should remain active or be dropped when the owning SSCP is lost. This is used only on a PU definition statement that defines a subarea physical unit that will be connected to a link station within an NCP.

**Note:**
1. This operand is ignored for switched subarea connections.
2. If the TRLE operand is coded, ANS is ignored. See “TRLE” on page 414 for more information.

**ANS=CONTINUE or CONT**

Indicates that LU-LU sessions will continue when the NCP enters automatic network shutdown.

You must code ANS=CONTINUE or CONT if the terminal is to participate in XRF sessions.

When a session is continued and takeover occurs, the dependent logical unit taken over must support ACTLU(ERP) or the session is terminated during takeover processing. An independent logical unit does not receive ACTLU, so this restriction does not apply.
Note that there are several restrictions for session continuity in certain configurations. For information on these restrictions, see z/OS Communications Server: SNA Network Implementation Guide.

**ANS=STOP**

Indicates that LU-LU sessions will be terminated when the NCP enters automatic network shutdown for all LUs associated with this PU.

### ASLENT

<table>
<thead>
<tr>
<th>Statement</th>
<th>Group, LU</th>
<th>Dependencies</th>
<th>Associated LU Table Entry Name</th>
</tr>
</thead>
</table>

Specifies the name of the associated LU table entry to be used for this logical unit.

If you code the ASLENT operand, you must also code the ASLTAB operand. If you omit ASLENT and include ASLTAB, VTAM uses the first entry in the associated LU table by default.

### ASLTAB

<table>
<thead>
<tr>
<th>Statement</th>
<th>Group, LU</th>
<th>Dependencies</th>
<th>Associated LU Table Name</th>
</tr>
</thead>
</table>

Specifies the name of the associated LU table to be used for the logical unit. See "Associated LU table" on page 981 for a description of how to define an associated LU table.

If you omit the ASLTAB operand, VTAM will not provide the names of associated logical units to the application during LU-LU session initiation. However, you can provide the associated LU names in other ways. You can enter them from the terminal on the PRINTER1 and PRINTER2 operands of the LOGON command, or VTAM can send them to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the PRINTER1 and PRINTER2 operands of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace an associated LU table (ASLTAB). See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

### AUTHLEN

<table>
<thead>
<tr>
<th>Statement</th>
<th>Group, LU</th>
<th>Dependencies</th>
<th>Associated LU Table Name</th>
</tr>
</thead>
</table>
AUTHLEN = YES
VTAM will pass the specified transmission priority.

AUTHLEN = NO
VTAM will not pass the specified transmission priority, instead the default priority of medium will be passed.

Note: If the TRLE operand is coded, AUTHLEN is ignored. See “TRLE” on page 414 for more information.

BATCH

BATCH = NO
Specifies a high priority (suitable for interactive application programs).

BATCH = YES
Specifies a low priority.

Note: This operand applies to pre-V4R3 NCPs only. If you code the BATCH operand and the NCP is V4R3 or later, the operand is ignored.

See NCP, SSP, and EP Resource Definition Reference for more information about this operand.

CAPACITY
**CAPACITY** – number of bits

**statements:**
**dependencies:**
**range:**
- PU
- APPN only
- 1 - 1000Kb
- 1 - 1000Mb
- 1 - 100Gb

Specifies the effective capacity of the link that comprises the transmission group (TG). Specify the value in Kb per second (for example, 100K), Mb per second (for example, 100M), or Gb per second (for example, 10G). This number approximates the bits per second that the link can transmit (the transmission rate of the link, times the maximum load factor expressed as a percentage).

**Note:** Because the value for capacity is represented as a single byte, the precision of the specified number of bits might be lost. For example, numbers that are close (such as 100K and 101K) can be interpreted by VTAM, and displayed, as the same value. See Table 45 on page 722 for a list of CAPACITY values you can specify and their corresponding values when displayed.

TG characteristics are not valid with DYNTYPE=VN because DYNTYPE=VN defines the dynamic PU used for a VN connection, not the virtual routing node (VRN) and its associated TGs. The TG created with the dynamic PU is not reported to Topology and Routing Services and is not involved in route calculation. To determine the weight of the TGs associated with the VRN, specify the TG characteristics in the XCA major node or the NCP major node.

**CERTIFY**
**CERTIFY**

Specifies whether VTAM on behalf of the application program will verify authentication.

**CERTIFY=YES**
- Specifies SLU authentication (verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

**CERTIFY=NO**
- Specifies no SLU authentication (not verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

**CKEYNAME**

Specifies the name of the key-encrypting key (KEK) in the cryptographic key data set (CKDS) for the LU. Use CKEYNAME to encipher the session key that is sent in the BIND to the LU when a cryptographic session is established. CKEYNAME has the same format and restrictions as other VTAM names. For information on naming conventions, see "Format of definition statements" on page 8.

If you do not specify CKEYNAME, the cryptographic key name defaults to the name on the LU definition statement.

**Note:** Independent LUs will ignore the CKEYNAME keyword and use their default name.

**CLRSESSQ**
Specifies whether a dependent LU is forced into an inhibited session state when
the LU is disabled. When the LU is enabled, it is returned to normal status.
Sessions, however, are not connected.

**CLRSESSQ=**YES
   Specifies that the LU is forced into an inhibited session state. Both active and
   pending sessions are removed from the session queue.

**CLRSESSQ=NO**
   Specifies that the LU is not forced into inhibited session state. This is the
default.

**CONNTYPE**

```
   CONNTYPE = value_of_conntype_start_option

   CONNTYPE = APPN
   - LEN
   (1)
   CONNTYPE = APPN
```

**Notes:**

1. **CONNTYPE=APPN** is the only valid value if the TRLE operand is coded.

   **Statements:**
   PU

   **Dependencies:**
   APPN only

Specifies for a type 2.1 whether the connection is to be established as a LEN
connection or attempted as an APPN connection.

If you do not code the NODETYPE start option, the CONNTYPE operand is
ignored, and the connection will be a LEN connection.

**Note:** If the TRLE operand is coded, CONNTYPE must be specified as APPN or
allowed to default. See "TRLE" on page 414 for more information.

**CONNTYPE=APPN**
   Specifies that this connection can support parallel TGs, CP-CP sessions, and CP
   name change support.

   When you code **CONNTYPE=APPN**, XID rules for parallel TGs are enforced.
   So, if you have PUs with duplicate CPNAMEs, you should make them unique
   before this node becomes an APPN node.
**CONNTYPE=LEN**

Specifies that this connection does not support CP-CP sessions and CP name change, and that VTAM does not check for duplicate CPNAMEs.

Because CONNTYPE=LEN indicates that the PU is not APPN-capable, do not code any APPN-only operands on the same PU statement. If an APPN-only operand is coded with CONNTYPE=LEN, VTAM will issue a message warning of a parameter conflict.

Although you need to code CONNTYPE=APPN for a PU to be an APPN resource, this does not necessarily mean the PU will be APPN; a PU’s characteristics are determined at activation. However, coding CONNTYPE=LEN means that the PU can only be a LEN resource, even if, at activation, the XID indicates the PU can be an APPN resource.

**COSTBYTE**

![Diagram of COSTBYTE]

- **statements:** PU
- **dependencies:** APPN only
- **range:** 0–255

Specifies a cost-per-byte-transmitted to be associated with the TG. A value of 0 is the least expensive cost per byte and 255 is the most expensive.

TG characteristics are not valid with DYNTYPE=VN because DYNTYPE=VN defines the dynamic PU used for a VN connection, not the virtual routing node (VRN) and its associated TGs. The TG created with the dynamic PU is not reported to Topology and Routing Services and is not involved in route calculation. To determine the weight of the TGs associated with the VRN, specify the TG characteristics in the XCA major node or the NCP major node.

**COSTTIME**
Specifies a cost-per-unit-time to be associated with the transmission group (TG). A value of 0 is the least expensive cost per unit of time and 255 is the most expensive.

TG characteristics are not valid with DYNTYPE=VN because DYNTYPE=VN defines the dynamic PU used for a VN connection, not the virtual routing node (VRN) and its associated TGs. The TG created with the dynamic PU is not reported to Topology and Routing Services and is not involved in route calculation. To determine the weight of the TGs associated with the VRN, specify the TG characteristics in the XCA major node or the NCP major node.

**CPCP**

Specifies whether CP-CP sessions are supported on this connection. If you do not code CPCP, the value defaults to YES only if you specified YES for the CPCP start option or if you coded DYNTYPE=RTP on this PU. Otherwise, the default value is NO.

**DATMODE**
Specifies whether the physical unit communicates in half-duplex or full-duplex data mode. The associated line must support simultaneous transmission in both directions.

**Note:** If the TRLE operand is coded, DATMODE is ignored. See “TRLE” on page 414 for more information.

**DATMODE=FULL**
Indicates that the physical unit can communicate in full-duplex data mode.

**DATMODE=HALF**
Indicates that the physical unit can communicate in half-duplex data mode.

**DISCNT**

**DISCNT values:**

**DISCNT DELAY values:**

Specifies when VTAM should terminate its SSCP-LU and SSCP-PU sessions and, when DISCNT=YES or DISCNT=DELAY, whether to indicate “final-use” status in the DACTPU request unit when it deactivates a PU.

**DISCNT=(YES)**
DISCNT=(NO)
Specifies whether VTAM physically disconnects the PU when the last LU ends its session with its application program (that is, when there are no more LU-LU sessions).

For a PU on a switched link, disconnection means that the dial connection is broken (in effect, the telephone is hung up, to save telephone charges) and the SSCP-PU session is terminated. Disconnection on a switched link does not involve deactivating the PU or its LUs (that is, sessions can be requested with those LUs causing the physical connection to be re-established).

See z/OS Communications Server: SNA Network Implementation Guide for more information about disconnection.

DISCNT=(YES,F)
DISCNT=(YES,NF)
Specifies whether VTAM indicates “final-use” status in the DACTPU request unit when it deactivates a physical unit as a result of DISCNT=YES. This value does not apply when DISCNT=NO, nor does it have any effect on the VARY INACT command.

If you code F or use the default, “final-use” status is indicated and the connection can be ended. If you code NF, “not-final-use” status is indicated and the connection should not be ended.

Each device has its own requirements regarding “final-use” status. To determine whether to code F or NF for a given device, consult the appropriate installation publication for the device.

DISCNT=(DELAY,F,delay_time)
DISCNT=(DELAY,NF,delay_time)
DISCNT=(DELAY,,delay_time)
Specifies that VTAM disconnects the physical unit if no LU-LU sessions exist after the value specified for delay_time expires. If delay_time is not specified, the current value for the DISCNTIM start option is used. delay_time specifies the number of seconds that VTAM delays disconnecting this PU if no LU-LU sessions exist. A decimal value in the range 5–65535 is valid.

The character value IMMED may be coded for delay_time to allow an immediate disconnect.

Note:
1. A delay time value can also be assigned for this PU through the MODIFY DEFAULTS operator command.

For HPR Sessions: If this VTAM is a node in a High-Performance Routing (HPR) route and is connected to an IBM 37xx Communication Controller, an Open Systems Adapter or through an IP network using Enterprise Extender, NCP or VTAM monitors the session traffic on the link to determine whether to deactivate a PU. When DISCNT=(DELAY,delay_time), NCP or VTAM disconnects the PU if no session traffic has occurred during the time interval specified by delay_time. When DISCNT=(DELAY) or DISCNT=(YES), NCP or VTAM disconnects the PU if no session traffic has occurred during the time interval specified on the DISCNTIM start option.

DLOGMOD
DLOGMOD = default_logon_mode_table_entry_name

**Statements:**

GROUP, LU

**Dependencies:**

M OdETAB

Specifies the name of the default logon mode table entry used by default if one is not otherwise provided. If you do not code this operand and the name of a logon mode table entry is not otherwise provided, VTAM uses the first entry in the applicable logon mode table (specified on the MODETAB operand or used by default).

If you specify MODETAB, the entry must be in either the specified table or in ISTINCLM, an IBM-supplied logon mode table. If you do not specify MODETAB, the entry must be in ISTINCLM. For more information on logon mode table entries, see "Logon mode table" on page 1026.

**DWINOP**

**Statements:**

PU

**Dependencies:**

DYNTYPE=EE only

Specifies whether the PU is to be redialed when the connection is broken by INOP, when this model definition is used for a dynamic non-connection-network Enterprise Extender PU (DYNTYPE=EE must be coded). The values specified by the REDIAL and the REDDELAY operands determine if any additional redial attempts are made and the interval between each retry. Coding DWINOP=YES can reduce the need for operator intervention in reestablishing Enterprise Extender connections.

**DWINOP=NO**

Specifies that the physical unit is not redialed after it has INOPed.

**DWINOP=YES**

Specifies that the physical unit is redialed after it has INOPed.
Notes:

1. DYNADJCP=YES is the only valid value if the TRLE operand is coded.

**DYNLU**
Notes:

1 DYNLU=YES is the only valid value if the TRLE operand is coded.

Specifies whether dynamic allocation of CDRSC definitions is supported. If you do not code this operand, the default is the value you coded on the DYNLU start option.

If dynamic allocation of CDRSC definitions is not supported for a particular PU, a resource that was dynamically defined as a CDRSC cannot use that PU to connect to a session.

**Note:** If the TRLE operand is coded, DYNLU must be specified as YES or allowed to default. See "TRLE" on page 414 for more information.

**DYNLU=NO**

Specifies that dynamic allocation of CDRSC definitions is not supported. If you code DYNLU=NO and plan to connect to other nodes using CP-CP sessions, then redefine each node in the CDRSC major node. If you do not, CP-CP session setup fails.

**DYNLU=YES**

Specifies that dynamic allocation of CDRSC definitions is supported.

For subarea nodes (HOSTSA is specified), DYNLU=YES is meaningful only when the host CDRM is activated and CDRDYN=YES is specified on the CDRM definition statement.

For APPN-only nodes (HOSTSA is not specified), DYNLU=YES is meaningful even though a host CDRM cannot be activated.

**Note:** If you have coded a DYNLU value in the adjacent CP major node for the adjacent CP to which this physical unit is linked, that value overrides the DYNLU value you code on this PU definition statement. For additional DYNLU considerations, see "DYNLU" on page 23 under the adjacent control point major node.

**DYNTYPE**
DYNTYPE=EE
Specifies that this is the model definition to be used for dynamic non-connection-network Enterprise Extender PUs. Dynamic connection network PUs will use the DYNTYPE=VN model definition. The following information shows a list of valid operands when DYNTYPE=EE is coded:

- CAPACITY
- COSTBYTE
- COSTTIME
- CPCP
- DISCNT
- DWINOP
- DYNADJCP
- DYNLU
- MAXDATA
- MAXOUT
- MTU
- PDELAY
- PUTYPE
- REDDELAY
- REDIAL
- SECURITY
- SPAN
- TGN
- TGP
- UPARM1
- UPARM2
- UPARM3
- VTAMTOPO

DYNTYPE=RTP
Specifies that this is the model definition to be used for dynamic HPR PUs. The following information shows a list of valid operands when DYNTYPE=RTP is coded:

- CPCP
- DISCNT
- DYNLU
- DYNTYPE
- LIMRES
- MAXDATA
• MAXOUT
• PUTYPE
• SPAN
• VTAMTOPO

**DYNTYPE=VN**

Specifies that this is the model definition to be used for dynamic Connection
Network PUs. The following information shows a list of valid operands when
DYNTYPE=VN is coded:

• DISCNT
• DYNADJCP
• DYNLU
• DYNTYPE
• MAXDATA
• MAXOUT
• PUTYPE
• SPAN
• VTAMTOPO

See [z/OS Communications Server: SNA Network Implementation Guide](z/OS Communications Server: SNA Network Implementation Guide) for additional information on defining an Enterprise Extender Connection
Network.

**DYNTYPE=XCF**

Specifies that this is the model definition to be used for dynamic XCF PUs. The
following information shows a list of valid operands when DYNTYPE=XCF is
coded:

• CAPACITY
• CONNTYPE
• COSTBYTE
• COSTTIME
• CPCP
• DYNADJCP
• DYNLU
• DYNTYPE
• LIMRES
• PDELAY
• PUTYPE
• SECNET
• SECURITY
• SPAN
• TGP
• TOPO
• TRLE
• UPARM1
• UPARM2
• UPARM3
• VTAMTOPO
• VERALSID
• XNETALS

You can change the default values for the first four characters of the dynamic XCF local SNA PU names added to ISTLSCXF by specifying different characters as the name of the model PU. The default values for the first four characters of the dynamic TRLE names added to the TRL major node can be changed by specifying different characters on the TRLE operand of the model PU. See the name operand and the TRLE operand for the naming requirements for the PU names and the TRLE names.

**EEVERIFY**

```plaintext
EEVERIFY = value_of_eeverify_start_option

EEVERIFY  = ACTIVATE
EEVERIFY  = NEVER
EEVERIFY  = Number_of_minutes
```

*statements:* PU
*dependencies:* DYNTYPE=EE only
*range:* 1 - 1440 (in minutes)

Specifies whether VTAM should automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the EE connection. The EEVERIFY option might also specify the time interval during which VTAM should send the probe on active EE connections.

The EEVERIFY operand is valid only for the EE connection that has its associated PU dynamically created using this model PU definition. If you do not code this operand, VTAM uses the value coded on the EEVERIFY start option.

**EEVERIFY=ACTIVATE**

Specifies that VTAM should automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the EE connection. If one or more ports are not reachable or are blocked, then VTAM issues an error message and does not activate the EE connection.

**EEVERIFY=NEVER**

Specifies that VTAM should never automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the EE connection.

**EEVERIFY=Number_of_minutes**

Specifies that VTAM should automatically send an LDLC probe during the activation of the EE connection and that VTAM should continue to periodically send LDLC probes at intervals specified by the `Number_of_minutes` value while the connection is active. Valid values for `Number_of_minutes` are in the range 1 - 1440 minutes (1440 minutes specifies 1 day). If one or more ports are not reachable or are blocked at activation, then VTAM issues an error message and does not activate the EE connection. If one or more ports are not reachable after the connection is active, VTAM issues an error message, but the connection remains active.
For predefined Enterprise Extender connections, define this parameter on the PU
definition in the switched major node. For EE connection networks, define this
parameter on the connection network GROUP definition statements in the EE XCA
major node.

**ENCR**

statements:  
GROUP, LU

dependencies:  
none

Tells whether this logical unit has any special requirements for enciphering and
deciphering messages.

**ENCR=NONE**

Specifies that this logical unit cannot engage in cryptographic sessions.

**ENCR=OPT**

Specifies that this logical unit can engage in cryptographic sessions, but allows
the application program to determine whether to use cryptography.

**ENCR=REQD**

Specifies that VTAM must encipher all messages to and from this logical unit.
If ENCR=REQD, no sessions can be established with this logical unit unless the
host with which it is associated can handle cryptographic sessions.

**ENCR=SEL**

For a model major node, SEL has no meaning for logical units. If you specify
ENCR=SEL, ENCR=OPT is used instead.

**ENCRTYPE**
Specifies the minimum type of encryption that VTAM should use on behalf of the application when performing session level encryption.

**ENCRTYPE=DES**

Specifies that VTAM must use a minimum of DES encryption with an 8-byte key when performing session level encryption. This is the default.

**ENCRTYPE=TDES24**

Specifies that VTAM must use a minimum of Triple_DES encryption with a 24-byte key when performing session level encryption.

The SYSDEFed ENCRTYPE of a LU can be changed at runtime through the MODIFY SECURITY command. See z/OS Communications Server: SNA Operation for specific details. However, the ENCRTYPE value will not be able to be downgraded. In other words, if the application is currently operating at ENCRTYPE=TDES24, then the encryption level cannot be changed to ENCRTYPE=DES using the MODIFY SECURITY command.

**Note:** If downgrading the security level is preferred, the current major node being used must be deactivated and another major node value must be coded.

In addition, a user may use the LOGMODE TABLE to override the ENCRTYPE value of the SLU based on the logmode entry being used. The ENCRTYPE value will not be able to be downgraded from TDES24 to DES using the logmode table. Only TDES24 can be specified for the encryption level (ENCRTYP). See “MODEENT macroinstruction” on page 1027 for more information on MODEENT macroinstructions.

**FASTPASS**
statements: GROUP, LU
dependencies: TSO sessions only

Determines how sessions are established for application programs that issue the CLSDST macroinstruction with the PASS option as part of their session establishment procedure. The application program acts as a secondary logical unit (SLU).

The PLU acknowledges the capability of the SLU during session establishment. If you have coded the logon performance enhancement in the PLU’s application program, only a single BIND is issued to establish the session.

Some SLUs are incompatible with this enhancement. If your SLU is incompatible with this enhancement, code FASTPASS=NO.

Note: FASTPASS applies only to application programs (acting as SLUs) that establish sessions with TSO.

**FASTPASS=NO**

Specifies that an unenhanced logon procedure requiring multiple binds is performed for application programs that issue the CLSDST PASS macroinstruction with the PASS option as part of their session establishment procedure.

**FASTPASS=YES**

Specifies that the logon performance enhancement is performed for application programs that issue the CLSDST macroinstruction with the PASS option as part of their session establishment procedure.

**FEATUR2**
Contains certain device information for virtual logical units that are supported by the network terminal option (NTO).

**FEATUR2=EDATS**
**FEATUR2=NOEDATS**

Specifies whether this terminal has the extended data stream feature. You cannot use this operand for terminals attached by SDLC lines.

**FEATUR2=DUALCSE**
**FEATUR2=LOWERCSE**

Tells VTAM how to send alphabetical characters coded with the TEXT operand on a USMSG macroinstruction to a non-SNA terminal over the SSCP-LU session. This value does not affect non-alphabetical characters or any characters coded on the BUFFER operand of a USMSG macroinstruction.

Code LOWERCSE to indicate that alphabetical characters are sent to the terminal over the SSCP-LU session in lowercase. Code DUALCSE to indicate that VTAM sends all characters as they are coded in the USMSG macroinstruction.

**FEATUR2=MODEL1**
**FEATUR2=MODEL2**

Identifies the specific model number (Model 1 or 2) for this 3275, 3277, 3284, or 3286 component. Code MODEL1 for those devices that have a default screen or buffer size of 480 bytes. Code MODEL2 for those devices that have a default screen or buffer size of 1920 bytes.

This information is available to an application program as part of the device characteristics pertaining to this terminal. You can obtain those characteristics by using the INQUIRE macroinstruction. For more information on using the INQUIRE macroinstruction, see z/OS Communications Server: SNA Programming.

**FEATUR2=NOPRINTR**
**FEATUR2=PRINTR**

Tells VTAM whether this terminal has an attached IBM 3284 Model 3 printer. This operand is valid only if TERM=3275.
FEATUR2=SELPEN
Specifies whether this terminal supports a selector pen.

HPREEARB

\[ \text{HPREEARB} \rightarrow \text{PROGRESS} \]

statements: PU
dependencies: DYNTYPE=EE only

This operand specifies that the progressive-mode adaptive-rate-based (ARB) flow control algorithm is preferred for the High-Performance Routing (HPR) pipes that traverse the Enterprise Extender (EE) connections that use this model PU definition to dynamically create their associated PUs. Progressive mode is used only when the HPR pipe is a single-hop pipe (which includes a single physical hop across a 2-hop EE virtual routing node [VRN]) and the RTP partner supports progressive mode.

For dial-in EE connections that have their associated PUs dynamically created, define this parameter on the model major node (DYNTYPE=EE) PU definition statement. For predefined EE connections, define this parameter on the GROUP or PU definition in the switched major node.

HPRPSDLY
Specifies the time that elapses before RTP pipes enter the patch switch state.

The ARB flow control algorithm is sensitive to minor variations in the round-trip
time and to unpredictable response time from the partner. This sensitivity might
cause RTP endpoints to prematurely enter the path switch state. Although this
situation does not affect availability, it does consume CPU cycles and it also causes
a significant number of path switch messages to be written to the console log.

The HPRPSDLY operand is valid only for RTP pipes that are directly attached to
an Enterprise Extender (EE) connection that uses this model PU definition to
dynamically create its associated PU. This operand affects only path switches that
are the result of an unresponsive partner (path switches that generate the message
IST1818I PATH SWITCH REASON: SHORT REQUEST RETRY LIMIT EXHAUSTED). The
HPRPSDLY operand does not control path switches that are initiated as a result of
a TG INOP, F RTP commands, or the PSRETRY function. If you do not code this
operand, VTAM uses the value that you coded on the HPRPSDLY start option.

HPRPSDLY=ps_delay

Specifies the number of seconds that RTP pipes must delay prior to entering
the path switch state. During this time, the RTP endpoint periodically tries to
contact the partner in an effort to avoid switching paths. The value 0 indicates
that the RTP nodes enter the path switch state when a predetermined number
of retry attempts have been unsuccessful.

HPRPSDLY=EEDELAY

Specifies that VTAM calculates the number of seconds that RTP pipes must
delay prior to entering the path switch state. The value that is calculated
allows enough time for the EE keep-alive mechanism to cause the EE
connection to become inoperative if connectivity to the partner is lost.
Unnecessary path switches are avoided while EE determines whether there is a
loss of connectivity to the partner.

For predefined EE connections, define this parameter on the PU definition in the
switched major node. For EE connection networks, define this parameter on the
connection network GROUP definition statements in the EE XCA major node.

Result: When the EEDELAY value is specified, the value that is calculated for the
HPR path-switch delay might be as long as 253 seconds.

Tips:
- This operand can be useful when the RTP partner is on a node that is
  CPU-constrained or is running in a virtualized environment. In both of these
  situations, allowing additional time for the RTP partner to respond might avoid
  unnecessary processing associated with a path switch.
- If alternate routes exist, specifying a long delay time might cause unnecessary
delays for the session that are using this RTP pipe.
INCLUD0E

\[
\begin{array}{c}
\text{INCLUD0E} = \text{NO} \\
\text{INCLUD0E} = \text{YES}
\end{array}
\]

**statements:** PU

**dependencies:** none

Specifies whether VTAM provides LU names to the logical units defined under this PU. The names are provided by passing a CV0E, Network Name control vector on the ACTLU requests.

**INCLUD0E=NO**

Indicates that VTAM will not provide LU names to the logical units defined under this PU.

**INCLUD0E=YES**

Indicates that VTAM will provide LU names to the logical units defined under this PU.

IRETRY

\[
\begin{array}{c}
\text{IRETRY} = \text{NO} \\
\text{IRETRY} = \text{YES}
\end{array}
\]

**statements:** PU

**dependencies:** none

Determines whether the boundary NCP (the NCP to which the switched physical unit becomes connected) retries a polling operation immediately for the device if an idle-detect timeout follows a polling operation. For more information on this operand, see the NCP, SSP, and EP Resource Definition Reference.

**Note:** If the TRLE operand is coded, IRETRY is ignored. See "TRLE" on page 414 for more information.

LANTEST
LANTEST = LOCAL

LANTEST = GLOBAL

Restriction: If you specify MEDIUM=CSMACD (Ethernet) on the associated XCA PORT definition statement, the parameter has no meaning and is ignored.

This parameter enables the user to define how the XCA connection network nodes connected on the XCA LAN will be located. There are two types of TEST commands sent: LOCAL and GLOBAL. The LOCAL command flows only on the local token ring. The GLOBAL flows with a routing information field. This routing information field has the broadcast bit on, which indicates to route the test into the entire network.

LANTEST=LOCAL

Specifies that a TEST frame will be sent out first on the LOCAL token ring. If no response is received within the PORT TIMER interval, a GLOBAL TEST frame is sent. Each time the GLOBAL TEST receives no response within the PORT TIMER interval, another GLOBAL TEST will be sent. The GLOBAL test will be repeated up to four times, at which time the connection setup is terminated if no response is received.

LANTEST=GLOBAL

Specifies that only GLOBAL TEST frames will be used to locate the device. Each time the GLOBAL TEST receives no response within the PORT TIMER interval, another GLOBAL TEST is sent. The GLOBAL test will be repeated up to five times at which time the connection setup is terminated if no response is received.

LIMRES
Notes:

1. LIMRES=NO is the only valid value if the TRLE operand is coded.

Statements: PU

Dependencies: LIMQSINT; LU 6.2 only; PUTYPE=1 or 2

Specifies whether this physical unit is to be treated as a limited resource. A limited resource allows you to limit the use of some network connections. It also causes any sessions that traverse the resource to be deactivated if no conversation is active. A session can also be deactivated if the time limit you specify for the LIMQSINT operand on the APPL definition statement expires.

This operand applies only to LU 6.2 conversations and PU types 1 and 2.

Note: If the TRLE operand is coded, LIMRES must be specified as NO or allowed to default. See “TRLE” on page 414 for more information.

If you do not code a value for LIMRES, resources are not considered to be limited resources. Furthermore, sessions will remain active even after all conversations have ended.

You can also use LIMRES in conjunction with the LIMINTCP start option to specify the interval to retain a free session using a limited resource for some network management flows (excluding flows using the APPN control point sessions or dependent LU server sessions). If you do not code a value for the LIMINTCP start option, resources are not considered to be limited resources, and the sessions will remain active even after all conversations have ended.

LIMRES=NO

Specifies that this physical unit is not to be treated as a limited resource.

LIMRES=YES

Specifies that this physical unit is to be treated as a limited resource.

LOCADDR
LOCADDR — address

**statements:** GROUP, LU
**dependencies:** see note
**range:** 0–255 (PU Type 2)
0–63 (PU Type 1)

Specifies the logical unit’s local address at the physical unit.

For dependent logical units, the range of valid local addresses depends on the
PUTYPE of the associated physical unit.

*Note:* The value of LOCADDR depends upon the requirements of the device being
defined. Consult the appropriate component description manual for these
restrictions.

**LOGAPPL**

LOGAPPL — controlling_primary_lu

**statements:** GROUP, LU
**dependencies:** NetView or TSO

Identifies the name of the primary logical unit that this logical unit will
automatically log on to as a secondary logical unit when this logical unit is
activated. The name you specify can also be a USERVAR that will be resolved into
its current value before being used to establish or redrive the session with the
controlling primary logical unit.

The LOGAPPL operand cannot contain the name of an independent logical unit
because independent logical units cannot be the controlling logical units for an
automatic logon session.

To ensure retry of a pending automatic logon, code the AUTORTRY or AUTOTI
start options.

You can specify a network-qualified name for LOGAPPL. You cannot specify an
LUALIAS name for LOGAPPL.

*Note:*
1. If an error (INOP) occurs for the switched connection, VTAM reallocates the
LOGAPPL session and attempts to redial the logical unit to recover the
connection. In other words, the session will be terminated by breaking the dial
connection instead of by normal session termination using LOGOFF and
TERMSELF.
   If you do not code the LOGAPPL operand, the PU and LU must be activated
   manually.
2. If you specify the NetView or TSO program as the application program on the
LOGAPPL operand, you will get unpredictable results in recovery situations.
For more information on establishing sessions, see "z/OS Communications Server: SNA Network Implementation Guide.

**LOGTAB**

```
LOGTAB = interpret_table_name
```

*statements*: `GROUP, LU`

*dependencies*: `none`

Specifies the name of an interpret table that VTAM uses when processing logon requests originating from the logical unit. VTAM uses the interpret table to interpret the name received from an initiation request. See "Interpret table" on page 1017 for a description of how to define an interpret table.

**LUGROUP**

```
LUGROUP = lu_group_name
```

*statements*: `PU`

*dependencies*: `none`

Specifies the name of the model LU group that VTAM uses to select a model LU definition when dynamically defining an LU. This operand also indicates that the device supports dynamic definition of LUs.

**Note**: If the TRLE operand is coded, LUGROUP is ignored. See "TRLE" on page 414 for more information.

**LUSEED**
LUSEED = __lu_pattern_name__

**statements:**

**dependencies:**

Provides a 1–8 character pattern name that the SDDLU exit routine uses to generate an LU name when dynamically defining an LU.

This operand is valid only when you code the LUGROUP operand.

**Note:** If the TRLE operand is coded, LUSEED is ignored. See "TRLE" on page 414 for more information.

If you use the IBM-supplied SDDLU exit routine, this operand is required and must have the following format:

- The first character must be either alphabetical or one of the national characters (@ or $).
- The remaining characters can be either alphabetical, numerical, or national.
- There must be either two or three contiguous national # characters in the name. These characters represent holders, in which the logical unit's local address is substituted when an LU's name is generated. If you specify three # characters, the LU's logical address is substituted in decimal for the # characters. If you specify two # characters, the LU's logical address is substituted in hexadecimal for the # characters.

When used with a user-written SDDLU exit routine, this operand is optional and there are no restrictions on its format. For more information on the SDDLU exit, see "z/OS Communications Server: SNA Customization".

**MAXDATA**

For PU type 1:

```
MAXDATA = 261
MAXDATA = number_of_bytes
```
For PU type 1:

<table>
<thead>
<tr>
<th>statements:</th>
<th>PU dependency:</th>
<th>range:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>none</td>
<td>20-65535</td>
</tr>
</tbody>
</table>

For PU type 2:

\[
\text{MAXDATA} = \begin{cases} 
265 & \text{if PU type 1} \\
\text{number of bytes} & \text{if PU type 2}
\end{cases}
\]

<table>
<thead>
<tr>
<th>statements:</th>
<th>PU dependency:</th>
<th>range:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>none</td>
<td>20-65535</td>
</tr>
</tbody>
</table>

Specifies the maximum number of bytes that the physical unit can receive in one PIU or PIU segment, including the transmission header (TH) and request or response header (RH). To determine the maximum PIU (or PIU segment) size that the physical unit can receive, consult the component publications for the specific type of SDLC station represented by this PU definition statement.

The maximum amount of user data that VTAM sends to the physical unit in one PIU is:

- MAXDATA minus 2 bytes (for a PU type 1)
- MAXDATA minus 6 bytes (for a PU type 2)

**Note:** The RU part of the PIU contains the user data. When VTAM segments the PIU, the middle and last PIUs contain three more bytes of user data in the RU because there is no RH field. The value for the PIU varies with the device; for example, the 3276 requires MAXDATA=262.

MAXDATA is ignored for type 2.1 nodes attached through an NCP with peripheral node type 2.1 support. The type 2.1 node supplies the actual value used when the connection is established.

See *NCP, SSP, and EP Resource Definition Reference* for more information about physical units connected through an NCP.

If the TRLE operand is coded, MAXDATA is ignored. See "TRLE" on page 414 for more information.

**Note:**

1. In a local area network, the value for the MAXDATA operand on the PU definition statement must be less than or equal to the value for the MAXDATA operand on the PORT definition statement for the LAN major node to which the switched physical unit is connected.

2. NCP V4R3 changed the MAXDATA calculation for a type 1 physical unit. See *NCP, SSP, and EP Resource Definition Reference* for information about calculating a MAXDATA value for physical units on non-switched links.

**MAXOUT**
MAXOUT = 8

`max_number_of_plius`

**statements:** PU

**dependencies:** none

**range:** 1–127

Specifies the maximum number of PIUs or PIU segments (if the session parameters allow segmenting of data) that VTAM sends to the physical unit represented by this definition statement before requesting a response.

For performance reasons, you should code MAXOUT=7 if the physical unit represented by this definition statement can handle more than one PIU before sending a link-level response. To determine this, consult the component publications for the specific type of SDLC station.

**Note:** If the TRLE operand is coded, MAXOUT is ignored. See “TRLE” on page 414 for more information.

MAXSESS

`max_number_LU-LU_sessions`

**statements:** PU

**dependencies:** independent LUs attached to an NCP

**range:** 1–65535

Specifies the maximum number of concurrent LU-LU sessions in which an independent LU can participate. The MAXSESS value applies to each connection the LU is using to establish LU-LU sessions. For example, if MAXSESS=2 was specified for LUA and LUA appears on three PUs, LUA can have two sessions on each of the three PUs for a total of six sessions.

If you do not code MAXSESS or code a value outside the range, the MAXSESS value defined on the BUILD definition statement (in the NCP major node) will be used. Reference the NCP, SSP, and EP Resource Definition Reference for more information about this operand.

**Note:** If the TRLE operand is coded, MAXSESS is ignored. See “TRLE” on page 414 for more information.

MDLENT
Specifies the name of the model name table entry to be used for this logical unit.

If you code the MDLENT operand, you must also code the MDLTAB operand. If you omit MDLENT and include MDLTAB, VTAM uses the first entry in the table named in MDLTAB.

**MDLTAB**

Specifies the name of the model name table to be used for the logical unit. See "Model name table" on page 1054 for a description of how to define a model name table.

If you omit the MDLTAB operand, VTAM will not provide the name of the model name table to the application during LU-LU session initiation. However, you can provide the model name in other ways. You can enter it from the terminal on the MODEL operand of the LOGON command, or VTAM can send it to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the MODEL operand of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace a model name table. See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.
**MODETAB**

Specifies the name of a logon mode table to be used to correlate each logon mode name with a set of session parameters for the logical unit. The name you code must be the name of a logon mode table created as described in "Logon mode table" on page 1026. If you do not supply a logon mode table for the logical unit on the MODETAB operand, an IBM-supplied default logon mode table (ISTINCLM) is used. If you specify a table, both the table you specify and the default table are used.

**MTU**

Defines the maximum transmission unit (MTU), in bytes, that is used for dial-in Enterprise Extender connections with associated PUs that are dynamically created using this model PU definition. This MTU size is reduced by VTAM to account for header lengths such as the IP, UDP, and LLC headers that are necessary for EE traffic. If MTU is not specified, VTAM uses the value learned from the TCP/IP stack. This learned value is the minimum MTU size (minus IP and UDP header sizes) associated with all the defined TCP/IP interfaces that can use this local static VIPA. If an MTU size with a value is specified, VTAM uses the lesser of this mtu_size value and the value learned from TCP/IP. If the TCP/IP stack presents a value less than 768 bytes, VTAM sets the MTU to 768 bytes because this is the smallest packet size allowed by the HPR architecture. Typically, the MTU size for an EE connection is constant after the EE connection is established. However, if the MTU size of the TCP/IP stack changes, RTP pipes with endpoints on the same node as the TCP/IP stack dynamically detect these changes when their outbound packets are being transmitted. If the learned value is less than the mtu_size value, VTAM uses the learned value. Reasons for MTU size changes include the following information:

- New IP routes with different local MTU sizes become available
- Existing IP routes become unavailable
- Path MTU discovery is enabled for IPv4 or IPv6 EE connections (see "PMTUD start option" on page 911 for more information), and path MTU changes are discovered in the IP network

You can enable path MTU discovery for IPv4 and IPv6 EE connections (see "PMTUD start option" on page 911 for more information). If path MTU discovery is enabled, VTAM can dynamically learn about any MTU size changes that occur in the IP network. RTP pipes traversing Enterprise Extender connections can reduce their packet sizes to avoid IP fragmentation. In some environments, you...
might not be able to use path MTU discovery because firewalls must allow ICMP messages to be forwarded back to the host. In these cases, specify the MTU operand to control the maximum packet size that VTAM sends out on an Enterprise Extender connection.

**Tips:**

- If the *mtu_size* value is specified and IPv4 Enterprise Extender packets are being fragmented in the IP network, you might need to reduce the *mtu_size* value to account for optional headers that might be present.
- For EE connection networks, you can define the MTU parameter on the connection network GROUP definition statements in the EE XCA major node.
- For predefined Enterprise Extender connections, you can define the MTU parameter on the PU definition in the switched major node.

**name**

```
<table>
<thead>
<tr>
<th>name</th>
</tr>
</thead>
</table>

statements: VBUILD

<table>
<thead>
<tr>
<th>name</th>
</tr>
</thead>
</table>

statements: PU, LU
```

Provides the name associated with the resource being defined.

**On the VBUILD:**

`name` is optional. If a syntax or definition error is detected with this statement during activation of the major node, `name` appears in the message issued to identify the error.

The optional `name` you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node’s status.

**On the GROUP:**

Provides the required name for the GROUP statement.

**On the PU:**

Assigns the required minor node name of a model physical unit that is used in conjunction with the installation exit routine to build a dynamic switched definition.

For a model PU for XCF connectivity (DYNTYPE=XCF), `name` can contain up to five characters. The last character must be an asterisk (*). The first 1-4 characters determine the prefix for the PU names added to the dynamic local SNA major node (ISTLSXCF).

**Requirements:** The requirements for the name of an XCF model PU are different from the requirements for names of other resources. These requirements are:

- To use the default prefix of ISTP* for dynamic XCF local SNA PUs, you can specify a single asterisk (*).
- To specify a different prefix, the following rules apply:
  - The name specified must be from 2-5 characters.
The first character can be alphabetical (A-Z) or the national characters @, #, or $.
The last character must be an asterisk (*).
Any character between the first and last characters can be alphabetical (A-Z),
umerical (0-9), or the national characters @, #, or $.

For RTP, VN, and EE model PUs, name provides the required name for the PU statement.

On the LU:
Assigns the required minor node name of the model logical unit that is used in
conjunction with the installation exit routine to build a dynamic switched
definition.

NATIVE

statements: PU
dependencies: BN=YES; APPN only

Specifies whether this link station represents a connection to a native node.

Note: If the TRLE operand is coded, NATIVE is ignored. See “TRLE” on page 414
for more information.

If NATIVE is not specified on either the ADJCP definition statement or on the PU
definition statement, the two nodes negotiate their subnetwork affiliation during
connection establishment.
• If the NETIDs match, the connection defaults to a native connection.
• If the NETIDs are different, the connection defaults to a nonnative connection.

No subnetwork negotiations will occur if NATIVE has been specified; in this case
the value between the two nodes must match for connection to occur.

If you code NATIVE on both the ADJCP and PU definition statements, the values
must match. If a transmission group (TG) is already active, subsequent TG
activations must not have a specified or negotiated value for NATIVE that conflicts
with the value of NATIVE for the active TG. If there is a conflict, the TG
attempting activation is not allowed to become active. If the activation is for a
virtual-route-based transmission group then the SSCP-SSCP session will also fail.

NETID
NETID=(network_id)

NETID=(,NOXNETLS)

Specifies a 1–8 character network name.

NETID is used in conjunction with the XNETALS start option in support of nonnative network connection. See “XNETALS start option” on page 978 for information about XNETALS.

Note:
1. The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.
2. If the TRLE operand is coded, NETID is ignored. See “TRLE” on page 414 for more information.

NETID=network_id

Specifies that VTAM ensures that the connecting resource is within the network specified by NETID when the physical unit is active. If you omit NETID, VTAM dynamically acknowledges the network ID during connection establishment. If you code NETID, and nonnative network connection is not supported, the definition is rejected unless the operand specifies VTAM's network ID.

NETID=(,NOXNETLS)

Specifies that VTAM does not allow nonnative network attachment for LU-LU sessions for this PU. When NETID=(,NOXNETLS), VTAM indicates to the NCP that this PU uses VTAM's native network ID.

LU-LU sessions might fail if the NETID sent in the BIND from the LU is different than the NETID specified in VTAM (even though PU-PU connections can be established). If no NETID is included in the BIND, or if the NETID is identical to that specified by VTAM, the LU-LU session should activate.

If dynamic dial-out connections are required (that is, a session request drives the dial), NETID must be predefined. Otherwise, VTAM might not be able to determine the connecting network ID and the resultant sessions might fail. If leased connections are not established, and if NETID is not predefined, directory requests could fail because of VTAM's inability to determine the resource's actual network.

NN
Specifies whether the adjacent node is expected to be a network node.

- If a value for the NN operand is specified, it will be validated when a connection is attempted to the adjacent node. If the adjacent CP is not the type of node that is expected, then connection setup fails.
- If the NN operand is not specified, then the APPN capabilities of the adjacent node are identified and accepted when a connection is established.

**Note:** If the TRLE operand is coded, NN is ignored. See "TRLE" on page 414 for more information.

**NN=NO**

Specifies that the adjacent node is expected to be an end node.

**NN=YES**

Specifies that the adjacent node is expected to be a network node.

**Restriction:** In a peripheral subnetwork boundary configuration between a border node and a network node without border node function, the border node appears to the non-border node as an end node. In this case, coding NN=YES on the definition statement that represents the border node, a connection failure will occur. Avoid the failure by not coding the NN operand or by coding NN=NO.

**PACING**
Statements: GROUP, LU, PU
Dependencies: none
Range: 0–63

Specifies how pacing is handled between the logical unit and the boundary node to which it is connected.

PACING can sift down from the GROUP statement to the LU statement, but it does not sift from the GROUP statement to the PU statement, nor does it sift from the PU statement to the LU statement.

Note: You can also use the FIXED option on this operand. FIXED is an NCP option that allows you to override adaptive session pacing and used fixed pacing instead. To use the FIXED option, you must code (N,,FIXED). The second comma represents a minimum value that is used by some pre-ENA nodes.

For more information on pacing, see z/OS Communications Server: SNA Network Implementation Guide

Note: If a dynamic CDRSC (ILU) uses a dynamic PU created from a MODEL definition, the PACING values are obtained from the PU definition in the MODEL major node that was used to define the dynamic PU.

PASSLIM
Determines, for switched physical units connected through an NCP, the maximum number of contiguous PIUs that the NCP sends to the physical unit at one time.

**Note:** If the TRLE operand is coded, PASSLIM is ignored. See “TRLE” on page 414 for more information.

**PDELAY**

Specifies the maximum propagation delay of the link for the transmission group. Propagation delay represents the time needed for a signal to travel from one end of the link to the other.

See “TRLE” on page 414 for more information about coding PDELAY for XCF connectivity.

**PDELAY=NEGLIGIB**

Indicates a local area network delay (less than 0.48 milliseconds).

**PDELAY=TERREST**

Indicates telephone network delay (between 0.48 and 49.152 milliseconds).

**PDELAY=PACKET**

Indicates a packet-switched network delay (between 49.152 and 245.76 milliseconds).

**PDELAY=LONG**

Indicates a satellite delay (greater than 245.76 milliseconds).

TG characteristics are not valid with DYNTYPE=VN because DYNTYPE=VN defines the dynamic PU used for a VN connection, not the virtual routing node (VRN) and its associated TGs. The TG created with the dynamic PU is not reported to Topology and Routing Services and is not involved in route calculation. To
determine the weight of the TGs associated with the VRN, specify the TG characteristics in the XCA major node or the NCP major node.

**PUTYPE**

PUTYPE = 1

PUTYPE = 2

PUTYPE = 2 (1)

PUTYPE = 2

Notes:

1. PUTYPE=2 is the only valid value if the TRLE operand is coded.

Specifies the physical unit type. The physical unit type depends on the type of cluster controller (such as 3767, 3174, or AS/400°). To determine the physical unit type for a given device, see the component description manual for that device.

Note: If the TRLE operand is coded, PUTYPE must be specified as 2 or allowed to default. See “TRLE” on page 414 for more information.

For a type 2.1 physical unit, code PUTYPE=2.

The physical unit type for a LAN switched station must be 2.

For PU types 1 and 2, or type 2.1, the value of PUTYPE occupies bits 4–7 of the 48-bit station ID constructed by VTAM for switched network operation. For a description of the station ID, see “IDNUM” on page 611.

**REDDELAY**
REDDELAY = number_of_seconds_between_redials

Specifies the number of seconds between each attempt to redial for an INOPed connection with non-connection network EE dynamic PU. The default is 30 seconds. The minimum value that can be coded is one second.

**Attention:** Customers with a very large number of connections with a non-connection-network EE dynamic PUs should not use a small value for REDDELAY, as the amount of activity generated during sustained outages could have significant CPU impact.

REDIAL

For dial-out operations attempted when an INOP occurs for a connection with a non-connection-network EE dynamic PU, specifies the number of times dialing is to be retried before returning a dialing error to VTAM. The interval specified by the REDDELAY operand will be observed between redial attempts.

If you code 0, dialing is not to be retried. If you code FOREVER, dialing will be retried as necessary until either a connection is established or the dynamic PU created from this model PU is deactivated.

**REGISTER**

For independent LUs:
Specifies how a resource should be registered. The default depends on whether you code REGISTER for dependent or independent LUs.

**REGISTER=CDSRVR**
Indicates that an end node resource is registered to a network node server and directory resource registration is requested for it. A network node resource is registered at the central directory server. If the node is configured as a central directory server, this value has the same effect as NO.

**REGISTER=NETSRVR**
Indicates that the end node resource should be registered to its network node server, but that directory registration should not be requested for it.

For dependent LUs, NETSRVR is the default, and LOCADDR must be specified as a nonzero value.

Dependent LUs are not usually the targets of LU-LU session setup attempts and are therefore not likely to be the targets of an APPN LOCATE search. However, because end nodes cannot be searched, you can register their dependent LUs with their network node server.

**REGISTER=NO**
Indicates that the resource should not be registered.

For independent LUs, NO is the default, and LOCADDR must be specified as 0.
RESETLU = YES
  Specifies that during clean-up processing for dependent LUs, VTAM will issue an ACTLU request after the DACTLU request.

RESETLU = NO
  Specifies that during clean-up processing for LU-LU sessions, RESETLU=NO prevents a DACTLU-ACTLU from flowing to an LU.

SECNET

SECNET = NO
  Specifies that this physical unit is not associated with the secondary network.

SECNET = YES
  Specifies that this physical unit is associated with the secondary network. When SECNET=YES, data that is received by VTAM from this physical unit is flagged as requiring special problem-determination considerations when routed to a communication network management application.

Notes:
1  SECNET=NO is the only valid value if the TRLE operand is coded.

Statements: GROUP, LU
Dependencies: dependent LUs only

Specifies whether a DACTLU request followed by a ACTLU request will be sent during clean-up processing.

Specifies whether this physical unit is associated with a secondary network containing resources whose connectivity is not defined to VTAM.

This operand applies only if you have IBM 3710 Network Controllers or IBM 3174 token-ring attachments in your network.

Note: If the TRLE operand is coded, SECNET must be specified as NO or allowed to default. See "TRLE" on page 414 for more information.
SECURITY

Specifies the security level of the transmission group.

See "TRLE" on page 414 for more information about coding SECURITY for XCF connectivity. The following options range from the least secure level to the most secure level.

SECURITY=UNSECURE
    Specifies no security level.

SECURITY=PUBLIC
    Specifies a public switched network.

SECURITY=UNDERGRO
    Specifies an underground cable, not guarded.

SECURITY=SECURE
    Specifies a secure conduit, not guarded.

SECURITY=GUARDED
    Specifies a guarded conduit, physical only.

SECURITY=ENCRYPT
    Specifies link encryption.

SECURITY=SHIELDED
    Specifies a guarded conduit, physical and radiation shielded.

TG characteristics are not valid with DYNTYPE=VN because DYNTYPE=VN defines the dynamic PU used for a VN connection, not the virtual routing node (VRN) and its associated TGs. The TG created with the dynamic PU is not reported to Topology and Routing Services and is not involved in route calculation. To determine the weight of the TGs associated with the VRN, specify the TG characteristics in the XCA major node or the NCP major node.

SPAN
Defines a span of control for VTAM minor node resources. Code this operand if you are using the NetView program.

The NetView program checks the SPAN value, but VTAM ignores it.

**SSCPFM**

**SSCPFM=FSS**

Specifies that field-formatted RUs are supported.

**Note**: You might receive character-coded messages when SSCPFM=FSS.

**SSCPFM=USSNTO**

Specifies that this logical unit is supported by Network Terminal Option (NTO) in the boundary node, and character-coded RUs are supported over its SSCP-LU sessions.

**SSCPFM=USSCS**

Specifies that character-coded RUs are supported.

**SSCPFM=USS3270**

Allows printing of a user-defined logon message (USSMSG10) on BSC printers. Code SSCPFM=USS3270 for terminals attached to a controller defined as an SDLC 3271 (PUTYPE=1) or SDLC 3275 (PUTYPE=1).

**SSCPFM=USS3275**

Prevents printing of a user-defined logon message (USSMSG10) for a terminal defined as an SDLC 3275 (PU type 1) that might have an attached printer. Code SSCPFM=USS3275 to prevent the user-defined logon message from printing on BSC printers.

**SSCPFM=USS3780**

Coded for non-SNA devices supported with NTO that use the 3780 protocol. These devices use USS command facilities such as LOGON and IBMTEST and are supported only by Release 2 of NTO. See *NTM Planning, Migration, and Resource Definition* for more information.
Identifies, for a virtual logical unit supported through NTO, the device data stream compatible characteristics. An application program can find these characteristics by executing an INQUIRE DEVCHAR macroinstruction that names this virtual logical unit. The information is placed in the DEVAUXTP field of the DEVCHAR DSECT. (DEVAUXTP is an 8-byte storage area provided by the application program.) For more information on the DEVCHAR DSECT, see z/OS Communications Server SNA Programming.

**TGN**

Specifies the transmission group (TG) numbers that represent the connection between this node and the destination APPN nodes. The \(tg\_number\) value indicates a preferred TG number that might or might not be used, depending on how the partner nodes negotiate the number.

- **TGN\(=tg\_number\)**
  - Specifies a decimal number assigned to the transmission group. You can code up to four values in the order that you want them to be used.

- **TGN\(=\)ANY**
  - Coding TGN=ANY or not coding TGN allows the transmission group number to be negotiated.

**Tip:** Code ANY and up to three \(tg\_number\) values if you want the transmission group number to be negotiated when none of the specified \(tg\_number\) values are available. For example, when TGN=(1,2,3,ANY) is specified and TGN 1, TGN 2, and TGN 3 are already in use, the TG number is assigned by the partner or is negotiated to be greater than or equal to 21. ANY can be placed anywhere in the list.

**Tip:** If none of the specified values for TGN is in the range 1-20, TGN uses the default value ANY.
TGP

```
TGP=tgp_name
```

**statements:** PU  
**dependencies:** APPN only

Specifies the name of a transmission group (TG) profile. The characteristics of the TG profile (along with any modifiers on this statement) become the characteristics of the PU. If TGP is not specified or has not been activated when the PU becomes active, default TG characteristic are assigned.

See “APPN transmission group profile” on page 720 for more information on transmission group profiles.

TG characteristics are not valid with DYNTYPE=VN because DYNTYPE=VN defines the dynamic PU used for a VN connection, not the virtual routing node (VRN) and its associated TGs. The TG created with the dynamic PU is not reported to Topology and Routing Services and is not involved in route calculation. To determine the weight of the TGs associated with the VRN, specify the TG characteristics in the XCA major node or the NCP major node.

TOPO

```
TOPO=CONNECT
```

**statements:** PU  
**dependencies:** APPN only

Controls the way APPN connections using this link station are reported to APPN topology and routing services for inclusion into APPN functions.

**Note:** Except in the switched major node, the only value for TOPO is CONNECT. If you specify APPN on the CONNTYPE start option, it is not necessary to code TOPO, as you automatically get a value of CONNECT.

**TOPO=CONNECT**  
Specifies that APPN topology and routing services receives information about the connection and its characteristics when the connection is activated. When the connection is deactivated, APPN topology and routing services receives notification that the TG is inactive.

TRLE
Statements: PU
dependency: XCF connectivity only

Specifies that this model PU is for XCF connectivity.

Tip: A model for dynamic XCF local SNA PUs can be defined to override the XCF default values for the operands of the PU definition statement. It can also be used to specify the first four characters of TRLEs defined for XCF connectivity. This is true even if the node is a pure subarea node and the PU RDTE definition will not be created.

The `trle_name` can contain up to five characters. The first 1–4 characters determine the prefix for the TRLE names added to the TRL major node (ISTTRI). The last character must be an asterisk (*). If you code a single asterisk, the default prefix of ISTT* is used for the TRLEs added to the TRL major node (ISTTRI).

The requirements for the `name` of an XCF model PU and the `trle_name` are different from the requirements for names of other resources. These requirements are:

- To use the default prefix of ISTP* for dynamic XCF local SNA PUs or ISTT* for the TRLEs added to the TRL major node, you can specify a single asterisk (*).
- To specify a different prefix, the following rules apply:
  - The name specified must be from 2–5 characters.
  - The first character can be alphabetical (A–Z) or the national characters @, #, or $.
  - The last character must be an asterisk (*).
  - Any characters between the first and last characters can be alphabetical (A–Z), numerical (0–9), or the national characters @, #, or $.

See Table 31 for the operands and values that can be specified when TRLE is coded.

**Table 31. XCF model PU operands used with TRLE**

<table>
<thead>
<tr>
<th>Operand</th>
<th>Coding required?</th>
<th>Value required for XCF</th>
<th>Default for XCF</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Yes</td>
<td>User specified value</td>
<td>None</td>
</tr>
<tr>
<td>CAPACITY</td>
<td>No</td>
<td>User specified value</td>
<td>8K</td>
</tr>
<tr>
<td>CONNTYPE</td>
<td>No</td>
<td>APPN</td>
<td>APPN *</td>
</tr>
<tr>
<td>COSTBYTE</td>
<td>No</td>
<td>User specified value</td>
<td>0</td>
</tr>
<tr>
<td>COSTTIME</td>
<td>No</td>
<td>User specified value</td>
<td>0</td>
</tr>
<tr>
<td>CPCP</td>
<td>No</td>
<td>User specified value</td>
<td>Value of CPCP start option</td>
</tr>
<tr>
<td>DYNADJCP</td>
<td>No</td>
<td>YES</td>
<td>YES *</td>
</tr>
<tr>
<td>DYNLU</td>
<td>No</td>
<td>YES</td>
<td>YES *</td>
</tr>
<tr>
<td>LIMRES</td>
<td>No</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>PDELAY</td>
<td>No</td>
<td>User specified value</td>
<td>NEGLIGIB *</td>
</tr>
</tbody>
</table>
Table 31. XCF model PU operands used with TRLE (continued)

<table>
<thead>
<tr>
<th>Operand</th>
<th>Coding required?</th>
<th>Value required for XCF</th>
<th>Default for XCF</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUTYPE</td>
<td>No</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>SECNET</td>
<td>No</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>SECURITY</td>
<td>No</td>
<td>User specified value</td>
<td>SHIELDED *</td>
</tr>
<tr>
<td>SPAN</td>
<td>No</td>
<td>User specified value</td>
<td>None</td>
</tr>
<tr>
<td>TGP</td>
<td>No</td>
<td>User specified value</td>
<td>None</td>
</tr>
<tr>
<td>TOPO</td>
<td>No</td>
<td>User specified value</td>
<td>CONNECT</td>
</tr>
<tr>
<td>U Parm 1</td>
<td>No</td>
<td>User specified value</td>
<td>128</td>
</tr>
<tr>
<td>U Parm 2</td>
<td>No</td>
<td>User specified value</td>
<td>128</td>
</tr>
<tr>
<td>U Parm 3</td>
<td>No</td>
<td>User specified value</td>
<td>128</td>
</tr>
<tr>
<td>VERALSID</td>
<td>No</td>
<td>NO</td>
<td>NO</td>
</tr>
</tbody>
</table>

* The XCF default for this operand is different than the default for other model PUs.

**TYPE**

```
>>-TYPE+-+-MODEL-<<
```

**statements:**

```
VBUILD
```

**dependencies:**

```
none
```

Means that this VBUILD definition statement defines a model major node to VTAM. Dynamic switched definitions for previously undefined devices will be built using the PU and LU definitions contained within this node.

**UPARM1**
UPARM1

Statements: PU
Dependencies: APPN only
Range: 0–255

Specifies a user-defined operand value.

See z/OS Communications Server: SNA Network Implementation Guide for an example of using the UPARM operands.

TG characteristics are not valid with DYNTYPE=VN because DYNTYPE=VN defines the dynamic PU used for a VN connection, not the virtual routing node (VRN) and its associated TGs. The TG created with the dynamic PU is not reported to Topology and Routing Services and is not involved in route calculation. To determine the weight of the TGs associated with the VRN, specify the TG characteristics in the XCA major node or the NCP major node.

UPARM2

Statements: PU
Dependencies: APPN only
Range: 0–255

Specifies a user-defined operand value.

TG characteristics are not valid with DYNTYPE=VN because DYNTYPE=VN defines the dynamic PU used for a VN connection, not the virtual routing node (VRN) and its associated TGs. The TG created with the dynamic PU is not reported to Topology and Routing Services and is not involved in route calculation. To determine the weight of the TGs associated with the VRN, specify the TG characteristics in the XCA major node or the NCP major node.

UPARM3
UPARM3

```
UPARM3—user-defined_operand_value
```

**statements:** PU

**dependencies:** APPN only

**range:** 0–255

Specifies a user-defined operand value.

TG characteristics are not valid with DYNTYPE=VN because DYNTYPE=VN defines the dynamic PU used for a VN connection, not the virtual routing node (VRN) and its associated TGs. The TG created with the dynamic PU is not reported to Topology and Routing Services and is not involved in route calculation. To determine the weight of the TGs associated with the VRN, specify the TG characteristics in the XCA major node or the NCP major node.

USSTAB

```
USSTAB—ISTINCDT
USSTAB—uss_table_name
```

**statements:** GROUP, LU

**dependencies:** dependent LUs only

Specifies the name of a USS table that VTAM uses to process character-coded input that it receives from the logical unit.

A terminal user can issue a USS command with the LANGTAB operand. This causes a second USS table to be associated with the logical unit, which overrides the table specified with USSTAB. If you do not code USSTAB and a LANGTAB USS table is not in use, the IBM-supplied USS table (ISTINCDT) is used. For more information on USS tables, see "Unformatted system services tables" on page 1069.

USSTAB is valid for dependent logical units only.

VERALSID
VERALSID = NO

Notes:
1 VERALSID=NO is the only valid value if the TRLE operand is coded.

statements: PU
dependencies: APPN only

Specifies whether the adjacent link station name should be used to determine the link station. When specified as YES, the connection endpoints must supply the same name in the CV0EF7 vector that is appended to XID3s. The name supplied in CV0EF7 is the name of a switched PU or the name of a PU under a non-switched line.

Note: If the TRLE operand is coded, VERALSID must be specified as NO or allowed to default. See “TRLE” on page 414 for more information.

VPACING

VPACING = 2

Determines how VTAM paces the flow of data from VTAM to the boundary node to which the logical unit is connected.

VPACING can sift down from the GROUP statement to the LU statement, but it does not sift from the GROUP statement to the PU statement, nor does it sift from the PU statement to the LU statement.

Note: You can also use the FIXED option on this operand. FIXED is an NCP option that allows you to override adaptive session pacing and used fixed pacing instead. To use the FIXED option, you must code (N,,FIXED). The second comma represents a minimum value that is used by some pre-ENA nodes.

For more information about pacing, see z/OS Communications Server: SNA Network Implementation Guide.
Note: If a dynamic CDRSC (ILU) uses a dynamic PU created from a MODEL definition, the VPACING values are obtained from the PU definition in the MODEL major node that was used to define the dynamic PU.

**VTAMTOPO**

Specifies whether the VTAM topology agent reports the dynamic PUs created from PU model definitions to the CMIP manager application programs for SNA local topology. This will affect only dynamic PUs created after any changes to the model definition. It will not affect existing dynamic PUs.

An explicit specification of VTAMTOPO on a model PU definition is used to initialize a dynamic PU created from the model definition, as though the VTAMTOPO value had been coded on the dynamic PU definition. If no VTAMTOPO value is specified on the model PU definition, the reporting state is taken from the default for the ISTDSWMN major node (not connected) or the related NCP or XCA major node (when the dynamic PU is connected). Consequently, coding VTAMTOPO or VBUILD has no affect at this time.

Specifying VTM.VTAMTOPO allows VTAMTOPO to be handled appropriately by the NCP Definition Facility for versions of NDF that do not recognize VTAMTOPO natively.

**VTAMTOPO=REPORT**

Specifies that the VTAM topology agent reports this PU.

**VTAMTOPO=NOREPORT**

Specifies that the VTAM topology agent does not report this PU.

**XNETALS**
Determines whether the network ID of an attaching node or VTAM’s network ID is used for an adjacent nonnative node. Used to control how a nonnative node can attach to VTAM.

**Note:** The XNETALS option has changed in the way VTAM treats PUs that send in nonnative NETIDs in their XID. VTAM used to let the PU connect and treated it as if it sent in the same NETID that VTAM started with but that is no longer true. If the NETID that the PU sent in its XID is not the same as VTAM’s and XNETALS=NO or is defaulted, IST680I will be issued.

**XNETALS=NO**
- Indicates that VTAM’s network ID will be used to represent the connecting node’s network ID.
  
  During connection:
  - If the NETID operand is specified on the PU statement, then it must specify VTAM’s network ID.
  - If the NETID operand is not coded on the PU statement, then VTAM’s network ID is assumed and the adjacent node’s network ID is ignored.
  
  During LU-LU session establishment:
  - If the network ID for the originating LU is specified (during connection time) and is not identical to VTAM’s network ID, the session will not be established.
  - If the originating LU is not using a network qualified name, VTAM’s network ID is assumed and the session will be established.

**Note:** If the PU is a dependent LU server or requester, nonnative network connection is not allowed if XNETALS=NO is specified and the NETID operand is not coded.

**XNETALS=YES**
- Indicates that the network ID of the adjacent PU will be used for connection and session establishment.
  
  During connection:
  - If the NETID operand on the PU statement is coded, it will be used as the PU’s network ID. In this case, the network ID specified on the PU statement must match the network ID specified by the attaching node during connection processing.
  - If the NETID operand is not specified on the PU statement, then the network ID specified by the attaching node during connection processing will be assumed.
  
  During LU-LU session establishment:
  - If the network ID for the originating LU is specified and does not match the PU’s network ID, the session will not be established.
• If the originating LU name is not a network qualified name, the session will be established. VTAM will qualify the LU's name with the attaching node's network ID.

Note: Providing that XNETALS=YES is coded, you can prohibit nonnative network attachment for LU-LU sessions for a particular PU by coding XNETALS=NO on the GROUP, LINE or PU statements.

Network control program major node

To define a network control program (NCP) major node, code the VTAM and NCP definition statements detailed in this section, and file them in the appropriate VTAM definition library. The name under which the major node is filed must match the NCP load module name on the NEWNAME operand of the BUILD definition statement.

Information required by VTAM to communicate with the NCP and its attached devices is provided to VTAM by coding VTAM definition statements and NCP definition statements with VTAM operands. The VTAM definition statements and operands must be coded in the NCP major node, even though they provide no information to the NCP.

Definition statements in the NCP major node can be divided into the following categories:

VTAM only
Statement is processed by VTAM only.

VTAM and NCP
Statement contains operands that are processed by both VTAM and by the NCP generation process.

If there are any VTAM-only operands or restrictions on the statement, you will find a description in this document, but the description of the whole definition statement is in the NCP, SSP, and EP Resource Definition Reference.

VTAM and NCP, no VTAM requirements
Statement contains operands that are processed by both VTAM and by the NCP generation process, but it has no special VTAM requirements and should be coded as described in the NCP, SSP, and EP Resource Definition Reference.

NCP only, with VTAM requirements
Statement contains operands processed only by NCP, however there are special VTAM requirements.

See NCP, SSP, and EP Resource Definition Reference for a description and VTAM requirements.

The following list shows the definition statements used by VTAM in node.

PCCU VTAM only
BUILD VTAM and NCP
HOST VTAM only (requires NCP V4R3 or later)
SYSCNTRL VTAM and NCP
LUDRPOOL
  NCP only, with VTAM requirements
PU:DRPOOL
  NCP only, with VTAM requirements
PATH
  VTAM and NCP
GROUP
  VTAM and NCP
LINE
  VTAM and NCP
CLUSTER
  VTAM and NCP
TERMINAL
  VTAM and NCP
GROUP
  VTAM and NCP
LINE
  VTAM and NCP
PU
  VTAM and NCP
LU
  VTAM and NCP
NETWORK
  VTAM and NCP
FRSESET
  VTAM and NCP
GWNAU
  NCP only, with VTAM requirements
GENEND
  VTAM and NCP, no VTAM requirements

For a complete list of definition statements coded in the NCP major node and coding sequence information, reference the NCP, SSP, and EP Resource Definition Reference and the NCP, SSP, and EP Resource Definition Guide.

Identifying the communication controller

The programmed communication control unit (PCCU) definition statement identifies the communication controller where the NCP is loaded. The PCCU definition statement also contains information related to VTAM's support of the NCP.

PCCU definition statement

Code at least one PCCU definition statement for each VTAM that is to activate the NCP. It is required in VTAM systems and defines the VTAM functions that are provided for this NCP.

If more than one VTAM host will activate the NCP, you can code a PCCU definition statement for each host. Each PCCU definition statement describes the functions performed by the host it is associated with.

The SUBAREA operand (see “SUBAREA” on page 533) and the NETID operand (see “NETID” on page 511) associate the communication controller with a VTAM host when the NCP is activated. If each host provides the same function for the
NCP, only one PCCU definition statement is necessary, but the operator might need to override unique attachment requirements.

PCCU definition statements must be at the beginning of the NCP generation deck.

**HOST definition statement**

Code the HOST definition statement only for NCPs in channel-attached controllers. The following operands provide information that VTAM uses in communicating with the NCP over the channel:

- INBFRS
- MAXBFRU
- NETID
- SUBAREA

A channel-attached NCP requires one HOST definition statement for each host that activates the NCP. Do not define a HOST definition statement for a link-attached NCP. For more information about defining NCP channel support, see [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/SSSLBV_2.2.0/com.ibm.zos.v2r2.snetap0/zos/snetap0_hststatement.htm).

In an SNA-interconnected network, more than one HOST definition statement can exist in the NCP major node. VTAM uses the following steps to determine which HOST definition statement applies to the VTAM host and should be processed:

1. The VTAM host processes the first HOST definition statement whose SUBAREA operand matches VTAM’s and whose NETID matches VTAM’s.
2. If HOST definition statements as described in step 1 do not exist, the VTAM host processes the first HOST definition statement that has the same SUBAREA value as VTAM and that does not have a NETID specified.

**Note:** A match is made on the SUBAREA when the value coded for the SUBAREA operand matches the value of VTAM’s SUBAREA or when the SUBAREA operand is not coded and VTAM’s subarea is 1.

If parallel TGs are used, only one HOST definition statement is allowed for one host subarea destination. You cannot have different HOST definition statements for every TG.

**NCP definition statements for the NCP**

The following NCP definition statements used to define an NCP load module also contain VTAM-related information:

- BUILD
- SYSCNTRL
- HOST
- LUDRPOOL
- PUDRPOOL
- PATH
- NETWORK
- GWNAU
- GENEND

**BUILD definition statement**

Most of the operands on the BUILD definition statement relate only to the NCP. However, VTAM processes the following operands:
SYSCNTRL definition statement
SYSCNTRL identifies the dynamic control facilities included in the NCP. These facilities allow the NCP to execute requests from VTAM to change certain NCP parameters or to determine the status of resources such as lines and stations.

VTAM always needs these OPTION parameters for BSC 3270s:

**MODE**
- Set destination mode

**RIMM**
- Reset immediate

VTAM needs these OPTION parameters for the listed operator commands:

<table>
<thead>
<tr>
<th>OPTION</th>
<th>VTAM command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAKLIM</td>
<td>MODIFY NEGPOLL</td>
<td>Change line negative polling response limit</td>
</tr>
<tr>
<td>SESSION</td>
<td>MODIFY SESSION</td>
<td>Change session limit</td>
</tr>
<tr>
<td>SSPAUSE</td>
<td>MODIFY POLL</td>
<td>Change service-seeking pause</td>
</tr>
<tr>
<td>STORDSP</td>
<td>MODIFY NCPSTOR</td>
<td>Display NCP storage</td>
</tr>
</tbody>
</table>

LUDRPOOL and PUDRPOOL definition statements
VTAM support for switched terminals, dynamic reconfiguration, and primary addresses for independent logical units requires that the LUDRPOOL and PUDRPOOL definition statements be coded at NCP generation. To support these functions, VTAM issues request network address assignment (RNAA) RUs requesting the NCP to assign a physical unit or logical unit address to a resource specified by VTAM. The addresses come from the address pools generated as a result of the LUDRPOOL or PUDRPOOL definition statements. For more information on the LUDRPOOL and PUDRPOOL definition statements, see NCP, SSP, and EP Resource Definition Reference and NCP, SSP, and EP Resource Definition Guide.

PATH definition statements
PATH definition statements are required in the NCP major node to routes to the VTAM subarea node. The PATH statement specifies the explicit routes, adjacent subareas, and transmission groups used to send data to VTAM and other destination subareas. Also, data flow threshold values for the transmission groups are specified. For more information about the PATH definition statement for NCP, see NCP, SSP, and EP Resource Definition Reference and NCP, SSP, and EP Resource Definition Guide.
**NETWORK definition statement (SNA network interconnection only)**

You can specify a number of predefined networks in the NCP major node to identify each network attached to a gateway NCP. Those networks are defined using the NETID operand on the BUILD and NETWORK definition statements. You do not have to predefine all networks if you define a model NETWORK definition statement. To define a model network, code the COPIES operand instead of the NETID operand. You must code NETID or COPIES, but not both.

Code only one model NETWORK definition statement for each NCP.

The following operands are processed by VTAM:

- **COPIES**
- **MAXSUBA**
- **NETID**
- **SALIMIT**
- **SUBAREA**

**FRSESET definition statement**

The FRSESET (frame-relay switching equipment set) definition statement contains the pairings of the frame relay PUs for a frame relay line.

In a FRSESET statement, you specify the primary frame relay PU partners and their optional backups in the SUBPORTS operand. There are a minimum of two and a maximum of four PUs in a FRSESET definition statement. The FRELAY operand on the GROUP definition statement indicates the type of frame relay group.

The following FRSESET rules are validated by VTAM:

- The name on the FRSESET is required and unique in a domain. If not, a message is issued stating that a duplicate resource has been defined.
- Each puname represents a PU type 1. If not, a message is issued stating the puname is defined incorrectly.
- Each puname listed in SUBPORTS was defined in this NCP source prior to this FRSESET. If the puname has not been defined previously, a message is issued stating an undefined resource has been used in the FRSESET statement.
- The required punames are coded. If not, a message is issued stating that a required parameter has been specified on the FRSESET statement.
- Each puname coded on the SUBPORTS operand must be distinct within SUBPORTS and across FRSESET definition statements. If punames in SUBPORTS are not distinct, a message is issued stating a duplicate resource has been used. If a puname has been used in more than one FRSESET, a message is issued stating that a duplicate resource has been used.
- None of the punames in SUBPORTS is owned by different VTAMs. If so, a message is issued stating that the resource is not owned by this VTAM.
- None of the PUs in SUBPORTS is the first PU under a LINE. The first PU under a LINE is the local management interface (LMI). If a PU in SUBPORTS is the LMI, a message is issued stating that a resource used in the definition is not valid.
- Each puname in the source is in a FRSESET. If the puname is defined as a frame-relay PU and is not in a FRSESET, a message is issued identifying that the PU cannot be processed.
The following rule is not validated by VTAM, but is validated by NCP, which sends a sense code if the rule is not followed:

- Each puname in SUBPORTS is on a distinct frame-relay line.

**GWNAU definition statement (SNA network interconnection only)**

The GWNAU definition statement is an NCP-only definition statement that defines representations of cross-network logical units and SSCP's within the gateway NCP. VTAM does not use the definition statement when processing the NCP generation definitions. In some configurations, VTAM's ability to activate SSCP-SSCP sessions depends on proper use of the GWNAU definition statement. See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) for information about how this definition statement relates to session activation.

**GENEND definition statement**

The GENEND definition statement is an NCP-only statement used to delimit the end of the NCP definition statements.

**NCP definition statements for resources controlled by the NCP**

The following NCP definition statements are used to define lines, PU type 4 link stations, and peripheral devices to VTAM.

- GROUP
- LINE
- PU
- LU
- CLUSTER
- TERMINAL

Most of the operands on the above definition statements relate to the NCP only. However, there are some that are VTAM only or that have restrictions placed by VTAM. This document contains information about operands processed by VTAM. Use this document in conjunction with the NCP coding requirements described in the [NCP, SSP, and EP Resource Definition Reference](https://www.ibm.com). See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) for information about coding these operands.

**GROUP definition statement**

The GROUP definition statement begins a series of definition statements that define links and devices attached to the NCP with common characteristics.

The following items are NCP-only operands that should be looked at carefully for VTAM restrictions. See [NCP, SSP, and EP Resource Definition Reference](https://www.ibm.com) for information about coding these operands.

- CHANLNK
- LINEADD
- NPARSC
- VIRTUAL

**LINE definition statement**

The LINE definition statement identifies characteristics and functions associated with links (SDLC, BSC, start-stop, 370 I/O channel adapter connections, or channel) connecting the communication controller to another node.
The following items are NCP-only operands that should be looked at carefully for VTAM restrictions. See *NCP, SSP, and EP Resource Definition Reference* for information about coding these operands.

- ADDRESS
- CTYPE
- DIRECTN
- PAUSE
- POLLED
- SESSION
- TYPE

**PU definition statement**
The PU definition statement defines SDLC devices, channel adapter connections, and adjacent NCP connections to the NCP. The PU specifies functions and characteristics related to these connections.

The following items show the NCP-only operands that should be looked at carefully for VTAM restrictions. See *NCP, SSP, and EP Resource Definition Reference* for information about coding these operands.

- BNNSUP
- IRETRY
- MAXDATA
- MAXLU
- MAXOUT
- PASSLIM

**LU definition statement**
The LU definition statement specifies characteristics and functions for the logical units associated with a device.

The following NCP-only operand should be looked at carefully for VTAM restrictions. See *NCP, SSP, and EP Resource Definition Reference* for information about coding this operand.

- BATCH

**CLUSTER definition statement**
The CLUSTER definition statement defines clustered BSC stations attached to the NCP.

The following items are NCP-only operands that should be looked at carefully for VTAM restrictions. See *NCP, SSP, and EP Resource Definition Reference* for information about coding these operands.

- CTYPE
- PASSLIM

**TERMINAL definition statement**
The TERMINAL definition statement defines BSC and start-stop devices attached to the NCP.

The following items are NCP-only operands that should be looked at carefully for VTAM restrictions. See *NCP, SSP, and EP Resource Definition Reference* for information about coding these operands.
Table 32. NCP major node statements and operands

<table>
<thead>
<tr>
<th>Name, Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCCU</td>
<td>AUTODMP</td>
<td>456</td>
<td></td>
<td>NO</td>
<td>DUMPDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AUTOIPL</td>
<td>457</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>AUTOSYN</td>
<td>458</td>
<td></td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>BACKUP</td>
<td>459</td>
<td></td>
<td>NO</td>
<td>OWNER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDUMPDS</td>
<td>463</td>
<td></td>
<td>DUMPDS value</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CHANCON</td>
<td>464</td>
<td></td>
<td>COND</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CFGGDS</td>
<td>466</td>
<td></td>
<td></td>
<td>CFGGDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CFGPW</td>
<td>466</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CUADDR</td>
<td>471</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DELAY</td>
<td>472</td>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DUMPDS</td>
<td>475</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DUMPLD</td>
<td>476</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DUMPSTA</td>
<td>476</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>GWCTL</td>
<td>488</td>
<td></td>
<td>SHR</td>
<td>SNI only</td>
<td></td>
</tr>
<tr>
<td></td>
<td>INITEST</td>
<td>490</td>
<td></td>
<td>NO</td>
<td>3705 only</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LOADFROM</td>
<td>497</td>
<td></td>
<td>HOST</td>
<td>see description</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LOADSTA</td>
<td>499</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAXDATA</td>
<td>506</td>
<td></td>
<td>6535</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MDUMPDS</td>
<td>509</td>
<td></td>
<td>DUMPDS value</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NETID</td>
<td>511</td>
<td></td>
<td></td>
<td>see description</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NEWPATH</td>
<td>514</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OWNER</td>
<td>516</td>
<td></td>
<td></td>
<td>VTAM</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RNAME</td>
<td>526</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SAVEMOD</td>
<td>528</td>
<td></td>
<td>NO</td>
<td>3720 or 3745</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SUBAREA</td>
<td>533</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TGN</td>
<td>536</td>
<td></td>
<td>1 (Subarea)</td>
<td>ANY (APPN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VFYC</td>
<td>542</td>
<td></td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VFYLM</td>
<td>543</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name, Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUILD</td>
<td>LOADLIB</td>
<td>498</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NETID</td>
<td>525</td>
<td></td>
<td></td>
<td>XNETALS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SUBAREA</td>
<td>533</td>
<td>R</td>
<td></td>
<td>VTAM &amp; NCP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COSTAB</td>
<td>468</td>
<td></td>
<td></td>
<td>ISTSDCOS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAXSUBA</td>
<td>507</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NEWNAME</td>
<td>514</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PUNAME</td>
<td>522</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SALIMIT</td>
<td>527</td>
<td></td>
<td>255</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VTAMTOPO</td>
<td>547</td>
<td></td>
<td>REPORT</td>
<td>DIAL=YES</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name, Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
</table>
| SYSCNTRL                  |          |      |     |             |         | See the NCP, SSP, and EP Resource Definition Reference for operand details.
<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOST</td>
<td>INBFRS</td>
<td></td>
<td>489</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAXBFRU</td>
<td></td>
<td>505</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NETID</td>
<td></td>
<td>511</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SUBAREA</td>
<td></td>
<td>533</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LUDRPOOL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PUDRPOOL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NETWORK</td>
<td>NETID</td>
<td></td>
<td>511</td>
<td>R</td>
<td></td>
<td></td>
<td>SNI only; COPIES</td>
</tr>
<tr>
<td></td>
<td>SUBAREA</td>
<td></td>
<td>533</td>
<td>R</td>
<td></td>
<td></td>
<td>SNI only</td>
</tr>
<tr>
<td></td>
<td>COPIES</td>
<td></td>
<td>468</td>
<td></td>
<td></td>
<td></td>
<td>SNI only; NETID</td>
</tr>
<tr>
<td></td>
<td>COSTAB</td>
<td></td>
<td>468</td>
<td></td>
<td>ISTSDCOS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAXSUBA</td>
<td></td>
<td>507</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SALIMIT</td>
<td></td>
<td>527</td>
<td></td>
<td>255</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FRSESET</td>
<td>SUBPORTS</td>
<td></td>
<td>534</td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP; PU type 1</td>
</tr>
<tr>
<td>GWNAU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GROUP</td>
<td>LNCTL</td>
<td></td>
<td>496</td>
<td>R</td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td>DIAL</td>
<td></td>
<td>472</td>
<td></td>
<td>NO</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td>DYNPU</td>
<td></td>
<td>479</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DYNPUPFX</td>
<td></td>
<td>480</td>
<td></td>
<td>CN</td>
<td></td>
<td>DYNPU=YES</td>
</tr>
<tr>
<td></td>
<td>DYNVNPFX</td>
<td></td>
<td>481</td>
<td></td>
<td></td>
<td></td>
<td>VNGROUP</td>
</tr>
<tr>
<td></td>
<td>ECLTYPE</td>
<td></td>
<td>481</td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td>ENCR</td>
<td></td>
<td>482</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td>ENCRTYPE</td>
<td></td>
<td>482</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FRELAY</td>
<td></td>
<td>486</td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP; PU type 1</td>
</tr>
<tr>
<td></td>
<td>PHYSRSC</td>
<td></td>
<td>519</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PUTYPE</td>
<td></td>
<td>522</td>
<td></td>
<td>5</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td>SHM</td>
<td></td>
<td>531</td>
<td></td>
<td>NO</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td>VTAMTOPO</td>
<td></td>
<td>547</td>
<td></td>
<td></td>
<td></td>
<td>DIAL=YES</td>
</tr>
<tr>
<td>LINE</td>
<td>ANSWER</td>
<td></td>
<td>453</td>
<td>G</td>
<td>ON</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td>AUTO</td>
<td></td>
<td>455</td>
<td>G</td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td>AUTODL</td>
<td></td>
<td>456</td>
<td>G</td>
<td>YES</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td>BUFCAP</td>
<td></td>
<td>461</td>
<td>G</td>
<td>1</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td>CALL</td>
<td></td>
<td>461</td>
<td>G</td>
<td>IN</td>
<td></td>
<td>VTAM, NCP, &amp; DIAL=YES</td>
</tr>
<tr>
<td></td>
<td>CKEYNAME</td>
<td></td>
<td>465</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td>ENCR</td>
<td></td>
<td>482</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td>ENCRTYPE</td>
<td></td>
<td>482</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ISTATUS</td>
<td></td>
<td>492</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td>LINEAUT</td>
<td></td>
<td>494</td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>Name</td>
<td>Definition statement</td>
<td>Operands</td>
<td>Page</td>
<td>Req</td>
<td>Sift effect</td>
<td>Default</td>
<td>Restrictions, dependencies</td>
</tr>
<tr>
<td>----------</td>
<td>----------------------</td>
<td>----------</td>
<td>------</td>
<td>-----</td>
<td>-------------</td>
<td>----------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>LOCADD</td>
<td></td>
<td></td>
<td>499</td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>POLIMIT</td>
<td></td>
<td></td>
<td>520</td>
<td></td>
<td></td>
<td>(1,NOWAIT)</td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>PU</td>
<td></td>
<td></td>
<td>521</td>
<td></td>
<td></td>
<td>YES</td>
<td>VTAM</td>
</tr>
<tr>
<td>SWNORDER</td>
<td></td>
<td></td>
<td>535</td>
<td></td>
<td></td>
<td>CPNAME, FIRST</td>
<td>VTAM</td>
</tr>
<tr>
<td>TERM</td>
<td></td>
<td></td>
<td>536</td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>USE</td>
<td></td>
<td></td>
<td>540</td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP; USE=REDEF; USE=SPARE</td>
</tr>
<tr>
<td>VNGROUP</td>
<td></td>
<td></td>
<td>544</td>
<td></td>
<td></td>
<td></td>
<td>VNNAME; VTAM only; ECLTYPE=PHYSICAL</td>
</tr>
<tr>
<td>VNNAME</td>
<td></td>
<td></td>
<td>544</td>
<td></td>
<td></td>
<td></td>
<td>VNGROUP; VTAM only; ECLTYPE=PHYSICAL</td>
</tr>
<tr>
<td>VTAMTOPO</td>
<td></td>
<td></td>
<td>547</td>
<td></td>
<td></td>
<td></td>
<td>DIAL=YES</td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td>ADDR</td>
<td>451</td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP; DLCI</td>
</tr>
<tr>
<td>ANS</td>
<td></td>
<td></td>
<td>452</td>
<td>G,L</td>
<td>STOP</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>AUTHLEN</td>
<td></td>
<td></td>
<td>455</td>
<td>G,L</td>
<td>YES</td>
<td>Type 2.1</td>
<td>VTAM</td>
</tr>
<tr>
<td>CAPACITY</td>
<td></td>
<td></td>
<td>462</td>
<td>G,L</td>
<td>8K or 4M</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>CONNTYPE</td>
<td></td>
<td></td>
<td>467</td>
<td>G,L</td>
<td>CONNTYPE</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>COSTBYTE</td>
<td></td>
<td></td>
<td>468</td>
<td>G,L</td>
<td>0</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>COSTTIME</td>
<td></td>
<td></td>
<td>469</td>
<td>G,L</td>
<td>0</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>CPCP</td>
<td></td>
<td></td>
<td>469</td>
<td>G,L</td>
<td>CPCP</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>CPNAME</td>
<td></td>
<td></td>
<td>470</td>
<td>G,L</td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>DATMODE</td>
<td></td>
<td></td>
<td>471</td>
<td>G,L</td>
<td>HALF</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>DISCNT</td>
<td></td>
<td></td>
<td>473</td>
<td>G,L</td>
<td>(NO)</td>
<td></td>
<td>VTAM; VTAM &amp; NCP; ADDR</td>
</tr>
<tr>
<td>DLCI</td>
<td></td>
<td></td>
<td>474</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>DYNADJCP</td>
<td></td>
<td></td>
<td>477</td>
<td>G,L</td>
<td>DYNADJCP</td>
<td></td>
<td>APPN only; CPNAME</td>
</tr>
<tr>
<td>DYNLU</td>
<td></td>
<td></td>
<td>478</td>
<td>G,L</td>
<td>DYNLU</td>
<td></td>
<td>APPN only; CPNAME</td>
</tr>
<tr>
<td>ENCR</td>
<td></td>
<td></td>
<td>482</td>
<td>G,L,P</td>
<td>NONE</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>ENCRTYPE</td>
<td></td>
<td></td>
<td>482</td>
<td>G,L,P</td>
<td>DES</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>GP3174</td>
<td></td>
<td></td>
<td>487</td>
<td>G,L</td>
<td>NO</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>HPR</td>
<td></td>
<td></td>
<td>489</td>
<td>G,L</td>
<td>YES</td>
<td></td>
<td>VTAM; APPN only; NCP V7R3</td>
</tr>
<tr>
<td>INCLUDE0E</td>
<td></td>
<td></td>
<td>490</td>
<td></td>
<td>NO</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>INNPORT</td>
<td></td>
<td></td>
<td>491</td>
<td></td>
<td>NO</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>ISTATUS</td>
<td></td>
<td></td>
<td>492</td>
<td>G</td>
<td>ACTIVE</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>LIMRES</td>
<td></td>
<td></td>
<td>493</td>
<td>G,L</td>
<td>NO</td>
<td></td>
<td>VTAM; LIMQSINT; LU 6.2; PUTYPE=1, 2</td>
</tr>
<tr>
<td>LLERP</td>
<td></td>
<td></td>
<td>495</td>
<td></td>
<td>see description</td>
<td></td>
<td>VTAM; HPR=YES</td>
</tr>
<tr>
<td>Name</td>
<td>Definition statement</td>
<td>Operands</td>
<td>Page</td>
<td>Req</td>
<td>Sift effect</td>
<td>Default</td>
<td>Restrictions, dependencies</td>
</tr>
<tr>
<td>--------</td>
<td>----------------------</td>
<td>----------</td>
<td>------</td>
<td>-----</td>
<td>-------------</td>
<td>---------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>LMODADR</td>
<td></td>
<td></td>
<td>495</td>
<td></td>
<td></td>
<td>see description</td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LPDATS; SEGMENT</td>
</tr>
<tr>
<td>LPDA</td>
<td></td>
<td></td>
<td>503</td>
<td></td>
<td>ALLOW</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>LUGROUP</td>
<td></td>
<td></td>
<td>504</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>LUSEED</td>
<td></td>
<td></td>
<td>505</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>NATIVE</td>
<td></td>
<td></td>
<td>511</td>
<td>G,L</td>
<td></td>
<td></td>
<td>BN=YES</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>NETID</td>
<td></td>
<td></td>
<td>511</td>
<td></td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>NN</td>
<td></td>
<td></td>
<td>515</td>
<td>G,L</td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>OWNER</td>
<td></td>
<td></td>
<td>516</td>
<td>G,L</td>
<td></td>
<td></td>
<td>VTAM only; PUTYPE=1 or 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>PDELAY</td>
<td></td>
<td></td>
<td>519</td>
<td>G,L</td>
<td>NEGLIGIB or</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TERRESTR</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>PUDR</td>
<td></td>
<td></td>
<td>521</td>
<td>G,L</td>
<td></td>
<td>5</td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>PUTYPE</td>
<td></td>
<td></td>
<td>522</td>
<td>G,L</td>
<td></td>
<td>5</td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>RETRIES</td>
<td></td>
<td></td>
<td>525</td>
<td>G,L</td>
<td></td>
<td>5</td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>SECNET</td>
<td></td>
<td></td>
<td>528</td>
<td>G,L</td>
<td></td>
<td>NO</td>
<td>VTAM</td>
</tr>
<tr>
<td>SECURITY</td>
<td></td>
<td></td>
<td>529</td>
<td>G,L</td>
<td>UNSECURE</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>SEGMENT</td>
<td></td>
<td></td>
<td>530</td>
<td>G,L</td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>TGN</td>
<td></td>
<td></td>
<td>536</td>
<td>G,L</td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>TGP</td>
<td></td>
<td></td>
<td>537</td>
<td>G,L</td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>TOPO</td>
<td></td>
<td></td>
<td>538</td>
<td>G,L</td>
<td>CONNECT</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>UPARM1</td>
<td></td>
<td></td>
<td>538</td>
<td>G,L</td>
<td></td>
<td>128</td>
<td>APPN only</td>
</tr>
<tr>
<td>UPARM2</td>
<td></td>
<td></td>
<td>539</td>
<td>G,L</td>
<td></td>
<td>128</td>
<td>APPN only</td>
</tr>
<tr>
<td>UPARM3</td>
<td></td>
<td></td>
<td>539</td>
<td>G,L</td>
<td></td>
<td>128</td>
<td>APPN only</td>
</tr>
<tr>
<td>VERALSID</td>
<td></td>
<td></td>
<td>541</td>
<td>G,L</td>
<td>NO</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>XID</td>
<td></td>
<td></td>
<td>549</td>
<td>G,L</td>
<td></td>
<td>NO</td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>XNETALS</td>
<td></td>
<td></td>
<td>550</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
</tbody>
</table>

**Table 32. NCP major node statements and operands (continued)**

<table>
<thead>
<tr>
<th>Name</th>
<th>LU</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCADDR</td>
<td></td>
<td></td>
<td></td>
<td>500</td>
<td>R</td>
<td></td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>ASLENT</td>
<td></td>
<td></td>
<td></td>
<td>454</td>
<td></td>
<td></td>
<td></td>
<td>ASLTAB; VTAM</td>
</tr>
<tr>
<td>ASLTAB</td>
<td></td>
<td></td>
<td></td>
<td>454</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>CERTIFY</td>
<td></td>
<td></td>
<td></td>
<td>464</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>CKEYNAME</td>
<td></td>
<td></td>
<td></td>
<td>465</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>CLRSESSQ</td>
<td></td>
<td></td>
<td></td>
<td>465</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>DLOGMOD</td>
<td></td>
<td></td>
<td></td>
<td>474</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>ENCR</td>
<td></td>
<td></td>
<td></td>
<td>482</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>ENCRTYPE</td>
<td></td>
<td></td>
<td></td>
<td>482</td>
<td></td>
<td></td>
<td></td>
<td>VTAM; DES</td>
</tr>
<tr>
<td>FASTPASS</td>
<td></td>
<td></td>
<td></td>
<td>485</td>
<td></td>
<td></td>
<td></td>
<td>VTAM; TSO</td>
</tr>
<tr>
<td>FEATUR2</td>
<td></td>
<td></td>
<td></td>
<td>484</td>
<td></td>
<td></td>
<td></td>
<td>VTAM; (MODEL1, NOEDATS, NOPRINTR, NOSELPEN)</td>
</tr>
<tr>
<td>ISTATUS</td>
<td></td>
<td></td>
<td></td>
<td>492</td>
<td></td>
<td>ACTIVE</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>LOGAPPL</td>
<td></td>
<td></td>
<td></td>
<td>501</td>
<td></td>
<td></td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>LOGTAB</td>
<td></td>
<td></td>
<td></td>
<td>502</td>
<td></td>
<td></td>
<td></td>
<td>VTAM; dependent LUs</td>
</tr>
<tr>
<td>LUDR</td>
<td></td>
<td></td>
<td></td>
<td>503</td>
<td></td>
<td>YES</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
</tbody>
</table>
Table 32. NCP major node statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAXSESS</td>
<td></td>
<td></td>
<td>507</td>
<td></td>
<td>G,L,P</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>MDLENT</td>
<td></td>
<td></td>
<td>508</td>
<td></td>
<td>G,L,P</td>
<td></td>
<td>VTAM; MDLTAB</td>
</tr>
<tr>
<td>MDLTAB</td>
<td></td>
<td></td>
<td>508</td>
<td></td>
<td>G,L,P</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>MODETAB</td>
<td></td>
<td></td>
<td>510</td>
<td></td>
<td>G,L,P</td>
<td>ISTINCLM</td>
<td>VTAM</td>
</tr>
<tr>
<td>PACING</td>
<td></td>
<td></td>
<td>518</td>
<td></td>
<td>G,L,P</td>
<td>(1,1,ADAP)</td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>REGISTER</td>
<td></td>
<td></td>
<td>523</td>
<td></td>
<td>P</td>
<td>see description</td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>RESETLU</td>
<td></td>
<td></td>
<td>524</td>
<td></td>
<td>P</td>
<td>YES</td>
<td>APPN only</td>
</tr>
<tr>
<td>RESSCB</td>
<td></td>
<td></td>
<td>525</td>
<td></td>
<td>0</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td>SSCPFM</td>
<td></td>
<td></td>
<td>532</td>
<td></td>
<td>G,L,P</td>
<td>FSS</td>
<td>VTAM</td>
</tr>
<tr>
<td>TERM</td>
<td></td>
<td></td>
<td>536</td>
<td></td>
<td>P</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td>USSTAB</td>
<td></td>
<td></td>
<td>541</td>
<td></td>
<td>G,L,P</td>
<td>ISTINCDT</td>
<td>VTAM; dependent LUs</td>
</tr>
<tr>
<td>VPACING</td>
<td></td>
<td></td>
<td>545</td>
<td></td>
<td>G,L,P</td>
<td>(2,1,ADAP)</td>
<td>VTAM</td>
</tr>
<tr>
<td>name</td>
<td>CLUSTER</td>
<td>GPOOLL</td>
<td>487</td>
<td></td>
<td>G</td>
<td>ACTIVE</td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td>TERMINAL</td>
<td>TERM</td>
<td>536</td>
<td>R</td>
<td>G,L,C</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASLENT</td>
<td>454</td>
<td></td>
<td>G,L,C</td>
<td>ASLTAB</td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASLTAB</td>
<td>454</td>
<td></td>
<td>G,L,C</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BHSET</td>
<td>460</td>
<td></td>
<td>G,L,C</td>
<td></td>
<td>VTAM &amp; NCP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DLOGMOD</td>
<td>474</td>
<td></td>
<td>G,L,C</td>
<td></td>
<td>VTAM; MODETAB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FEATUR2</td>
<td>484</td>
<td></td>
<td>G,L,C</td>
<td>(MODEL1, NOEDATS, NOPRINTR, NOSELPEN)</td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISTATUS</td>
<td>492</td>
<td></td>
<td>G</td>
<td>ACTIVE</td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOGAPPL</td>
<td>501</td>
<td></td>
<td>G,L,C</td>
<td></td>
<td>VTAM; NetView or TSO</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOGTAB</td>
<td>502</td>
<td></td>
<td>G,L,C</td>
<td></td>
<td>dependent LUs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MDLENT</td>
<td>508</td>
<td></td>
<td>G,L,C</td>
<td>MDLTAB</td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MDLTAB</td>
<td>508</td>
<td></td>
<td>G,L,C</td>
<td></td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODETAB</td>
<td>510</td>
<td></td>
<td>G,L,C</td>
<td>ISTINCLM</td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SSCPFM</td>
<td>532</td>
<td></td>
<td>G,L,C</td>
<td>FSS</td>
<td>VTAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USSTAB</td>
<td>541</td>
<td></td>
<td>G,L,C</td>
<td>ISTINCDT</td>
<td>dependent LUs</td>
</tr>
</tbody>
</table>

Legend:

R Required operand.

G Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.

L Code this operand on the higher-level LINE definition statement to take advantage of the sift effect, or code it on this definition statement.

P Code this operand on the higher-level PU definition statement to take advantage of the sift effect, or code it on this definition statement.

C Code this operand on the higher-level CLUSTER definition statement to take advantage of the sift effect, or code it on this definition statement.
Network control program major node coding notes

After changing a VTAM-only or VTAM-restricted operand in an NCP definition statement, a copy of the updated NCP definition statements must also be filed as a VTAM major node.

Although you do not code all the information in the NCP definition statements, you should consider the possible requirements of VTAM and NCP when you code the NCP definition statements.

**NODETYPE start option**

If you do not code the NODETYPE start option, any APPN operands you code in this major node are ignored.

The NODETYPE start option allows APPN function. The combination of NODETYPE and the HOSTSA start option determines the various configurations of subarea node, interchange node, migration data host, network node, or end node.

**Dynamic change of operands for network control program major node**

You can dynamically change the following NCP major node operands by editing the VTAMLST data set and then issuing the VARY ACT,UPDATE=ALL command. You can specify the new operand value on a higher-level definition statement for sifting, if applicable.
<table>
<thead>
<tr>
<th>PU</th>
<th>LU</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAPACITY</td>
<td>ASLENT</td>
</tr>
<tr>
<td>COMRATE</td>
<td>ASLTAB</td>
</tr>
<tr>
<td>COSTBYTE</td>
<td>CLRSEQ</td>
</tr>
<tr>
<td>COSTIME</td>
<td>DLOGMOD</td>
</tr>
<tr>
<td>CPCP</td>
<td>ENCR</td>
</tr>
<tr>
<td>DISCNT</td>
<td>ENCRTYP</td>
</tr>
<tr>
<td>DLCI</td>
<td>FASTPASS</td>
</tr>
<tr>
<td>DYNADJCP</td>
<td>FEATUR2</td>
</tr>
<tr>
<td>DYNLU</td>
<td>ISTATUS</td>
</tr>
<tr>
<td>DYNWIND</td>
<td>LOGAPPL</td>
</tr>
<tr>
<td>HPRQLIM</td>
<td>LOGTAB</td>
</tr>
<tr>
<td>INNPORT</td>
<td>MDLEN</td>
</tr>
<tr>
<td>INTFACE</td>
<td>MDLTAB</td>
</tr>
<tr>
<td>IPQLIM</td>
<td>MODETAB</td>
</tr>
<tr>
<td>ISTATUS</td>
<td>REGISTER</td>
</tr>
<tr>
<td>LADDR</td>
<td>SSPFM</td>
</tr>
<tr>
<td>LIMRES</td>
<td>TERM</td>
</tr>
<tr>
<td>LUGROUP</td>
<td>USSTAB</td>
</tr>
<tr>
<td>LUSEED</td>
<td>VPACING</td>
</tr>
<tr>
<td>MAXSESS</td>
<td></td>
</tr>
<tr>
<td>NN</td>
<td></td>
</tr>
<tr>
<td>PDELAY</td>
<td></td>
</tr>
<tr>
<td>SECNET</td>
<td></td>
</tr>
<tr>
<td>SECURITY</td>
<td></td>
</tr>
<tr>
<td>SNETMASK</td>
<td></td>
</tr>
<tr>
<td>T1TIMER</td>
<td></td>
</tr>
<tr>
<td>T2TIMER</td>
<td></td>
</tr>
<tr>
<td>TGP</td>
<td></td>
</tr>
<tr>
<td>TOPO</td>
<td></td>
</tr>
<tr>
<td>UPPARM1</td>
<td></td>
</tr>
<tr>
<td>UPPARM2</td>
<td></td>
</tr>
<tr>
<td>UPPARM3</td>
<td></td>
</tr>
<tr>
<td>VERALSID</td>
<td></td>
</tr>
<tr>
<td>VPACING</td>
<td></td>
</tr>
</tbody>
</table>

For information about the following NCP operands, from the preceding list, see *NCP, SSP, and EP Resource Definition Reference*.

- COMRATE
- DLCI. DLCI has a VTAM restriction. See “DLCI” on page 474 for details.
- DYNWIND
- HPRQLIM
- INTFACE
- IPQLIM
- LADDR
- SNETMASK
- T1TIMER
- T2TIMER

To dynamically change the operand, the resource to which it applies must be inactive. For instance, an operand on an LU definition statement which receives its value from a PU definition statement can be dynamically changed only when the LU is inactive; the PU can remain active.
You can dynamically change the operands associated with tables (ASLTAB, LOGTAB, MDLTAB, MODETAB, and USSTAB) at any time without deactivating resources.

VTAM cannot recognize dynamically changed ADDRs or LOCADDRs on the first activation of an NCP, even when VARY ACT,UPDATE=ALL is used. (Warm starts are considered initial activations.)

The resource resolution table (RRT) from the NCP generation does not contain ADDR or LOCADDR information, so VTAM cannot recognize that the ADDR and LOCADDR values have been changed on the first activation of the NCP. However, on changes to ADDR and LOCADDR after the first activation, VTAM can compare the value in the definition statements to the value previously coded and recognize the change.

See z/OS Communications Server: SNA Network Implementation Guide and z/OS Communications Server: SNA Operation for more information about the VARY ACT,UPDATE=ALL command.

**Note:** For an NCP major node, you cannot add APPN capability or SDDLU support to a PU with the dynamic change function. If a PU has not been coded as APPN-capable, you cannot add the APPN operands to it. Likewise, if you have not coded LUGROUP (which allows SDDLU support) on a PU definition statement, you cannot add it with dynamic change.

**Network control program major node full syntax**

The full syntax for the network control program (NCP) major node follows. The syntax for each operand is repeated in "Network control program major node operand descriptions" on page 451.

**PCCU**
Notes:
1 TGN defaults to 1 for subarea.
2 TGN defaults to ANY for APPN.

BUILD

HOST
HOST

```
<table>
<thead>
<tr>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>INBFRS</td>
</tr>
<tr>
<td>number_of_buffers</td>
</tr>
</tbody>
</table>
```

MAXBFRU

```
| number_of_buffers |
```

```
<table>
<thead>
<tr>
<th>NETID</th>
</tr>
</thead>
<tbody>
<tr>
<td>network_id</td>
</tr>
</tbody>
</table>
```

SUBAREA

```
| 1 |

SUBAREA

```

 NETWORK

```
<table>
<thead>
<tr>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>NETWORK</td>
</tr>
<tr>
<td>COPIES</td>
</tr>
<tr>
<td>number_of_networks</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>COSTAB</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTSDCOS</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>COSTAB</th>
</tr>
</thead>
<tbody>
<tr>
<td>class-of-service_table_name</td>
</tr>
<tr>
<td>MAXSUBA</td>
</tr>
<tr>
<td>n</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>NETID</th>
</tr>
</thead>
<tbody>
<tr>
<td>network_id</td>
</tr>
<tr>
<td>SALIMIT</td>
</tr>
<tr>
<td>255</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>SUBAREA</th>
</tr>
</thead>
<tbody>
<tr>
<td>suba_addr</td>
</tr>
</tbody>
</table>
```

FRSESET

```
| 440 |
```

```
| z/OS V2R1.0 Communications Server: SNA Resource Definition Reference |
```

```
| 440 |
```

```
| 440 |
```

```
| 440 |
```
VTAMTOPO = NOREPORT
NOLLINES
NOSMPUS

LINE

name = LINE

ANSWER = ON
AUTO = auto_call_unit_address

AUTODL = YES
BUFCAP = 1
CALL = IN

AUTODL = NO
BUFCAP = number_of_plus
CALL = IN

CKEYNAME = name
ENCR = NONE
ENCTYPE = DES

CKEYNAME = crypto_key_name
ENCR = NONE
ENCTYPE = TDES24

ISTATUS = ACTIVE
LINEAUT = YES
LOCADD = address

ISTATUS = INACTIVE
LINEAUT = YES
LOCADD = address

POLIMIT = (1, NOWAIT)
PU = YES
RETRIES = (pause_interval, maximum_retries)

SWNORDER = (swnorder_start_option_value1, swnorder_start_option_value2)
TERM = type
USE = NCP
NCP
USE = EP
REDEF
SPARE

VTAMTOPO = REPORT
NOREPORT

PU

name = PU
ADDR = link_station_address

AUTHLEN = YES
CAPACITY = 4M

AUTHLEN = NO
CAPACITY = number_of_bits

CONNTYPE = value_of_conntype_start_option

CONNTYPE = APPN
LEN

CONNTYPE = APPN (1)

COSTBYTE = 0
COSTTIME = 0

COSTBYTE = cost_per_byte
COSTTIME = cost_per_unit_time
Notes:
1  CONNTYPE=APPN is the only valid value if the TRLE operand is coded.
2  CPCP defaults to YES for the model RTP PU.
3  CPCP defaults to the value of the CPCP start options for all PUs except the
    model RTP PU.
4 DYNADJCP=YES is the only valid value if the TRLE operand is coded.
5 HPR defaults to NO when the HPR start option is set to NONE, (RTP,NONE), or (ANR,NONE).
6 LIMRES=NO is the only valid value if the TRLE operand is coded.
7 For type 2.1, the default is LLERP=REQUIRED. For SDLC, X.25, and channels links, the default is required. Otherwise, the default is LLERP=NOTPREF.
8 NOXNETLS applies only to the PU definition statement.
9 NEGLIGIB is the default if the TRLE operand is coded.
10 TGN defaults to 1 for subarea.
11 TGN defaults to ANY for APPN.
12 VERALSID=NO is the only valid value if the TRLE operand is coded.

LU

```plaintext
>>> name = LU
>>> LOCADDR = address
>>> ASLENT = associated_lu_table_entry_name

>>> ASLTAB = associated_lu_table_name
>>> CERTIFY = NO
>>> CERTIFY = YES

>>> CKEYNAME = name
>>> CKEYNAME = crypto_key_name
>>> CLRSESSQ = NO
>>> CLRSESSQ = YES

>>> DLOGMOD = default_logon_mode_table_entry_name
>>> ENCR = NONE
>>> ENCR = OPT
>>> ENCR = REQD
>>> ENCR = SEL

>>> ENCRTYPE = DES
>>> ENCRTYPE = TDES24

>>> FASTPASS = YES
>>> FASTPASS = NO

>>> FEATUR2 = ( MODEL1 , NOEDATS , NOPRINTR , NOSELPEN )
>>> FEATUR2 = ( MODEL2 , EDATS , PRINTR , SELPEN )

>>> ISTATUS = ACTIVE
>>> ISTATUS = INACTIVE
>>> LOGAPPL = controlling_primary_lu
```
Notes:
1 REGISTER defaults to NETSRVR for dependent LUs.
2 REGISTER defaults to NO for independent LUs.
Network control program major node operand descriptions

The following information describes the operands you can code for the network control program (NCP) major node.

ADDR
ADDR = link_station_address

**statements:** PU
**dependencies:** VTAM and NCP; DLCI

Specifies the 2-character hexadecimal representation of the 8-bit address of a PU on a non-switched line. Any bit configuration except X'00' or X'FF' is valid.

ADDR cannot be coded on the first PU under a frame relay line, but needs to be coded on the other PUs.

**Note:** VTAM cannot recognize a dynamically changed ADDR on the first activation of an NCP, even when VARY ACT,UPDATE=ALL is used. (Warm starts are considered initial activations.)

The resource resolution table (RRT) from the NCP generation does not contain ADDR information, so VTAM cannot recognize that the ADDR value has been changed on the first activation of the NCP. However, on changes to ADDR after the first activation, VTAM can compare the value in the definition statement to the value previously coded and recognize the change.

**Note:** You can code ADDR or DLCI on frame relay PUs, but not both.

**ANS**
Determined whether the physical unit continues to operate when the NCP enters automatic network shutdown.

**ANS=CONTINUE or CONT**
- Specifies the LU-LU session is to continue when the NCP enters into automatic network shutdown mode. Code **ANS=CONTINUE**
  - If the terminal participates in XRF sessions
  - For frame relay PUs

When a session is continued and takeover occurs, if the LU taken over is a dependent logical unit, it must support ACTLU(ERP), or the session is terminated during takeover processing. An independent logical unit does not receive ACTLU, so this restriction does not apply. Note that there are several restrictions for session continuity in certain configurations. For information about these restrictions, see [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com). **ANS=STOP**
- Specifies that the LU-LU session will be terminated when the NCP enters automatic network shutdown mode.

See [NCP, SSP, and EP Resource Definition Reference](https://www.ibm.com) for code **ANS** for BSC and SS resources.

**ANSWER**
Specifies whether physical units can dial in to the NCP. ANSWER is valid only if the SDLC line has dial-in capability.

**ANSWER=OFF**

The physical units cannot dial in to the NCP, regardless of whether the line is active or inactive.

**ANSWER=ON**

The physical unit can dial in to the NCP when the line is activated.

The operator can issue a VARY ANS command after activation to change the answer state of the line. See [z/OS Communications Server: SNA Operation](https://www.ibm.com/support/knowledgecenter/en/POWER9总觉得_9/OS%20Communications%20Server%3A%20SNA%20Operation) for information about using the VARY ANS command.

### ASLENT

```
ASLENT—associated_lu_table_entry_name
```

Specifies the name of the associated LU table entry to be used for this logical unit.

If you code the ASLENT operand, you must also code the ASLTAB operand. If you omit ASLENT and include ASLTAB, VTAM uses the first entry in the associated LU table by default.

### ASLTAB
ASLTAB = associated_lu_table_name

**ASLTAB**

**statements:** GROUP, LINE, PU, LU, CLUSTER, TERMINAL

**dependencies:** VTAM only

Specifies the name of the associated LU table to be used for the logical unit. See "Associated LU table" on page 981 for a description of how to define an associated LU table.

If you omit the ASLTAB operand, VTAM will not provide the names of associated logical units to the application during LU-LU session initiation. However, you can provide the associated LU names in other ways. You can enter them from the terminal on the PRINTER1 and PRINTER2 operands of the LOGON command, or VTAM can send them to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the PRINTER1 and PRINTER2 operands of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace an associated LU table (ASLTAB). See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**AUTHLEN**

**statements:** GROUP, LINE, PU

**dependencies:** Type 2.1

Specifies whether VTAM should pass the transmission priority field specified by this PU to another PU.

**AUTHLEN=NO**

VTAM will not pass the specified transmission priority, instead the default priority of medium will be passed.

**AUTHLEN=YES**

VTAM will pass the specified transmission priority.
**AUTO**

Specifies the address of an automatic calling unit for this line.

When AUTO is coded, VTAM sends the dial digits to NCP. Otherwise, VTAM issues a console message to inform the operator that a manual dial should be done.

### AUTODL

- **AUTODL=NO**
  - Limits an X.21 line to direct-call-only status. As a result, when VTAM searches a line group for an available line, it cannot select a direct-call-only line for an auto-call operation.

- **AUTODL=YES**
  - Indicates that the NCP is to run the normal callout procedure for the line.

**Note:** Code AUTODL=YES for switched subarea connections using X.21 lines.
Determines whether a dump of communication controller storage is taken automatically after an unrecoverable failure in the NCP or the communication controller.

**AUTODMP=NO**
- Indicates that the VTAM operator is to be asked whether a dump should be taken.

**AUTODMP=YES**
- Indicates the dump is taken automatically.

The AUTODMP operand is valid only if you code a DUMPDS operand in this PCCU definition statement. Otherwise, no NCP dump can be taken. (See “DUMPDS” on page 473 for a description of DUMPDS.) After the dump, (or dumps, if more than one host takes a dump), VTAM interprets the AUTOIPL setting to determine whether it should load another copy of the same NCP and restart it.

If the operator specifies DUMPLOAD=YES in the VARY ACT command for a 3720 or 3745 communication controller, or if DUMPLD=YES is specified on this PCCU definition statement, the communication controller attempts to dump and load the NCP. If the dump and load by the communication controller is successful, the AUTODMP and AUTOIPL operands are ignored.

In an NCP for a communication controller that is attached by multiple channels, more than one PCCU definition statement can have AUTODMP=YES.

**AUTOIPL**
Determines whether VTAM loads another copy of the NCP and restarts it after either an unrecoverable failure occurs in the NCP or the communication controller.

**AUTOIPL=NO**
Indicates that the VTAM operator is prompted on whether VTAM should load and restart another copy of the same NCP.

**AUTOIPL=YES**
Indicates that another copy of the same NCP is automatically loaded and restarted.

In an NCP for a communication controller that is attached by multiple channels, more than one PCCU definition statement can have AUTOIPL=YES.

If the operator specifies DUMPLOAD=YES in the VARY ACT command for a 3720 or 3745 communication controller, or DUMPLD=YES on this statement, the communication controller attempts to dump and load the NCP. If the dump and load by the communication controller is successful, the AUTODMP and AUTOIPL operands are ignored.

**Note:** All VTAMs in the network must be at V3R4.1 or higher for the IPL/Dump Contention function to be used. Any mixture of back-level VTAMs including V3R4 VM, or V3R4 VSE make this function not valid.

**AUTOSYN**
Used to determine whether the NCP load module specified replaces the current load module in the communication controller. This operand applies only if the VTAM operator requests (or uses by default) LOAD=U on the VARY ACT command for an NCP. VTAM uses this operand when the specified load module name matches the current load module name (defined by NEWNAME on the BUILD definition statement).

**AUTOSYN=NO**
- The operator is asked whether the communication controller is to be refreshed with the specified load module.

**AUTOSYN=YES**
- The communication controller will not be loaded with the specified load module.

When NEWNAME does not match VTAM’s major node name, the value of VFYLM determines whether the controller is automatically reloaded. (See “VFYLM” on page 543 for a description of VFYLM.)

**BACKUP**
Determine whether this NCP’s resources can be taken over by a backup host if the owning host fails.

**BACKUP=NO**
- Indicates that the backup host cannot take over the NCP’s resources.

**BACKUP=YES**
- Indicates that a backup host can take over the NCP’s resources.
  
  If BACKUP=YES, resources are held in reserve unless they are owned by the host VTAM to which this NCP is defined. (Such resources have the same OWNER name as the one on this PCCU definition statement.) The host VTAM cannot use the resources until a VTAM VARY ACQ command is issued for this NCP.

**Note:** If the OWNER operand is not coded, BACKUP is ignored. (See “OWNER” on page 516 for a description of OWNER.)

**BHSET**
Block handler sets cannot be dynamically selected for BSC 3270s defined to VTAM. The BHSET operand can be coded by other owners of the terminal provided that the dynamic BHSET selection option (BHSASSC) is selected on the SYSCNTRL definition statement.

**BUFCAP**

Specifies the maximum number of PIUs that can be read at one time from a line without retransmissions, assuming each PIU is the maximum length allowed (MAXBFRU(MAX)×IOBUFSIZE). If an individual PIU length is smaller than the maximum size allowed for a PIU, more PIUs than BUFCAP’s value could be read in. The greatest number of buffers that VTAM will allocate is BUFCAP×MAXBFRU(MAX).

A retry value of 0 will not allow a type 2.0 or type 2.1 connection to become active. One retry is required to build the XID that will be exchanged.

**Note:** No matter what value you give for BUFCAP, VTAM uses only as many buffers as necessary to read in all incoming data.

For example, suppose that after initial startup, BUFCAP=35, IOBUFSIZE=100, MAXBFRU=(2,8), and all the PIUs are maximum size (each PIU would be 800 bytes long, requiring eight buffers). The NCP sends only MAXOUT PIUs at one time. If MAXOUT=7, the number of buffers required to have no retransmissions is 56. The BUFCAP operand allows up to 280 (35×8), but they will not be used.
Specifies whether VTAM, the device, or both can set up switched connections over the line to which this operand applies. The CALL operand acts as a security device.

**CALL=IN**
Specifies that the line is used only for terminal-initiated connections.

**CALL=INOUT**
Specifies that the line is used for connections initiated by terminals, in addition to connections initiated by VTAM, the VTAM operator, or application programs.

**CALL=OUT**
Specifies that the line is used only for outgoing calls.

**Note:**
1. If you code CALL=OUT, or if you code CALL=INOUT without coding an AUTO operand, outgoing calls are dialed by the operator in response to a VTAM console message. For switched subarea lines with CALL=OUT or CALL=INOUT, AUTO is required. There are no VTAM restrictions on CALL=OUT.
2. This operand is valid only if DIAL=YES on the GROUP definition statement, and applies only to line operation in network control mode.
3. CALL=OUT is not supported by non-SNA terminals using Network Terminal Option (NTO).

**CAPACITY**
CAPACITY = number of bits

**CAPACITY**

**Statements:**
- GROUP, LINE, PU

**Dependencies:**
- APPN only

**Range:**
- 1 - 1000Kb
- 1 - 1000Mb
- 1 - 100Gb

Specifies the effective capacity of the link that comprises the transmission group (TG). Specify the value in Kb per second (for example, 100K), Mb per second (for example, 100M), or Gb per second (for example, 10G). This number approximates the bits per second that the link can transmit (the transmission rate of the link, times the maximum load factor expressed as a percentage).

**Note:** Because the value for capacity is represented as a single byte, the precision of the specified number of bits might be lost. For example, numbers that are close (such as 100K and 101K) can be interpreted by VTAM, and displayed, as the same value. See [Table 45 on page 722](#) for a list of CAPACITY values you can specify and their corresponding values when displayed.

**CDUMPDS**
CDUMPDS

CDUMPDS

CDUMPDS = value_coded_on_dumpds

CDUMPDS = cps_dump_name

Statements: PCCU

Dependencies: none

Names the file that contains communication scanner processor (CSP) dumps.

cps_dump_name must be on a DD statement, which defines the file that is to contain the data from a storage dump of a communication controller.

If CDUMPDS is not coded, the CSP dumps are contained in the file named on the DUMPDS operand.

Note: CDUMPDS applies only to 3720, 3725, and 3745 communication controllers.

CHANCON

CHANCON

CHANCON = COND

CHANCON = UNCOND

Statements: PCCU

Dependencies: none

Determines whether VTAM sends a conditional or an unconditional channel contact request to the NCP when the host is to activate the NCP over a channel.

CHANCON=COND

VTAM allows the NCP to reject the host’s contact request, if the host’s subarea number is already in use by another host (attached to another one of the NCP’s channels). This results if duplicate subareas are assigned to more than one host (attached to more than one channel interface). Two hosts might have inadvertently been assigned the same number, or they might have been assigned duplicate numbers to allow mutual backup capabilities.

CHANCON=UNCOND

The contact request is unconditional; the NCP must accept it. If another host is using the same subarea number, the NCP breaks contact with that host.

Note: If you plan to define the channel to the NCP in a channel-attachment major node, this operand is replaced by the CHANCON operand on the PU definition statement for the channel-attachment major node. See “CHANCON” on page 92 for a description of the CHANCON operand in a channel-attachment major node.

CERTIFY
CERTIFY statements: GROUP, LINE, PU, LU

dependencies: CERTIFY is ignored unless ENCR is also active. Before enabling the CERTIFY function, ensure that the device or software providing the partner's LU support also supports the CERTIFY function.

Specifies whether VTAM on behalf of the application program will verify authentication.

CERTIFY=YES
   Specifies SLU authentication (verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

CERTIFY=NO
   Specifies no SLU authentication (not verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

CEKEYNAME

statements: LU

dependencies: dependent LUs only

Specifies the name of the key-encrypting key (KEK) in the cryptographic key data set (CKDS) for the LU. Use CKEYNAME to encipher the session key that is sent in the BIND to the LU when a cryptographic session is established. CKEYNAME has the same format and restrictions as other VTAM names. For information on naming conventions, see "Format of definition statements" on page 8.

If you do not specify CKEYNAME, the cryptographic key name defaults to the name on the LU definition statement.

Note: Independent LUs will ignore the CKEYNAME keyword and use their default name.

CLRSESSQ
Specifies whether a dependent LU is forced into an inhibited session state when
the LU is disabled.

When the LU is enabled, it is returned to normal status. The sessions, however, are
not reconnected.

**CLRSESSQ=NO**

Specifies that the LU is not forced into inhibited session state.

**CLRSESSQ=YES**

Specifies that the LU is forced into an inhibited session state. Both active and
pending sessions are removed from the session queue.

### CONFGDS

Specifies a 1–8 character data definition name that identifies the configuration
restart data set defined by the user for this major node. Include a DD statement
that has this data definition name in the VTAM start procedure.
CONFGPW = confgds_password

statements: PCCU
dependencies: CONFGDS

Specifies a 1–7 alphanumeric character password that VTAM uses to access the configuration restart data set. If you do not code CONFGPW, but VSAM requires it, VSAM prompts the VTAM operator for the correct password when VTAM attempts to open the data set.

Code CONFGPW only if you also code CONFGDS.

CONNTYPE

CONNTYPE = value_of_conntype_start_option

Notes:
1 CONNTYPE=APPN is the only valid value if the TRLE operand is coded.

specifies for a type 2.1 whether the connection is to be established as a LEN connection or attempted as an APPN connection.

If you do not code the NODETYPE start option, the CONNTYPE operand is ignored, and the connection will be a LEN connection.

CONNTYPE=APPN

This connection can support parallel TGs, CP-CP sessions, and CP name change support. When you code CONNTYPE=APPN, XID rules for parallel TGs are enforced. So, if you have PUs with duplicate CPNAMEs, you should make them unique before this node becomes an APPN node.

CONNTYPE=LEN

Specifies that this connection does not support CP-CP sessions and CP name change, and that VTAM does not check for duplicate CPNAMEs.

Although you need to code CONNTYPE=APPN for a PU to be an APPN resource, this does not necessarily mean the PU will be APPN; a PU’s characteristics are determined at activation. However, coding CONNTYPE=LEN means that the PU can only be a LEN resource, even if, at activation, the XID indicates the PU can be an APPN resource.
Because CONNTYPE=LEN indicates that the PU is not APPN-capable, do not code any APPN-only operands on the same PU statement. If an APPN-only operand is coded with CONNTYPE=LEN, VTAM will issue a message warning of a parameter conflict.

**COPIES**

```plaintext
COPIES = number_of_networks
```

*statements:* NETWORK

*dependencies:* SNI only; NETID

Specifies the number of unique networks that this network definition statement can support.

**Note:** NETID and COPIES cannot be coded on the same NETWORK definition statement.

**COSTAB**

```plaintext
COSTAB = ISTSDCOS
```

```plaintext
COSTAB = class-of-service_table_name
```

*statements:* BUILD, NETWORK

*dependencies:* none

Names a Class of Service (CoS) table for the network identified by NETID. If you omit this operand, the default CoS table is ISTSDCOS. This operand is processed regardless of which network corresponds to the NETWORK definition statement.

The CoS table name matches the name of a Class of Service table in the VTAM load module library. VTAM loads the table from the library, associates it with the network identified by the NETID operand, and uses it to resolve Class of Service names for cross-network sessions when the primary logical unit is represented by an alias address in the gateway NCP.

This table is also used to resolve a Class of Service name to a virtual route (VR) list when the operator issues the command:

```plaintext
DISPLAY ROUTE,ORIGIN=ncp subarea node,COSNAME=cos name
```

**COSTBYTE**
COSTBYTE

COSTBYTE = cost_per_byte

statements: GROUP, LINE, PU
dependencies: APPN only
range: 0–255

Specifies a cost-per-byte-transmitted to be associated with the transmission group (TG). A value of 0 is the least expensive cost per byte and 255 is the most expensive.

COSTTIME

COSTTIME = cost_per_unit_time

statements: GROUP, LINE, PU
dependencies: APPN only
range: 0–255

Specifies a cost-per-unit-of-time to be associated with the transmission group (TG). A value of 0 is the least expensive cost per unit of time and 255 is the most expensive.

CPCP
CPCP = value_of_cpcp_start_option

(1)

(2)

Notes:
1 CPCP defaults to YES for the model RTP PU.
2 CPCP defaults to the value of the CPCP start options for all PUs except the model RTP PU.

statements: GROUP, LINE, PU
dependencies: APPN only

Specifies whether CP-CP sessions are supported on this connection. If you do not code this operand, the value defaults to the value specified on the CPCP start option.

**CPCP=NO**
Indicates CP-CP sessions are not supported on this connection.

**CPCP=YES**
Indicates CP-CP sessions are supported on this connection.

**CPNAME**
Identifies the resource's owning control point name. To allow definition of casually connected resources, this operand can be fully qualified with a network ID.

**CUADDR**

Defines the 3-digit or 4-digit hexadecimal device number of the channel attachment for the communication controller in which the NCP runs.

If you omit CUADDR and the VTAM operator wants to activate the communication controller over a channel, the operator should provide an address in the U operand when issuing the VTAM VARY ACT command.

If no channel device address or link station contact points are coded (with the CUADDR and RNAME operands of the PCCU definition statement) and none are provided on the VARY ACT command, a link station adjacent to the communication controller must already be active and defined to the host before the operator can activate the controller.

**Note:** If you plan to define the channel to the NCP in a channel-attachment major node, this operand is replaced by the ADDRESS operand on the LINE definition statement for the channel-attachment major node. See "ADDRESS" on page 91 for a description of the ADDRESS operand in a channel-attachment major node.
Specifies to the NCP whether the physical unit communicates in half-duplex or full-duplex data mode. The associated line must support simultaneous transmission in both directions.

**DATMODE=FULL**
- Specifies that the physical unit communicates in full-duplex mode.

**DATMODE=HALF**
- Specifies that the physical unit communicates in half-duplex mode.

**DELAY**
- Specifies the maximum amount of time, in seconds and thousandths of seconds, that VTAM should wait after getting low priority (that is, PIU transmission priority 0 or 1) data to transmit.

For more efficient operation, code this value as high as possible. For maximum speed across the interface, set this value to 0. See z/OS Communications Server: SNA Network Implementation Guide for more information.

**Note:** If you plan to define the channel to the NCP in a channel-attachment major node, this operand is replaced by the DELAY operand on the PU definition statement for the channel-attachment major node. See “DELAY” on page 93 for a description of the DELAY operand in a channel-attachment major node.

**DIAL**
Specifies whether the lines in the group require a dialed (switched) or leased connection.

**DIAL=NO**
Specifies that the lines require a leased (non-switched) connection.

**DIAL=YES**
Specifies that the lines require a dialed connection. When DIAL=YES, the only operands you can code in the PU definition statements subordinate to the GROUP definition statement are PUTYPE, SPAN, and XID.

**DISCNT**

**DISCNT values:**

**DISCNT=(NO)**
If you code NO or use the default, VTAM terminates the SSCP-LU and SSCP-PU sessions when certain conditions are met. See z/OS Communications Server: SNA Network Implementation Guide for a discussion about those conditions.

**DISCNT=(YES)**
If you code YES, VTAM automatically ends the SSCP-LU and SSCP-PU sessions when the last LU-LU session is ended.

**DISCNT=(YES,F)**
**DISCNT=(YES,NF)**
specifies whether VTAM indicates a final-use status in the DACTPU request
unit when it deactivates a physical unit as a result of DISCNT=YES. This value
does not apply when DISCNT=NO, nor does it have any effect on the
VARY INACT command.

If you code F or use the default, final-use status is indicated and the
connection can be ended. If you code NF, not-final-use status is indicated and
the connection should not be ended.

Each device has its own requirements regarding final-use status. To determine
whether to code F or NF for a given device, consult the appropriate installation
publication for the device.

**DLCI**

```
DLCI — address
```

*statements: PU
*dependencies: VTAM and NCP; ADDR

DLCI specifies the DLCI address of the frame-relay subport or internet host. All
DLCI values associated with a single frame-relay physical line or a NTRI frame
handler logical line must be unique. For example, if you use DLCI X'30' on a
frame-relay physical line to identify a subarea terminating equipment subport
(TESP) or a frame handler subport (FHSP), address X'30' cannot be used by a
peripheral device to establish an incoming logical peripheral connection to NCP
over that same physical line.

You should avoid using DLCI X'20' (decimal 32) when you define frame-relay
peripheral links, because peripheral devices often use DLCI X'20' for incoming
calls. When a subarea link TESP or FHSP uses this DLCI, the peripheral devices
must be reconfigured to use a different DLCI.

**Note:** You can code DLCI or ADDR on frame relay PUs, but not both.

See *NCP, SSP, and EP Resource Definition Reference* for more information about the
DLCI operand.

**DLOGMOD**
DLOGMOD = default_logon_mode_table_entry_name

**Statements:** GROUP, LINE, PU, LU, CLUSTER, TERMINAL

**Dependencies:** VTAM only; MODETAB

Specifies the name of the logon mode table entry used by default if one is not otherwise provided. If you do not code this operand and the name of a logon mode table entry is not otherwise provided, VTAM uses the first entry in the applicable logon mode table (specified on the MODETAB operand or used by default).

If you specify MODETAB, the entry must be in either the specified table or in ISTINCLM, an IBM-supplied logon mode table. If you do not specify MODETAB, the entry must be in ISTINCLM. For more information on the logon mode entries, see "Logon mode table" on page 1026.

**DUMPDS**

DUMPDS = dump_name

**Statements:** PCCU

**Dependencies:** none

Specifies one of three operands (the others are CDUMPDS and MDUMPDS) that name dump files for this communication controller. DUMPDS applies to any communication controller. CDUMPDS and MDUMPDS apply only to 3720, 3725, and 3745 communication controllers.

- **DUMPDS** names the file that contains a storage dump of a communication controller. For a 3720, 3725, or 3745, this file also contains communication scanner processor (CSP) dumps and maintenance and operator subsystem (MOSS) dumps if you have not set up another file for that purpose.
- For a 3720, 3725, or 3745, **CDUMPDS** names the file that contains CSP dumps.
- For a 3720, 3725, or 3745, **MDUMPDS** names the file that contains MOSS dumps.

`dump_name` must be on a DD statement, which defines the file that is to contain the data from a storage dump of a communication controller.

Code DUMPDS if any of the following items are true:
- **AUTODMP=YES** on this PCCU definition statement.
- **AUTODMP=NO** on this PCCU definition statement and the VTAM operator answers YES to the resulting prompt for a dump.
- The VTAM operator issues the MODIFY DUMP command without the DUMPDS operand.

To format and print the data in a dump file, use the NCP independent dump utility program described in the *NCP, SSP, and EP Diagnosis Guide*. To display an NCP dump that is stored on a 3720 or 3745 communication controller, use the `DISPLAY NCPSTOR` operator command described in *Z/OS Communications Server: SNA Operation*. 
Indicates whether an NCP that has abended should be automatically dumped to the 3720 or 3745 communication controller disk, and then reloaded from that disk.

**DUMPLD=NO**

Specifies that the NCP that has abended is not automatically dumped and then reloaded from that disk.

**DUMPLD=YES**

Specifies that the NCP that has abended is automatically dumped and then reloaded from that disk.

**Note:**

1. The value of DUMPLD is used only if you do not specify DUMPLOAD on the VARY ACT command, and only if a load is actually performed.
2. If you code LOADFROM=EXTERNAL and SAVEMOD=NO on the PCCU definition statement, the initial load will not use the SAVEMOD value.
3. If you code LOADFROM=HOST and SAVEMOD=NO on the VARY ACT command, and code DUMPLD=YES on the PCCU definition statement, the value of DUMPLD is not used for the initial load.
4. If you specify DUMPLD=YES and allow LOADFROM to default to HOST and SAVEMOD to default to NO, DUMPLD is forced to NO unless you specify LOADFROM=EXT on the VARY ACT command.

See [z/OS Communications Server: SNA Operation](#) for additional information about the VARY ACT command.

**DUMPSTA**
DUMPSTA = link_station_name

statements: PCCU
dependencies: none

Specifies an adjacent link station, channel- or link-attached, that is used to perform dump operations. If you do not use DUMPSTA on either the PCCU definition statement or the VTAM VARY ACT command, VTAM chooses a default link station whenever a dump is necessary. See z/OS Communications Server: SNA Network Implementation Guide for information about how VTAM chooses a default link station.

For a channel-attached communication controller, if the link station name is not defined (in the channel attached major node), VTAM assigns a link station name by taking the device number and adding “-S”. If the device number is 3-digits, a leading zero is added to the name. For example, the device number 0C2 has the link station name of 00C2-S.

You can designate the same link station for both dumping and loading.

For more information on link stations and use of the VTAM VARY ACT command, see z/OS Communications Server: SNA Operation.

DYNADJCP
Notes:

1. DYNADJCP=YES is the only valid value if the TRLE operand is coded.

   *statements:* GROUP, LINE, PU
   *dependencies:* APPN only; ignored if CPNAME coded.

Specifies whether an adjacent control point can be created dynamically for the adjacent node.

**Note:** This operand is ignored if CPNAME is coded on the physical unit.

**DYNADJCP=NO**
   Indicates that an adjacent control point cannot be dynamically created.

**DYNADJCP=YES**
   Indicates that an adjacent control point can be dynamically created.

**DYNLU**
Specifies whether dynamic allocation of CDRSC definitions is supported. If you do not code this operand, the default is the value you coded on the DYNLU start option.

This operand is valid only when PUTYPE=1 or PUTYPE=2.

If dynamic allocation of CDRSC definitions is not supported for a particular PU, a resource that was dynamically defined as a CDRSC cannot use that PU to connect to a session.

**DYNLU=NO**

Specifies that dynamic allocation of CDRSC definitions is not supported.

If dynamic allocation of CDRSC definitions is not supported for a particular PU, an independent logical unit that was dynamically defined using CDRSC definitions over another adjacent link station cannot use that particular PU to connect to a session.

**DYNLU=YES**

Specifies that dynamic allocation of CDRSC definitions is supported.

For subarea nodes (HOSTSA is specified), DYNLU=YES is meaningful only when the host CDRM is activated and CDRDYN=YES is specified on the CDRM definition statement.

For APPN-only nodes (HOSTSA is not specified), DYNLU=YES is meaningful even though a host CDRM cannot be activated.

**Result:** The DYNLU value is associated with an adjacent APPN node when the first link that adjacent node is activated. When the DYNLU value is associated with the adjacent node, that value is propagated to all other links to that adjacent node as they are activated, regardless of the DYNLU value specified for each link. See Table 2 on page 24 for more information about determining the source of the DYNLU value assigned to an adjacent CP and attached resources.

**DYNPU**
Specifies whether a PU is to be dynamically allocated when the calling PU cannot be identified during a switched call-in operation. DYNPU applies to APPN and subarea PUs.

A PU created by the DYNPU operand will use the switched major node PU operand defaults, except for the following operands which will use the values noted:

- MAXOUT=8
- ANS=CONT
- DISCNT=(YES,F)
- DYNADJCP=YES

A model PU definition can be created to customize the characteristics of dynamically created connection network PUs. Use the DYNTYPE=VN operand for the model PU definition in the model major node.

**DYNPU=NO**

Specifies that dynamic PUs are not to be used.

**DYNPU=YES**

Specifies that dynamic PUs can be used.
DYNUPFX = CN

Specifies a 2-character identifier used to create an 8-byte PU name when a dynamic PU is created for the link station. The two characters are used at the beginning of the PU name.

Note: This operand is valid only when DYNPU is YES.

DYNUPFX=CN

Indicates the beginning of the PU name is CN. The default value enables you to quickly identify the name of a dynamically defined PU.

DYNUPFX=string

 Allows you to create your own identifier. Specify two characters.

   If you create your own identifier instead of using the default, follow the naming conventions for VTAM as stated in "Coding VTAM definitions" on page 8.

DYNVPNFX

Specifies the first two characters of the PU name assigned when a dynamic connection network PU is created. VTAM concatenates the characters specified to form a VTAM-generated character sequence.

Note: This operand is valid only if this GROUP was identified by the VNGROUP operand on a LINE definition statement in the same NCP Major Node.

DYNVPNFX=string

 Allows you to specify the first two characters of the name assigned to dynamic connection network PUs. If you specify your own identifier (using this operand) instead of using the default, you must follow VTAM naming conventions. See "Restrictions on names" on page 14 for naming conventions.

ECLTYPE
VTAM uses the ECLTYPE operand and DIAL=YES to detect lines used for peripheral devices that are connected through NCP/Token-Ring interface (NTRI) support.

If you have NCPs connected over a token ring, you can use ECLTYPE to define a private frame-relay network over the token ring without requiring additional physical lines.


**ENCR**

Specifies whether this logical unit has any special requirements for enciphering and deciphering messages.

**ENCR=NONE**
- Specifies that this logical unit cannot engage in cryptographic sessions.

**ENCR=OPT**
- Specifies that this logical unit can engage in cryptographic sessions, but allows the application program to determine whether to use cryptography.

**ENCR=REQD**
- Specifies that VTAM must encipher all messages to and from this logical unit. If you code ENCR=REQD, no session can be established with this logical unit unless the host with which it is associated can handle cryptographic sessions.

**ENCR=SEL**
- Has no meaning for logical units. If you code ENCR=SEL, ENCR=OPT is used instead.
Specifies the minimum type of encryption that VTAM should use on behalf of the application when performing session level encryption.

**ENCRTYPE=DES**

Specifies that VTAM must use a minimum of DES encryption with an 8-byte key when performing session level encryption. This is the default.

**ENCRTYPE=TDES24**

Specifies that VTAM must use a minimum of Triple_DES encryption with a 24-byte key when performing session level encryption.

**Note:** The ENCRTYPE value coded on the GROUP, PU, or LINE will be sifted down to the appropriate resource (LU).

The SYSDEFed ENCRTYPE of a LU can be changed at runtime through the MODIFY SECURITY command. See z/OS Communications Server: SNA Operation for specific details. However, the ENCRTYPE value will not be able to be downgraded. In other words, if the application is currently operating at ENCRTYPE=TDES24, then the encryption level cannot be changed to ENCRTYPE=DES using the MODIFY SECURITY command.

**Note:** If downgrading the security level is preferred, the current major node being used must be deactivated and another major node value must be coded.

In addition, a user may use the LOGMODE TABLE to override the ENCRTYPE value of the SLU based on the logmode entry being used. The ENCRTYPE value will not be able to be downgraded from TDES24 to DES using the logmode table. Only TDES24 can be specified for the encryption level (ENCRTYP). See “MODEENT macroinstruction” on page 1027 for more information about MODEENT macroinstructions.

**FASTPASS**
FASTPASS specifies how session establishment is performed for application programs that issue the CLSDST macroinstruction with the PASS option as part of their session establishment procedure. The application program acts as a secondary logical unit (SLU).

The PLU acknowledges the capability of the SLU during session establishment. If you have coded the logon performance enhancement in the PLU's application program, only a single BIND is issued to establish the session.

Some SLUs are incompatible with this enhancement. If your SLU is incompatible with this enhancement, code FASTPASS=NO.

**Note:** FASTPASS applies only to application programs (acting as SLUs) that establish sessions with TSO.

**FASTPASS=**YES

Specifies that the logon performance enhancement is performed for application programs that issue the CLSDST macroinstruction with the PASS option as part of their session establishment procedure.

**FASTPASS=**NO

Specifies that an unenhanced logon procedure requiring multiple binds is performed for application programs that issue the CLSDST PASS macroinstruction with the PASS option as part of their session establishment procedure.

**FEATUR2**
FEATUR2 = (MODEL1, NOEDATS, NOPRINTR, NOSELPEN)

FEATUR2 = ()

MODEL2 EDATS PRINTR SELPEN

FEATUR2 = DUALCSE
FEATUR2 = LOWERCSE

FEATUR2 = EDATS
FEATUR2 = NOEDATS

Tells VTAM whether this terminal has the extended data stream feature. You cannot use this operand for terminals attached by SDLC lines.

Coding EDATS for a BSC line will cause the NCP to place the line in transparent text mode.

FEATUR2 = MODEL1
FEATUR2 = MODEL2

Identifies the specific model number (Model 1 or 2) for this 3275, 3277, 3284, or 3286 component. Code MODEL1 for those devices that have a default screen or buffer size of 480 bytes. Code MODEL2 for those devices that have a default screen or buffer size of 1920 bytes.

This information is available to an application program as part of the device characteristics pertaining to this terminal. You can obtain the characteristics using the INQUIRE macroinstruction. For more information about using the INQUIRE macroinstruction, see "Z/OS Communications Server: SNA Programming"

FEATUR2 = NOPRINTR
FEATUR2 = PRINTR

Specifies whether this terminal has an attached IBM 3284 Model 3 printer. This operand is valid only if you code TERM=3275 or if TERM=3275 is assumed by NCP definition statement sequencing for this statement.

FEATUR2 = NOSELPEN
FEATUR2 = SELPEN

Specifies whether this terminal supports a selector pen.

Describes the machine features for a specific link-attached non-SNA terminal. See Planning for NetView, NCP, and VTAM for a description of start-stop terminal planning considerations.

When coding the FEATUR2 operand on the PU and LU definition statements, see NTO Planning, Migration, and Resource Definition for more information.
FRELAY

FRELAY = (PHYSICAL)
    - - PERIPHERAL
    - - SUBAREA
    - - ANY

FRELAY = (LOGICAL)
    - - PERIPHERAL
    - - SUBAREA
    - - IP

---

**Statements:**

- GROUP

**Dependencies:**

- VTAM and NCP; PU type 1 only

---

Indicates the type of frame-relay group and the types of logical lines supported.

The names of PUs under the GROUP and LINE statements in a frame-relay group must be unique.

In addition, every frame handler PU under a physical frame-relay line must be put in a FRSESET definition statement.

When a frame-relay PU is dynamically defined but not included in a FRSESET definition statement, the resource is validated by VTAM, but will not be activated because one of the requirements for defining a frame-relay PU is that the PU needs to be included in a FRSESET definition. Frame-relay PUs not included in a FRSESET will generate messages informing you of the requirement. Update the definition list to include a FRSESET definition for the frame-relay PUs that are not included in one, replace the existing definition list with the revised one, and issue a V NET,ACT,UPDATE=ALL.

**FRELAY=PHYSICAL**

Indicates frame-relay physical lines. The types of logical lines supported by the physical line is determined by the second suboperand (PER, SUB, ANY).

- Code FRELAY=(PHYSICAL, SUBAREA | SUB) to define a physical line that supports logical subarea lines only.
- Code FRELAY=(PHYSICAL, PERIPHERAL | PER) to define a physical line that supports logical peripheral lines only.
- Code FRELAY=(PHYSICAL, ANY) to define a physical line that supports both logical peripheral and logical subarea lines.

If you code FRELAY=PHYSICAL and do not specify SUBAREA, PERIPHERAL, or ANY, the default for the second suboperand depends on the line address specified by ADDRESS on the LINE definition statement. See NCP, SSP, and EP Resource Definition Reference for information about the ADDRESS operand.

- If the line address is 1–895 then the default is (PHYSICAL, SUBAREA).
- If the line address is 1024–1039 or 2112–2750, then the default is (PHYSICAL, ANY).

**FRELAY=LOGICAL**

Defines the logical line group.

- Code FRELAY=LOGICAL or FRELAY=(LOGICAL, SUBAREA | SUB) to define a logical subarea line group. The PU definition statement specified by the PHYSRSC operand must be included in a line group with FRELAY=PHYSICAL.
• Code FRELAY=(LOGICAL,PERIPHERAL | PER) to define a logical peripheral line group. If a pu_name is coded on the PHYSRSC keyword (not PHYSRSC=NONE), then the PU definition statement specified by the PHYSRSC operand must be included in a line group with FRELAY=PHYSICAL.

• Code FRELAY=(LOGICAL,IP) to define a logical line group to be used for IP connections over a switched physical line. The PU definition statement specified by the PHYSRSC operand must be included in a line group with FRELAY=PHYSICAL and DIAL=YES.

**G POLL**

```
G POLL = polling_character
```

*statements: CLUSTER*
*dependencies: VTAM and NCP*

G POLL (general polling) must be coded for BSC 3270 clusters (that is, 3271 and 3275); VTAM does not support specific polling.


**GP3174**
GP3174 = NO

GP3174 = group_address

Specifies that a poll to an IBM 3174 gateway controller can be a group poll for any station on the IBM 3174, including the IBM 3174 and the stations attached to the token ring.

*group address* indicates the hexadecimal group poll address and can be any two hexadecimal digits except X'00' and X'FF'.

You can change the value of GP3174 for SDLC non-switched PUs by dynamically adding the PUs using the DR ADD definition statement.

**GWCTL**

GWCTL = SHR

GWCTL = ONLY

statements: PCCU
dependencies: SNI only

Determines whether the host’s SSCP does all the gateway control functions during LU-LU session setup. GWCTL determines the SSCP’s role for only the gateway NCP being defined, and applies only if the SSCP is on the session setup path.

**GWCTL=ONLY**

Specifies that only one SSCP controls the gateway NCP.

If the session setup path crosses three networks within a single gateway NCP, and if one or more gateway SSCPs are in session with the gateway NCP, the gateway SSCP in the center network must be designated to perform all of the gateway functions. Code GWCTL=ONLY on the PCCU statement for the gateway SSCP in the center network.

**GWCTL=SHR**

Indicates control of the gateway NCP is shared. If one gateway SSCP cannot perform the gateway control function, the other SSCP performs the function.

For an inactive NCP, the VARY ACT command processes all operands coded on the PCCU definition statement. The VARY ACQ command, however, processes only the GWCTL operand, regardless of what is coded on the PCCU definition statement.

For more information, see z/OS Communications Server: SNA Network Implementation Guide
HPR

HPR = YES
Indicates that this GROUP, LINE, or PU provides HPR support.

HPR = NO
Indicates that this GROUP, LINE, or PU does not provide HPR support.

Notes:
1. HPR defaults to NO when the HPR start option is set to NONE, (RTP,NONE), or (ANR,NONE).

Statements: GROUP, LINE, PU
Dependencies: Type 2.1; NCP V7R3 required

Specifies whether High-Performance Routing (HPR) support is enabled for a particular GROUP, LINE, or PU.

INBFRS
INBFRS—_number_of_buffers_

**statements:**
  
**HOST**

**dependencies:**

**none**

**range:**

**1–255**

Defines the number of buffers to be allocated by an NCP to receive data from a host.

**INCLUD0E**

**statements:**

**PU**

**dependencies:**

**none**

Specifies whether VTAM provides LU names to the logical units defined under this PU. The names are provided by passing a CV0E, Network Name control vector on the ACTLU requests.

**INCLUD0E=NO**

Indicates that VTAM will not provide LU names to the logical units defined under this PU.

**INCLUD0E=YES**

Indicates that VTAM will provide LU names to the logical units defined under this PU.

**INITEST**
Specifies whether VTAM loads the initial test routine, which is a diagnostic routine for a channel-attached communication controller. This routine checks the communication controller for any machine malfunctions before VTAM loads the NCP into it.

\textbf{INITEST=NO}

Specifies that VTAM will not load the initial test routine.

\textbf{INITEST=YES}

Specifies that VTAM will load the initial test routine.

\textbf{Note:}

1. Applies only to channel-attached IBM 3705 Communication Controllers.
2. If you code INITEST=YES for any communication controller other than an IBM 3705, the load fails.
3. If you code INITEST=YES for an NCP that controls a link-attached IBM 3705 Communication Controller, VTAM ignores it.

\textbf{INNPORT}
Specifies whether this physical unit supports intermediate network node (INN) traffic. This operand is not necessary for PU type 4; it is only for PU type 1 used for INN traffic.

**INNPORT=NO**

INNPORT=NO indicates that the physical unit is either not an INN port or is not a port.

**INNPORT=YES**

INNPORT=YES means that the physical unit represents a port that supports INN traffic. If you specify INNPORT=YES, this physical unit should not be deactivated when CDLINK=ACT on either the HALT or VARY INACT commands.

For frame-relay PUs, code INNPORT=YES.

This operand is created with its default value by the NCP generation process. Code INNPORT=YES only if network definition facility (NDF) changes the default value during the generation process.

Note that in certain definitions, NDF sets INNPORT to YES. When any of the following operands is specified on the GROUP definition statement of the NCP generation definition, NDF adds INNPORT=YES to all PU definition statements within that GROUP statement when the PU statements are written to the NEWDEFN file:

- `ECLTYPE=(PHYSICAL, SUBAREA)`
- `ECLTYPE=(PHYSICAL, ANY)`
- `FRELAY=PHYSICAL`
- `ETHERNET=PHYSICAL`

**ISTATUS**
Determines whether this minor node is to be initially active when the NCP major node to which it belongs is first activated, either because of a cold start of VTAM, or, after VTAM is started, by issuing the VTAM VARY ACT,COLD command for the NCP major node. For logical units, ISTATUS is also effective whenever the associated physical unit is activated.

When coding the ISTATUS operand, consider these special cases:
• For a line with TYPE=PEP, ISTATUS does not apply and is ignored if you code it.
• For a line with USE=NCP, the line is initially active.
• For a line with USE=EP, the line is initially inactive.

Code ISTATUS=INACTIVE for all start-stop lines supported by Network Terminal Option (NTO).

The operator can override the ISTATUS specification using the SCOPE operand on the VARY ACT command.

If LNCTL=CA (on GROUP), you must specify ISTATUS=INACTIVE on the LINE definition statement if PUTYPE=5 on the PU definition statement.

**ISTATUS=ACTIVE**
Indicates that the minor node is automatically activated when its major node is activated.

**ISTATUS=INACTIVE**
Indicates that the minor node is not automatically activated when its major node is activated.

**LIMRES**
Notes:
1. LIMRES=NO is the only valid value if the TRLE operand is coded.

Statements: GROUP, LINE, PU
Dependencies: VTAM; LIMQSINT; LU 6.2 only; PUTYPE=1 or 2

Specifies whether a group of lines, a line, or a physical unit is to be treated as a limited resource. A limited resource allows you to limit the use of some network connections. It also causes any sessions that traverse the resource to be deactivated if no conversation is active. A session can also be deactivated if the time limit you specify for the LIMQSINT operand on the APPL definition statement expires.

This operand applies only to LU 6.2 conversations and PU types 1 and 2.

If you do not code a value for LIMRES, resources are not considered to be limited resources. Furthermore, sessions will remain active even after all conversations have ended.

You can also use LIMRES in conjunction with the LIMINTCP start option to specify the interval to retain a free session using a limited resource for some network management flows (excluding flows using the APPN control point sessions or dependent LU server sessions). If you do not code a value for the LIMINTCP start option, resources are not considered to be limited resources, and the sessions will remain active even after all conversations have ended.

Note:
1. For NCP-defined resources, the NCP sets the limited resource indicator. If the NCP is pre-V5R4, then the limited resource indicator is not set by the NCP, even if VTAM definitions indicate that a limited resource is defined.
2. This operand is not valid when PUTYPE=4 or PUTYPE=5.
3. You should code LIMRES only for leased lines in the NCP major node. For switched lines used by the NCP major node, code LIMRES in the switched major node.

LINEAUT
LINEAUT = YES

Statements: GROUP, LINE
Dependencies: VTAM and NCP

Specifies that VTAM is to consider every line in a group as an auto-call line. If you omit this operand, but LINEADD=NONE and DIAL=YES, then LINEAUT=YES is assumed by default.


LLERP

LLERP = NOTPREF
LLERP = REQUIRED

Notes:
1 For type 2.1, the default is LLERP=REQUIRED. For SDLC, X.25, and channels links, the default is required. Otherwise, the default is LLERP=NOTPREF.

Statements: PU
Dependencies: HPR=YES

Specifies the link-level error recovery procedures (LLERP) preference for HPR connections.

LLERP=REQUIRED
   Specifies that link-level error recovery procedures are required.

LLERP=NOTPREF
   Specifies that link-level error recovery procedures are required by this PU only when the adjacent link station requires it.

LMODADR
LMODADR = (seg1_loc_mod_addr, seg2_loc_mod_addr)

statements: PU
dependencies: VTAM and NCP; LPDATS; SEGMENT
range: X'0'–X'F'

Specifies the local modem addresses used on the line going to the physical unit. LMODADR is valid on the PU definition statement only if you specify LPDATS=LPDA2 on the LINE definition statement.

Specify a local modem address (seg1_loc_mod_addr) for link segment 1 and, if necessary, a local modem address (seg2_loc_mod_addr) for link segment 2. If SEGMENT=1, the default value is 01. If SEGMENT=2, the default value is (01,02).

Note: seg2_loc_mod_addr is valid only if SEGMENT=2.

VTAM uses this operand only when processing DR ADD definition statements for PUs that support LPDA2.

LNCTL
Specifies the type of line control used by the lines in the group being defined. For more details, see the NCP, SSP, and EP Resource Definition Reference.

**LNCTL=BSC**
Indicates that the definition statements after this GROUP definition statement use the BSC line protocol.

**LNCTL=CA**
Indicates that the definition statements after this GROUP definition statement identify intermediate routing node and boundary node channel definitions.

**LNCTL=SDLC**
Indicates that the definition statements after this GROUP definition statement use the SDLC line protocol.

**LNCTL=SS**
Indicates that the definition statements after this GROUP definition statement use the SS line protocol.

**LNCTL=USER**
Indicates that either the user or NCP provides the level-5 code. The LEVEL2 and LEVEL3 operands must be specified if LNCTL=USER is coded.

**LOADFROM**
Specifies whether VTAM loads the NCP module from the host or from the communication controller external disk storage.

**LOADFROM=HOST**

    Loads the NCP from the host.

**LOADFROM=EXTERNAL**

    Loads the NCP from the 3720 or 3745 communication controller.

**Note:**

1. You cannot specify both SAVEMOD=YES and LOADFROM=EXTERNAL on the PCCU definition statement. If you specify LOADFROM=EXTERNAL, you should take the default value for SAVEMOD.

2. You cannot specify DUMPLOAD=YES, LOADFROM=HOST, and SAVEMOD=NO all together. If you specify LOADFROM=HOST and SAVEMOD=NO, you should take the default value for DUMPLOAD.

3. You cannot specify SAVEMOD=YES on the VARY ACT command and LOADFROM=EXTERNAL on this PCCU definition statement. If you do, VTAM does not use LOADFROM=EXTERNAL on the initial load.
LOADLIB

```
LOADLIB — ddname
```

**statements:** BUILD

**dependencies:** none

Specifies the data set or file in which the NCP resides.

Name the data set on a DD statement in the VTAM start procedure.

**LOADSTA**

```
LOADSTA — link_station_name
```

**statements:** PCCU

**dependencies:** none

Specifies an adjacent link station (channel- or link-attached) that is used to load the NCP. If you do not code LOADSTA on the PCCU definition statement or specify it in the VTAM VARY ACT command, VTAM chooses a default link station whenever it needs one. See z/OS Communications Server: SNA Network Implementation Guide for information about how VTAM chooses a default link station.

For a channel-attached communication controller, if the link station name is not defined (in the channel attached major node), VTAM assigns a link station name by taking the device number and adding "-S". If the device number is 3-digits, a leading zero is added to the name. For example, the device number 0C2 has the link station name of 00C2-S.

You can designate the same link station for both dumping and loading.

For more information about link stations and use of the VTAM VARY ACT command, see z/OS Communications Server: SNA Operation

**LOCADD**
LOCADD—address

Specify the locally administered address for the token-ring interface coupler (TIC). The value of LOCADD must specify an address among all the local and remote token-rings to which this TIC is connected. LOCADD is required when you specify physical resources on the GROUP definition statement.

Use the following format when coding LOCADD:

LOCADD=cccccccccccc

where:

cccccccccccc

is the last 6 bytes of the terminal’s ring-station address (the first digit must be in the range 4–7). The address must consist of 12 hexadecimal digits that range from X'4000 0000 0000' to X'7FFF FFFF FFFF' using the following rules.

- If bytes 0 and 1 are equal to either X'4000' or X'7FFF' then bytes 2 through 5 must be unique on the network.
- If bytes 0 and 1 are in the address range X'4001' through X'7FFE' then bytes 0 through 5 must be unique on the network.

A locally administered medium access control (MAC) address coded for LOCADD cannot have the same value as that specified on the ADDR operand on the PU definition statement. See NCP, SSP, and EP Resource Definition Reference for additional information.

LOCADDR
LOCADDR

Specifies the logical unit's local address at the associated physical unit. Specify the addresses in ascending order. LOCADDR=0 specifies an independent LU.

An LU definition statement is not required for every possible local address, and LOCADDR values need not be consecutive. Unused local addresses smaller than the largest local address at a station are not assigned network resources.

Note:
1. The value of LOCADDR depends on the requirements of the device being defined. Consult the appropriate component description manual for these restrictions; for example, logical units associated with a 3279 or 3276 control unit must have a LOCADDR in the range 2–33.
2. Although you can use this operand to define independent LUs, it is recommended that you either use dynamic definition or predefine the independent LUs as CDRSCs.
3. VTAM cannot recognize a dynamically changed LOCADDR on the first activation of an NCP, even when VARY ACT,UPDATE=ALL is used. (Warm starts are considered initial activations.) However, on changes to LOCADDR after the first activation, VTAM can compare the value in the definition statement to the value previously coded and recognize the change.
LOGAPPL controlling_primary_lu

statements: GROUP, LINE, PU, LU, CLUSTER, TERMINAL
dependencies: VTAM only; NetView or TSO

Identifies the name of the primary logical unit that this logical unit will automatically log on to as a secondary logical unit when this logical unit is activated. The name you specify can also be a USERVAR that will be resolved into its current value before being used to establish or redrive the session with the controlling primary logical unit.

For independent logical units, the LOGAPPL operand establishes a session with the primary logical unit designated in the LOGAPPL operand if a session does not currently exist with this logical unit. The session is established regardless of whether sessions with other logical units exist. The LOGAPPL operand cannot contain the name of an independent logical unit because independent logical units cannot be the controlling logical units for an automatic logon session.

Automatic logon LU-LU sessions are not redriven after CP-CP outages even though the CP-CP sessions might redrive.

You can specify a network-qualified name for LOGAPPL. You cannot specify an LUALIAS name for LOGAPPL.

Note: If you specify the NetView or TSO program as the application program on the LOGAPPL operand, you will get unpredictable results in recovery situations.

For more information on establishing sessions, see the z/OS Communications Server: SNA Network Implementation Guide.
LOGTAB - \_interpret_table_name\_

**Statements:** GROUP, LINE, PU, LU, CLUSTER, TERMINAL

**Dependencies:** VTAM only; dependent LUs only

Specifies the name of an interpret table that VTAM uses to associate a logon request with the name of an application program or routine to which this logon is sent. VTAM uses the interpret table to interpret the name received from an initiation request. For more information on using interpret tables in conjunction with USS tables, see "Unformatted system services tables" on page 1069.

LOGTAB is valid for dependent logical units only.

**LPDA**

**Statements:** PU

**Dependencies:** VTAM and NCP

Specifies whether Link Problem Determination Aid (LPDA) tests can be run for the physical unit.

**Note:** Do not specify this keyword if PUTYPE=4 has been specified.

**LPDA=ALLOW**

Indicates LPDA test will run for the PU.

**LPDA=BLOCK**

Indicates no LPDA test will run for the PU.

**LUDR**
Specifies whether an LU is eligible for dynamic reconfiguration.

This operand is ignored for independent LUs.

If you use sifting for LUDR and use VARY ACT,UPDATE=ALL to move an LU, make sure the sifted value is the same before and after any moves. If LUDR gets sifted differently as the result of a move and its value is no longer the same, VTAM and NCP will have mismatching LUDR values on the next initial activation of the NCP.

**LUDR=NO**

Indicates that VTAM rejects any attempt to move or delete the LU.

LUDR=NO is not valid if PUDR=YES. However, by using DRDS definitions you can create an LU with LUDR=NO under a PU with PUDR=YES, and DR operations on the LU will fail, but DR operations on the PU ignore the LUDR=NO value.

**LUDR=YES**

Indicates that the LU can be dynamically deleted or moved. Valid only if dynamic reconfiguration is supported for this LU.

**LUGROUP**
LUGROUP

statements: PU
dependencies: VTAM only

Specifies the name of the model LU group that VTAM uses to select a model LU definition when dynamically defining an LU. This operand also indicates that the device supports dynamic definition of LUs.

LUSED

statements: PU
dependencies: VTAM only; LUGROUP

Provides a 1–8 character pattern name that the SDDLU exit routine uses to generate an LU name when dynamically defining an LU. LUSED is valid only when you code the LUGROUP operand.

If you use the IBM-supplied SDDLU exit routine, this operand is required and must have the following format:

• The first character must be either alphabetical or one of the national characters (@ or $).
• The remaining characters can be either alphabetical, numerical, or national.
• There must be either two or three contiguous national # characters in the name. These characters represent holders, in which the logical unit’s local address is substituted when an LU’s name is generated. If you specify three # characters, the LU’s logical address is substituted in decimal for the # characters. If you specify two # characters, the LU’s logical address is substituted in hexadecimal for the # characters.

When used with a user-written SDDLU exit routine, this operand is optional and there are no restrictions on its format. For more information on the SDDLU exit, see [Z/OS Communications Server: SNA Customization](#).

MAXBFRU
MAXBFRU

Specifies the number of buffers to be allocated by VTAM to receive data from an NCP. The value coded for MAXBFRU must satisfy the condition that MAXBFRU times the IOBUF size is less than 65535.

If you define LNCTL=NCP in a channel-attachment major node that is activated before the NCP major node, ensure that the MAXBFRU value in the NCP major node matches the MAXBFRU value in the channel-attachment major node. Because the MAXBFRU value in the channel-attachment major node is processed first, an error message is issued if the MAXBFRU value in the NCP major node differs from the value in the channel-attachment major node.

Note: If you are defining the channel to the NCP in a channel-attachment major node, VTAM might ignore this operand when the NCP is activated by a CMC host. See z/OS Communications Server: SNA Network Implementation Guide for more information about CMC configurations.

MAXDATA

Specifies the maximum number of bytes that the NCP can receive in one PIU or PIU segment, including the transmission header (TH) and the request/response header (RH).

To optimize the processing of a MODIFY DUMP, OPTION=DYNAMIC command, MAXDATA should be at least 2082 (2048 plus 34, the size of the PIU header).

Note: If you plan to define the channel to the NCP in a channel-attachment major node, this operand is replaced by the MAXDATA operand on the PU definition statement for the channel-attachment major node. See z/OS Communications Server: SNA Network Implementation Guide for a description of the MAXDATA operand in a channel-attachment major node.
MAXOUT = number_of_pius

Specifies the maximum number of unacknowledged information frames that NCP can have outstanding for a PU before requesting a response from the PU.

For frame-relay PUs, the default is 8.

MAXSESS

MAXSESS = max_number_LU-LU_sessions

Specifies the maximum number of concurrent LU-LU sessions in which an independent LU can participate. The MAXSESS value applies to each connection the LU is using to establish LU-LU sessions. For example, if MAXSESS=2 was specified for LUA and LUA appears on three PUs, LUA can have two sessions on each of the three PUs for a total of six sessions.

If MAXSESS is not coded, the value specified on the BUILD definition statement will be used. See NCP, SSP, and EP Resource Definition Reference for additional information about this operand.

MAXSUBA
**MAXSUBA**

```
MAXSUBA = n
```

**statements:** BUILD, NETWORK  
**dependencies:** ENA  
**range:** 1–255

Specifies the highest subarea value that can be assigned within the network for communicating with a subarea node that does not support extended network addressing (ENA).

This operand is required if any host or NCP in the network does not support ENA. All nodes must specify the same value.

For more information, see “MAXSUBA start option” on page 894. See z/OS Communications Server: SNA Network Implementation Guide for considerations when coding this operand.

---

**MDLENT**

```
MDLENT = model_name_table_entry_name
```

**statements:** GROUP, LINE, PU, LU, CLUSTER, TERMINAL  
**dependencies:** VTAM only; MDLTAB

Specifies the name of the model name table entry to be used for a logical unit.

If you code the MDLENT operand, you must also code the MDLTAB operand. If you omit MDLENT and include MDLTAB, VTAM uses the first entry in the table named in MDLTAB.

---

**MDLTAB**

---
MDLTAB

statements: GROUP, LINE, PU, LU, CLUSTER, TERMINAL
dependencies: VTAM only

Specifies the name of the model name table to be used for a logical unit. See "Model name table" on page 1054 for a description of how to define a model name table.

The model name is passed only when VTAM initiates the session with the application program, for example, only when the LOGAPPL = parameter is also coded.

If you omit the LOGAPPL operand or the MDLTAB operand, VTAM will not provide the name of the model name table to the application during LU-LU session initiation. However, you can provide the model name in other ways. You can enter it from the terminal on the MODEL operand of the LOGON command, or VTAM can send it to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the MODEL operand of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace a model name table. See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

MDUMPDS
Specifies one of three operands (the others are DUMPDS and CDUMPDS) that name dump files for this communication controller. MDUMPDS applies only to 3720, 3725, and 3745 communication controllers.

This operand is optional. If it is coded, it names the file that contains maintenance and operator subsystem (MOSS) dumps. If it is not coded, the MOSS dumps are contained in the file named on the DUMPDS operand. For more information on DUMPDS, see “DUMPDS” on page 475.

*moss_dump_name* must be on a DD statement, which defines the file that is to contain the data from a storage dump of a communication controller.

**MODETAB**

Specifies the name of a logon mode table to be used to correlate each logon mode name with a set of session parameters for the logical unit. The name you code must be the name of a logon mode table created as described in “Logon mode table” on page 1026. If you do not supply a logon mode table for the logical unit on the MODETAB operand, an IBM-supplied default logon mode table (ISTINCLM) is used. If you specify a table, both the table you specify and the default table are used.

*name*
Assigns a name to the resource being defined by the definition statement.

**On GROUP, LINE, PU, LU, CLUSTER, TERMINAL:**
Provides the required network name for the resource being defined by the definition statement.

**On PCCU:**
Provides an optional name for this major node. Although name is optional, it is recommended that you code one because the node can then be identified by name when messages are issued regarding the node's status.

**On BUILD:**
Provides an optional name for the definition statement.

**NATIVE**

NATIVE = NO

NATIVE = YES

Specifies whether this link station represents a connection to a native node.

If NATIVE is not specified on the ADJCP definition statement or on the PU definition statement, the two nodes negotiate their subnetwork affiliation during connection establishment.

- If the NETIDs match, the connection defaults to a native connection.
- If the NETIDs are different, the connection defaults to a nonnative connection.

If you code NATIVE on both the ADJCP and PU definition statement, the values must match.

For transmission groups (TGs) between nodes, the value specified or negotiated for NATIVE must match. If a TG is already active, subsequent TG activations must not have a specified or negotiated value for NATIVE that conflicts with the value of NATIVE for the active TG. If there is a conflict, the TG attempting activation is not allowed to become active. If the activation is for a virtual-route-based transmission group then the SSCP-SSCP session will also fail.

**NETID**
NETID=network_id

Notes:
1 NOXNETLS applies only to the PU definition statement.
Specifies 1–8 character network identifier.

**Note:** The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

**NETID=(network_id)**

- **On PCCU:** (network_id) specifies the 1–8 character name of the network of the host represented by this PCCU definition statement. VTAM uses this name along with the SUBAREA operand to determine which PCCU definition statement it should process in the NCP major node.
- **On BUILD:** (network_id) specifies the network name of the host coded on the NETID start option. This operand is required for NCP V4R3 and later releases.
- **On HOST:** (network_id) specifies a 1–8 character name that identifies the network containing the corresponding host node. It is used along with the SUBAREA operand to determine which HOST definition statement the VTAM host should process. This operand is optional.
- **On NETWORK:** (network_id) specifies the 1–8 character network identifier of the nonnative network in which the gateway SSCP resides. If the value for NETID matches the value of VTAM’s NETID start option, VTAM uses the MAXSUBA value on this statement as the maximum subarea value, and the SUBAREA value is the actual subarea for the gateway NCP representation in the VTAM network.
- **On PU:** (network_id) specifies a 1–8 character network identifier. For type 4 and 5 physical units, network_id identifies the network of the subarea serviced by the line. NETID is used only in gateway NCPs, and is required for a connection to another network.

**NETID=(NOXNETLS)**

- **PU only:** (NOXNETLS) specifies that VTAM does not allow nonnative network attachment for LU-LU sessions for this PU. When NETID=(NOXNETLS), VTAM indicates to the NCP that this PU uses VTAM’s native network ID.

LU-LU sessions might fail if the NETID sent in the BIND from the LU is different than the NETID specified in VTAM (even though PU-PU connections can be established). If no NETID is included in the BIND, or if the NETID is identical to that specified by VTAM, the LU-LU session should activate.

**Note:** For switched subarea connections, all lines within the same group must be defined with the same network_id.

For type 2 physical units, NETID is used in conjunction with the XNETALS start option in support of nonnative network connection. See "XNETALS start option" on page 978 for information about XNETALS.
If you specify NETID on the PU definition statement, VTAM ensures that when the physical unit is active, the connecting resource is within the network specified by NETID. If you omit NETID, VTAM dynamically processes the network ID when a connection is established. If you code NETID, and nonnative network connection is not supported, the definition is rejected unless the operand specifies VTAM’s network ID.

If dynamic dial-out connections are required (that is, a session request drives the dial), NETID must be predefined. Otherwise, VTAM might not be able to determine the connecting network ID and the resultant sessions might fail. If leased connections are not established, and if NETID is not predefined, directory requests could fail because of VTAM’s inability to determine the resource’s actual network.

**NEWNAME**

```plaintext
NEWNAME = ncp_load_module
```

*statements:*

BUILD

*dependencies:*

none

Specifies 1–7 characters for the NCP load module name. It must be a unique NCP name across the network. This name must match the member name for the NCP source module in both VTAMLST and the LOADLIB specified.

If you do not code PUNAME, NEWNAME also represents the resource name for the NCP physical unit (PU Type 4).

If you code NEWNAME, and LOADMOD is specified on the VARY ACT command, they must match. See *z/OS Communications Server: SNA Operation* for information about LOADMOD.

**NEWPATH**
Specifies the dynamic path update member names in the VTAM definition library. When the NCP is successfully activated, VTAM processes and sends path table update specifications to the NCP.

Up to three member names can be specified; if more than three member names are specified on the PCCU definition statement, only the first three are used.

VTAM sends the path table update specifications over the SSCP-PU session before any links, other than the link for the SSCP-PU session, are activated. If you attempt to change or delete the explicit route used for the SSCP-PU session, the NCP rejects the request (because the explicit route is currently operative), and VTAM displays a warning message.

Any NEWPATH operand specification on the PCCU definition statement is nullified if the NEWPATH operand is specified in the VARY ACT command for the NCP. See [z/OS Communications Server: SNA Operation] for more information about the VARY ACT command.

When a dynamic path update member is processed as a result of a NEWPATH operand on the PCCU definition statement, path update specification sets for other subareas, if any, are not applied.
**NN**

Specifies whether the adjacent node is expected to be a network node. If a value for the NN operand is specified, it will be validated when a connection is attempted to the adjacent node. If the adjacent CP is not the type of node that is expected, then connection setup fails. If the NN operand is not specified, then the APPN capabilities of the adjacent node are identified and accepted when a connection is established.

**NN=NO**

Specifies that the adjacent node is expected to be an end node.

**NN=YES**

Specifies that the adjacent node is expected to be a network node.

**Restriction:** In a peripheral subnetwork boundary configuration between a border node and a network node without border node function, the border node appears to the non-border node as an end node. In this case, coding NN=YES on the definition statement that represents the border node, a connection failure will occur. Avoid the failure by not coding the NN operand or by coding NN=NO.

**OWNER**
statements: PCCU, GROUP, LINE, PU
dependencies: VTAM only; on PU, PUTYPE=1 or 2

**PCCU:**
Defines an identifier for the VTAM defined by the PCCU statement. The identifier is used to associate a host VTAM with the resources it controls. You can use this operand to divide the resources among the hosts that are in session with an NCP.

If you code OWNER, the activating host owns the resources that do not have the OWNER operand specified or have OWNER coded with a value that matches owner name. If you do not specify OWNER, all resources are owned by the activating host, whether the OWNER operand is specified on the resources.

**GROUP, LINE, PU:**
Associates the resources in a communication controller with the VTAM that controls these resources. OWNER associates those resources with the PCCU definition statement that has the same OWNER value. A resource can be associated with several VTAMs. If you do not specify the OWNER operand, the resource can be owned by any host.

Note that for the PU definition statement, OWNER is valid only when PUTYPE=1 or 2.

**Note:**
1. The OWNER operand is valid on the LINE or PU definition statement for non-switched lines. It is also valid on the LINE definition statement for switched lines.
2. If you do not specify the OWNER operand on the PCCU definition statement, the OWNER operand coded on a GROUP, LINE or PU definition statement is ignored.
3. If the OWNER operand on a GROUP, LINE or PU definition statement does not match the OWNER operand on the PCCU definition statement:
   - If you specify BACKUP=YES on the PCCU definition statement, the resource cannot be used until a VARY ACQ command is issued for the NCP.
   - If you specify BACKUP=NO or use it by default on the PCCU definition statement:
     - The OWNER operand on the GROUP definition statement will be processed, but the corresponding line and physical unit cannot be used by this host.
     - A PU, and consequently, its LUs, with a different OWNER value than its higher-level resources cannot be activated.
4. The GROUP and LINE definition statements can contain more than one owner name, but the PU definition statement can specify only one OWNER value.
5. All PUs in the FRSESET should be defined with the same OWNER.
6. If a higher-level definition statement is not associated with this host, coding the OWNER operand on a lower-level definition statement does not establish ownership for this host.
PACING

PACING = (1,1, ADAP)

PACING = max (max), ADAP

PACING = min (min), FIXED, ADAP

PACING = (max, min, type)

PACING = (max, type)

PACING = (min, type)

statements: GROUP, LINE, PU, LU, CLUSTER

dependencies: none

range: 0–63

Specifies how pacing is handled between a logical unit and the NCP to which it is connected.

The formats for PACING values are:

- PACING=(max, min, type)
- PACING=max
- PACING=(max, min)
- PACING=(max, type)
- PACING=(min, type)

max

Specifies the maximum window size. max is the maximum number of normal-flow requests that VTAM sends for a given LU-LU session before waiting for a pacing response. No further normal-flow requests can be sent to the logical unit until it is ready to receive more requests.

max can be expressed with leading zeros and is limited to eight digits.

min

Specifies the minimum window size.

min can be expressed with leading zeros and is limited to eight digits.

type

Specifies the pacing type: FIXED or ADAPTIVE (ADAP).

type is used only by NCP, and it requires at least NCP V5R4.

Defaults for PACING follow:

- If none of the values are specified, the defaults are PACING=(1,1,ADAP).
- If only max is specified, its value is also used for min. type defaults to ADAP.
- If max and min are specified, and max equals 0, then the defaults are PACING=(1,1,ADAP).
If \( \text{max} \) and \( \text{min} \) are specified, and \( \text{max} \) does not equal 0, \( \text{max} \) must be greater than or equal to \( \text{min} \) for those values to be used. \text{type} \) is set to ADAP. If \( \text{max} \) is less than \( \text{min} \), the values are set to PACING=(1,1,ADAP).

If \( \text{max} \) and \( \text{type} \) are specified, \( \text{min} \) takes the value of \( \text{max} \).

If only \( \text{type} \) is specified, \( \text{max} \) and \( \text{min} \) are set to 1.

For more information about pacing, see [z/OS Communications Server: SNA Network Implementation Guide](#).

### PDELAY

<table>
<thead>
<tr>
<th>PDELAY</th>
<th>TERRESTR</th>
<th>NEGLIGIB</th>
</tr>
</thead>
</table>

Notes:

1. NEGLIGIB is the default if the TRLE operand is coded.

*statements:* GROUP, LINE, PU  
*dependencies:* APPN only

Specifies the maximum propagation delay of the link for the transmission group. Propagation delay represents the time needed for a signal to travel from one end of the link to the other.

**PDELAY=NEGLIGIB**  
Indicates a local area network delay (less than 0.48 milliseconds).

**PDELAY=TERRESTR**  
Indicates telephone network delay (between 0.48 and 49.152 milliseconds).

**PDELAY=PACKET**  
Indicates a packet-switched network delay (between 49.152 and 245.76 milliseconds).

**PDELAY=LONG**  
Indicates a satellite delay (greater than 245.76 milliseconds).

### PHYSRSC
Indicates the lines in the group are logical resources having dependency on a higher level (physical) resource, where *puname* specifies the label of a PU statement for the physical resource with which the logical lines are associated. A *puname* of NONE, or NONE as the first item in a list, indicates no hierarchical dependency between the physical and the logical resources.

**Note:** The PHYSRSC operand is processed only when a VTAM activates an NCP and becomes the owner of the NCP’s resources. In a configuration where either of two VTAMs can activate the same NCP’s resources (twin-tail configuration), there will be a discrepancy between a display of resources after the NCP’s resources are deactivated by the owning VTAM and later activated by the other VTAM.

For example, if the owning VTAM deactivates a frame relay physical line, and the other VTAM then activates the line, a display of a PU on the physical line shows the PU but does not show the logical lines which were displayed before the physical line was deactivated. The logical lines remained active on the owning VTAM. The PU is now active under the other VTAM but does not have any logical lines associated with it. To reassign the logical lines to the PU on the other VTAM, you would have to deactivate the NCP and reactivate it under the other VTAM, which would then be the owning VTAM.

**POLIMIT**
POLIMIT = (1, NOWAIT)

Must be coded for a polled non-switched line. It determines the action taken by the NCP if the number \( n \) of negative responses is exceeded when polling.

Because the QUEUE parameter is required by NCP, you must code POLIMIT=(,QUEUE) or POLIMIT=(n,QUEUE).


**PU**

Specifies whether a BSC 3270 is treated as a physical unit and whether all terminals subordinate to the cluster are treated as logical units.

**PU=NO**

If PU=NO, it is assumed that the associated terminals are supported by a line control program such as the IBM licensed program Network Terminal Option (NTO) or a similar user-written program.

**PU=YES**

If PU=YES, the device type must be a 3270 type.

You cannot code this operand on the CLUSTER definition statement. You must code it on the GROUP or LINE definition statement, and let it sift down to the CLUSTER level.
PUDR = NO
YES

Specifies whether a PU is eligible for dynamic reconfiguration.

If you use sifting for PUDR and use VARY ACT,UPDATE=ALL to move a PU, make sure the sifted value is the same before and after any moves. If PUDR gets sifted differently as the result of a move and its value is no longer the same, VTAM and NCP will have mismatching PUDR values on the next initial activation of the NCP.

PUNAME

Specifies the physical unit name that is to be associated with the NCP after activation is complete. The value coded here must match the name coded on the ID operand for the VARY ACT command issued to activate the NCP, and it must be a unique NCP name across the network. After the NCP is active, this physical unit name is the name used to identify the NCP (not the load module name specified on NEWNAME).

If you do not code PUNAME, the value of the NEWNAME operand will be used the physical unit name.

PUTYPE
PUTYPE = \text{5} \\
\text{PUTYPE = 1, 2, 4}

\text{PUTYPE = 5}

\text{statements: GROUP, LINE, PU}
\text{dependencies: VTAM and NCP}

Specifies the physical unit type represented by the PU definition statement.

\text{Note:}
1. If you code LNCTL=CA on the GROUP definition statement, 1, 2, and 5 are the valid values for PUTYPE, and PUTYPE=5 is the default. The physical unit with PUTYPE=1 is valid only for those groups created specifically for physical serial optical channel (SOC) lines. The physical unit with PUTYPE=2 is a channel adapter used to communicate with a type 2.1 node. The physical unit with PUTYPE=5 is a host processor or VTAM node.
2. If you code PUTYPE=4, DIAL=YES, and LNCTL=SDLC on the GROUP definition statement, the group is used for switched subarea connections.
3. For a type 2.1 node, code PUTYPE=2 and XID=YES.
4. For a non-switched frame relay PU (FRELAY=PHYSICAL), code PUTYPE=1 and XID=NO.
5. If you code PUTYPE=5, and if LNCTL=CA, you must specify ISTATUS=INACTIVE on the LINE definition statement. If you specify ISTATUS=ACTIVE, the activation fails.

For more details, see the \textit{NCP, SSP, and EP Resource Definition Reference}

\textbf{REGISTER}

\textbf{For independent LUs:}

\textbf{For dependent LUs:}
Specifies how a resource should be registered. Dependent or independent LUs.

**REGISTER=CDSERVR**
Indicates that an end node resource is registered to a network node server and directory resource registration is requested for it. A network node resource is registered at the central directory server. If the node is configured as a central directory server, this value has the same effect as NO.

**REGISTER=NETSRVR**
Indicates that the end node resource should be registered to its network node server, but that directory registration should not be requested for it.

For dependent LUs, NETSRVR is the default, and LOCADDR must be specified as a nonzero value.

Dependent LUs are not usually the targets of LU-LU session setup attempts and are therefore not likely to be the targets of an APPN LOCATE search. However, because end nodes cannot be searched, you can register their dependent LUs with their network node server.

**REGISTER=NO**
Indicates that the resource should not be registered.

For independent LUs, NO is the default, and LOCADDR must be specified as 0.

**RESETLU**
RESETLU = YES
Specifies that during cleanup processing for dependent LUs, VTAM will issue an ACTLU request after the DACTLU request.

RESETLU = NO
Specifies that during clean-up processing for LU-LU sessions, RESETLU=NO prevents a DACTLU-ACTLU from flowing to an LU.

RESSCB
Specifies the number of session control blocks reserved by the NCP for the logical unit. This operand is valid for independent logical units only. If you code it for other logical units, VTAM ignores it.

For information about NCP restrictions on this operand, see NCP, SSP, and EP Resource Definition Reference.

RETRIES
RETRIES = ( , pause_interval , maximum_retries , )

statements: GROUP, LINE, PU
dependencies: VTAM and NCP
ranges: 0–255
0–127

Specifies the number of attempts to be made to recover from errors occurring
during transmission over the link. Each attempt at recovery, or retry, is a
retransmission of data or control commands. Retries are performed in sets, which
are called retry sequences.

pause_interval
  Specifies the pause interval in seconds between two retry sequences. Valid
  values are 0–255. When the number of retries in a retry sequence is 0,
pause_interval must be 0.
  
  If maximum_retries is specified and valid and pause_interval is not specified, the
  NCP default of 0 is used for pause_interval.

maximum_retries
  Specifies the maximum number of retry sequences. Valid values are 0–127,
  except for LNCTL=BSC or LNCTL=SS where valid values are 0–255.
  
  If pause_interval is specified and valid and maximum_retries is not specified, the
  NCP default of 1 is used for maximum_retries.

If neither pause_interval nor maximum_retries is specified, the RETRIES operand is
ignored. If a value that is not valid is specified for pause_interval or
maximum_retries, an error message is given and the operand is ignored.

You can change the time interval and number of retry sequences of RETRIES for
SDLC non-switched PUs by dynamically adding the PUs using the DR ADD
definition statement.

RNAME
RNAME = \texttt{link\_station\_name} \quad (\texttt{-link\_station\_name-})

\begin{verbatim}
statements: PCCU
dependencies: none
\end{verbatim}

Must match the label on the PU definition statement that describes this NCP in an adjacent communication controller or host. (That PU definition statement identifies, to the other NCP or host, the link that connects it to this communication controller.) You can specify up to 13 SDLC or channel link station names; each one represents a contact point that a host can use to reach this communication controller. As you specify values for RNAME, keep in mind that VTAM reads multiple RNAMEs from right to left.

If you do not code RNAME when defining a link-attached communication controller, the VTAM operator should supply it as an operand on the VARY ACT command.

When the VTAM operator issues a VARY ACT command to activate this communication controller, if no link-station or channel-device address contact points are supplied (either with the RNAME and CUADDR operands or the VARY ACT command), a link station adjacent to the communication controller must already be active.

For information about how to code RNAME when the NCP is defined in a channel-attachment major node, see \texttt{z/OS Communications Server: SNA Network Implementation Guide}.

\textbf{SALIMIT}
SALIMIT

Specifies the largest subarea address supported by this NCP in the native network.

If you code a value other than one of these, it is rounded up to the next valid value. If you code a value above 65535, 65535 is used.

SAVEMOD

Indicates whether you want a copy of the load module saved on the 3720 or 3745 communication controller external disk.

**SAVEMOD=NO**

If you code NO, the NCP is not saved.

**SAVEMOD=YES**

If you code YES, the NCP load module is saved after it is loaded from the host.

**Note:**

1. This operand is used only if there is no SAVEMOD operand specified on the VARY ACT command, and if a load is actually performed.
2. You should not specify both DUMPLD=YES and SAVEMOD=NO on this PCCU definition statement. If you do, DUMPLD is forced to NO unless you specify LOADFROM=EXT on the VARY ACT command.

For more information about the VARY ACT command, see "z/OS Communications Server: SNA Operation".

SECNET
Specifies whether a physical unit attached to a non-switched line is associated with a secondary network containing resources whose connectivity is not defined to VTAM.

This operand applies only if you have IBM 3710 Network Controllers or IBM 3174 token-ring attachments in your network.

The default for this operand is the value specified for SECNET on the LINE definition statement, if coded. However, if you do not code SECNET on the LINE definition statement, its default value on the PU definition statement is NO.

**SECNET=NO**

Indicates that data will not require special problem determination considerations.

**SECNET=YES**

Specifies data that is received by VTAM from this physical unit is flagged as requiring special problem determination considerations when routed to a communication network management (CNM) application.

**SECURITY**
Specifies the security level of the transmission group. The following options range from the least secure level to the most secure level.

**SECURITY=UNSECURE**
- Specifies no security level.

**SECURITY=PUBLIC**
- Specifies a public switched network.

**SECURITY=UNDERGRO**
- Specifies an underground cable, not guarded.

**SECURITY=SECURE**
- Specifies a secure conduit, not guarded.

**SECURITY=GUARDED**
- Specifies a guarded conduit, physical only.

**SECURITY=ENCRYPT**
- Specifies link encryption.

**SECURITY=SHIELDED**
- Specifies a guarded conduit, physical and radiation shielded.

**SEGMENT**
SEGMENT statements: PU
dependencies: VTAM and NCP

Specifies which link segment a station is on and allows the NCP to set up Link Problem Determination Aid-2 (LPDA2) tests accordingly. SEGMENT is valid only if you specify LPDATS=LPDA2 on the LINE definition statement.

SEGMENT=1
Indicates the station is on segment number 1. This is the default value if TAILING=NO is specified.

SEGMENT=2
Indicates the station is on segment number 2. This is the default value if TAILING=YES is specified.

For more information on the TAILING operand, see the NCP, SSP, and EP Resource Definition Reference.

Note: VTAM uses this operand only when processing DR ADD definition statements for PUs that support LPDA-2.

Reference the NCP, SSP, and EP Resource Definition Reference for proper coding of the SEGMENT operand.

SHM
Specifies whether short-hold mode is supported. Only lines representing switched virtual circuits (SVCs) controlled by X.25 NCP Packet Switching Interface (NPSI) are capable of short-hold mode.

**SHM=NO**
Indicates that short-hold mode is not supported.

**SHM=YES**
Indicates that short-hold mode is supported and any short-hold mode operands specified on the PATH definition statement in the switched major node should be used for the connection. If you specify SHM=YES, code DIAL=YES and PUTYPE=4 on the GROUP definition statement.

**SSCPFM**

Determines whether a logical unit or terminal can support character-coded messages in its communication with the SSCP.

For BSC devices, the only valid values are USS3275 and USS3270. For non-SNA devices supported with Network Terminal Option (NTO), the only valid values are USS3780 and USSNTO.

For BSC devices, the default is USS3275. For all other devices, the default is SSCPFM=FSS.

The SSCPFM operand has no effect on BSC terminals that are not printers.

**SSCPFM=FSS**
Specifies that formatted messages are supported by this logical unit or terminal for SSCP communication. Consult the individual terminal component description manual to determine whether this value or SSCPFM=USSSCS should be coded.
Note: You might receive character-coded messages when SSCP FM=F SS.

SSCP FM=US SNTO
Coded for all non-SNA devices supported with NTO for which USS3780 does not apply. These devices use the USS command facilities and are supported by all releases of NTO. See NTO Planning, Migration, and Resource Definition for more information.

SSCP FM=US SSSCS
Specifies that character-coded messages are supported by this logical unit or terminal for SSCP communication. Consult the individual terminal component description manual to determine whether this value or SSCP FM=F S should be coded.

SSCP FM=US S3270
Allows printing of a user-defined logon message (USS MSG10) on BSC printers. Code SSCP FM=US S3270 for terminals attached to a controller defined as an SDLC 3271 (PUTYPE=1) or SDLC 3275 (PUTYPE=1).

SSCP FM=US S3275
Prevents printing of a user-defined logon message (USS MSG10) for a terminal defined as an SDLC 3275 (PU type 1) that might have an attached printer. Code SSCP FM=US S3275 to prevent user-defined logon messages from printing on BSC printers.

Note: Coding SSCP FM=US S3275 prevents the local copy hardware function key from working with certain BSC printers. To make sure the local copy function works, code SSCP FM=US S3270.

SSCP FM=US S3780
Coded for non-SNA devices, supported with the network terminal option (NTO), that use the 3780 protocol. These devices use USS command facilities (LOGON and IBM TEST) and are supported only by Release 2 of NTO.

SUBAREA
On PCCU:
Specifies the subarea number of the host to which this PCCU definition statement applies. suba_num is a decimal integer in the range 1–65535.

The subarea value should match the value assigned to the HOSTSA start option when VTAM is started. VTAM uses the subarea value to determine which PCCU definition statement it should process in the NCP major node. See the Communications Server: SNA Network Implementation Guide for a discussion about how VTAM selects a particular PCCU definition statement.

On BUILD:
Specifies the native network subarea address that represents the gateway NCP to the network being defined. This subarea address is unique only in the network being defined by this BUILD definition statement.

This operand is required.

On NETWORK:
Specifies the subarea address that represents the gateway NCP to the network being defined. This subarea is unique only in the network being defined by this NETWORK definition statement.

On HOST:
Specifies the subarea address of the access method or access methods that the NCP can communicate with over a channel.

Each subarea in a network (whether the network is controlled by one or multiple NCPs) must have a unique subarea address. In SNA interconnected networks, this is true for each network that the gateway NCP joins.

**SUBPORTS**
Specifies the names of the primary type 1 PUs and their optional backups. There are a minimum of two and a maximum of four PUs in SUBPORTS.

### SWNORDER

Specifies the way VTAM locates a switched PU. If you do not code SWNORDER, the default is the value of the SWNORDER start option. If VTAM does not find the PU using the SWNORDER value you specify, and the second parameter of SWNORDER is FIRST, it searches by the other value. For example, if you code SWNORDER=CPNAME and VTAM cannot find a PU with a matching CPNAME, it searches by the station identifier (IDBLK and IDNUM operands on the PU definition statement for the switched major nodes).

If VTAM does not find the PU using the SWNORDER value you specify and the second parameter of SWNORDER is ONLY, the search for the PU terminates. For example, if you code SWNORDER=(CPNAME,ONLY) and VTAM cannot find a PU with a matching CPNAME, no attempt will be made to find the PU using the station identifier. If dynamic PUs are allowed, a dynamic PU will be created and PU connection processing will continue. If dynamic PUs are not allowed, the PU connection process will fail.

These operands override the SWNORDER start option values for SWNORDER.

**SWNORDER=(value_1,value_2) :**

- **value_1=CPNAME**
  - Specifies that VTAM searches for a switched PU by the CPNAME.
**value_1=STATNID**
Specifies that VTAM searches for a switched PU by the station identifier.

**value_2=FIRST**
If a switched PU is not found using the value specified by `value_1`, the alternate value of `value_1` will be used to search for a switched PU.

**value_2=ONLY**
If a switched PU is not found using the value specified by `value_1`, a search using the alternate value of `value_1` will not be performed.

If either `value_1` or `value_2` are omitted from the SWNORDER definition, the corresponding start option SWNORDER value will be the default for the omitted value.

**TERM**

---

**TERM = type**

**statements:** PU, LU  
**statements:** GROUP, LINE, CLUSTER, TERMINAL

**dependency:** VTAM and NCP  
**dependency:** VTAM and NCP

Describes, for a virtual logical unit supported through Network Terminal Option (NTO), the device data stream compatible characteristics. These characteristics are placed in an 8-byte program-supplied storage area, after an INQUIRE DEVCHAR macroinstruction is executed by an application program that specifies the virtual logical unit name. For more information about the DEVCHAR DSECT, see [z/OS Communications Server: SNA Programming](https://www.ibm.com/support/knowledgecenter/S570T_2.3.0/com.ibm.mif.ref.doc/gf002649.html).

For specific considerations for BSC terminals, see [NCP, SSP, and EP Resource Definition Reference](https://www.ibm.com/support/knowledgecenter/S570T_2.3.0/com.ibm.mif.ref.doc/gm201166.html).

See [NTO Planning, Migration, and Resource Definition](https://www.ibm.com/support/knowledgecenter/S570T_2.3.0/com.ibm.mif.ref.doc/gf002649.html) for more information.

**TGN**

For subarea TGN:  
For APPN TGN:

---
Subarea:
Assigns the transmission group (TG) number that represents the connection between two nodes. A connection can be:

- The channel between host and communication controller, where PUTYPE=5 is coded
- The channel between two communication controllers, where PUTYPE=4 is coded.

APPN:
Specifies the transmission group (TG) number that represents the connection between this node and the destination APPN node. For APPN, this operand is valid when PUTYPE=2.

By specifying a \textit{tg\_number}, you indicate a preferred TG that might be used depending on how the partner nodes negotiate the number.

For APPN-capable nodes, the CPNAME and NETID operands are required with the TGN operand. Note that you can specify the CPNAME operand without the TGN operand and the transmission group number will be negotiated. However, if a value for TGN is specified, then a value for CPNAME and NETID must also be specified.

\textbf{TGN=}tg\_number

Specifies a decimal number assigned to the transmission group.

For APPN, coding TGN=0 means that the PU does not support parallel TGs.

\textbf{TGN=}ANY

For subarea, allows the adjacent node to accept whatever valid transmission group number it receives on the XID. You can specify TGN=ANY for only one end of the transmission group. For the other end, specify a valid TGN number or let it default to 1. If you specify TGN=ANY for both ends, an error occurs. For APPN, coding TGN=ANY, or not coding TGN, allows the transmission group number to be negotiated. For connections to LEN nodes, or to any node that does not support parallel TGs, TGN=0 is always assumed.

\textbf{Note:} VTAM does not validate TGN when coded for subarea resources.

\textbf{TGP}
TGP

```
TGP=tgp_name
```

**statements:**  GROUP, LINE, PU  
**dependencies:**  APPN only  

Specifies the name of a transmission group (TG) profile definition. The characteristics of the TG profile (along with any modifiers) become the characteristics of the PU. If TGP is not specified or has not been activated when the PU becomes active, default TG characteristics are assigned.

See "APPN transmission group profile" on page 720 for more information on transmission group profiles.

TOPO

```
TOPO=CONNECT
```

**statements:**  PU  
**dependencies:**  APPN only  

Controls the way APPN connections using this link station will be reported to APPN topology and routing services for inclusion into APPN functions.

**Note:** Except in the switched major node, the only value for TOPO is CONNECT. If you specify APPN on the CONNTYPE start option, it is not necessary to code TOPO, as you automatically get a value of CONNECT.

TOPO=CONNECT

APPN topology and routing services receives information about the connection and its characteristics when the connection is activated. When the connection is deactivated, APPN topology and routing services receives notification that the transmission group is inactive.

UPARM1
UPARM1

`UPARM1 = 128`  
`UPARM1 = user-defined_operand_value`

**Statements:** `GROUP, LINE, PU`  
**Dependencies:** `APPN only`  
**Range:** `0–255`

Specifies a user-defined operand value.

See [z/OS Communications Server: SNA Network Implementation Guide](http://www.ibm.com) for an example of using the UPARM operands.

---

**UPARM2**

`UPARM2 = 128`  
`UPARM2 = user-defined_operand_value`

**Statements:** `GROUP, LINE, PU`  
**Dependencies:** `APPN only`  
**Range:** `0–255`

Specifies a user-defined operand value.

---

**UPARM3**
UPARM3 = 128

UPARM3 = user-defined_operand_value

**statements:**

**GROUP, LINE, PU**

**dependencies:**

**APPN only**

Specifies a user-defined operand value.

**USE**

**statements:**

**GROUP, LINE**

**Note:** code USE=REDEF, USE=SPARE only on LINE.

**dependencies:**

**VTAM and NCP**

Specifies the initial operating condition of the line for lines that can operate alternately in network control mode and emulation mode.

USE also allows you to specify SDLC lines in the NCP generation for line slots that do not have physical lines attached, but which can be attached later, and allows you to redefine a line that is attached.

See **NCP, SSP, and EP Resource Definition Reference** for proper coding of the USE operand.

**USE=EP**

Indicates that the line operates initially in emulation mode. Prevents the line from being activated unless a VARY ACT command is issued for it. Activating such a line with a VARY command gives the line to VTAM and NCP, whether or not it is in use by EP.

**USE=NCP**

Specifies that the line operates initially in network control mode.

**USE=REDEF**

Specifies that the line is being used, but can be changed to a spare line. The mode of operation is assumed to be NCP. USE=REDEF is used with the MODIFY command to change it from a defined line to a spare line and vice versa. Note that a line defined with USE=REDEF must be the only line in a line group.

**USE=SPARE**

Specifies that the line is being defined for later use, although there is no line attached to that line slot in the NCP.
The mode of operation is assumed to be NCP. Note that a line defined with USE=SPARE must be the only line in a line group.

**USSTAB**

```
USSTAB = ISTINCDT
USSTAB = uss_table_name
```

*statements:* GROUP, LINE, PU, LU, CLUSTER, TERMINAL
*dependencies:* VTAM only; dependent LUs only

Specifies the name of a USS table that VTAM uses to process character-coded input that it receives from the logical unit. USSTAB is valid for dependent logical units only.

A terminal user can issue a USS command with the LANGTAB operand. This causes a second USS table to be associated with the logical unit, which overrides the table specified with USSTAB. If you do not code USSTAB and a LANGTAB USS table is not in use, the IBM-supplied USS table (ISTINCDT) is used. For more information on USS tables, see "Unformatted system services tables" on page 1069.

If you code USSTAB on a CLUSTER or TERMINAL definition statement, you must also code PU=YES on the GROUP or LINE definition statement.

**VERALSID**
Notes:
1. VERALSID=NO is the only valid value if the TRLE operand is coded.

```
statements:           GROUP, LINE, PU
dependencies:        APPN only
```

Specifies whether the adjacent link station name should be used to determine the link station.

**VERALSID=NO**

Specifies that the adjacent link station name is not used to determine the link station.

**VERALSID=YES**

Specifies that the adjacent link station name is used to determine the link station. When specified as YES, the connection endpoints must supply the same name in the CV0EF7 vector that is appended to XID3s. The name supplied in CV0EF7 is the name of a switched PU or the name of a PU under a non-switched line.

**VFYC**
Specifies what action VTAM should take if the correlators on the NCP load module and the NCP/EP definition facility (NDF) resource resolution table (RRT) do not match. When the correlators do not match, VTAM can take one of the following actions:

**VFYC=IGNORE**
- Specifies that VTAM continues the current activation process.

**VFYC=NO**
- Specifies that VTAM reloads the communication controller.

**Note:** If VFYC=NO, an interruption of all the NCP-attached hosts occurs. The NCP load module and the RRT in this host must match. If they do not match, a repeated reload of the NCP occurs until an operator deactivates it.

**VFYC=YES**
- Specifies that VTAM asks the operator whether to reload the communication controller with the requested NCP, stop the activation process, or ignore the mismatch.

VFYC is checked only after a VFYLM check reveals a match between the load module name and the subarea.

**VFYLM**
Specifies how VTAM handles mismatches between the name and subarea of the load module currently in the NCP and the NEWNAME operand on the BUILD definition statement. When the names or subareas do not match, one of the following actions occurs:

**VFYLM=NO**
- Specifies that the operator must load the NCP with the different load module.

**VFYLM=YES**
- Specifies that VTAM asks the operator whether to reload the communication controller with the requested NCP or stop the activation process.

When the load module name matches VTAM's major node name, the value of AUTOSYN applies.

### VNGROUP

(1) (2)

VNGROUP — group_name

**Notes:**
1. Code VNGROUP if you code VNNAME (or VNTYPE=GLOBAL if you default the name to IP.IP)
2. VNNAME and VNTYPE can not be specified on the GROUP definition statement when that GROUP is specified on the VNGROUP operand on the PORT definition statement.

**Statements:**
- GROUP, LINE

**Dependencies:**
- VTAM only; VNNAME; ECLTYPE=PHYSICAL on GROUP

Specifies the name of the logical GROUP containing dial-out links available for use on the connection network named on the VNNAME operand.

For sessions that are routed through the connection network, a dynamic PU is created to represent the destination node. In effect, DYNPU=YES is enforced automatically when a session is established through the connection network. The name specified on the VNGROUP operand provides the name of the group for the dynamic PU that is allocated.

### VNNAME
Notes:

1. On the PORT statement for all connections other than Enterprise Extender, code VNNAME if you code VNGROUP.

2. On the PORT statement for Enterprise Extender connections, VNNAME is not required if VNTYPE=GLOBAL, otherwise code VNNAME if you code VNGROUP.

3. VNNAME is valid on the GROUP statement only if this is an Enterprise Extender XCA major node (MEDIUM=HPHIP on the PORT statement).

Specifies a 1–17 character network-qualified CPNAME for the connection network. VNNAME is reported to the network topology as a virtual node and is treated as an adjacent CP to this node. If the name is not network-qualified, the NETID of the SSCP that owns the connection network (NETID of the host) is used.

For sessions that are routed through the connection network, a dynamic PU is created to represent the destination node. In effect, DYNPU=YES is enforced automatically when a session is established through the connection network.

VPACING
VPACING = \( (\text{max}, \text{min}, \text{type}) \)

**max**
- Specifies the maximum window size. \( \text{max} \) is the maximum number of normal-flow requests that VTAM sends for a given LU-LU session before waiting for a pacing response. No further normal-flow requests can be sent to the logical unit until it is ready to receive more requests.
- \( \text{max} \) can be expressed with leading zeros and is limited to eight digits.

**min**
- Specifies the minimum window size.
- \( \text{min} \) can be expressed with leading zeros and is limited to eight digits.

**type**
- Specifies the pacing type: FIXED or ADAPTIVE (ADAP).
- \( \text{type} \) is used only by NCP, and it requires at least NCP V5R4.

Defaults for VPACING follow:
- If none of the values are specified, the defaults are VPACING=(2,1,ADAP).
- If only \( \text{max} \) is specified, its value is also used for \( \text{min} \). \( \text{type} \) defaults to ADAP.
- If \( \text{max} \) and \( \text{min} \) are specified, and \( \text{max} \) equals 0, then the defaults are VPACING=(2,1,ADAP).
- If \( \text{max} \) and \( \text{min} \) are specified, and \( \text{max} \) does not equal 0, \( \text{max} \) must be greater than or equal to \( \text{min} \) for those values to be used. \( \text{type} \) is set to ADAP. If \( \text{max} \) is less than \( \text{min} \), the values are set to VPACING=(2,1,ADAP).
If `max` and `type` are specified, `min` takes the value of `max`.
If only `type` is specified, `max` and `min` are set to 1.

For more information about pacing, see z/OS Communications Server: SNA Network Implementation Guide.

**VTAMTOPO**

- **VTAMTOPO**
  - **REPORT**
    - **REPORT**
      - **NOREPORT**
      - **NOLINES**
      - **NOSWPUS**
      - **INCLUDE**
      - **IGNORE**

  **statements:** BUILD
  **dependencies:** none

- **VTAMTOPO**
  - **REPORT**
    - **NOREPORT**
    - **NOLINES**
    - **NOSWPUS**

  **statements:** GROUP
  **dependencies:** (switched logical line)
Specifies whether the VTAM topology agent reports the major nodes or subordinate minor nodes to the CMIP manager application programs for SNA local topology.

Specifying VTM.VTAMTOPO allows VTAMTOPO to be handled appropriately by the NCP Definition Facility for versions of NDF that do not recognize VTAMTOPO natively.

Note: See Appendix C, “VTAMTOPO filtering option reporting,” on page 1373 for more information about VTAMTOPO filtering option reporting.

<table>
<thead>
<tr>
<th>Logical line type</th>
<th>Dependency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Token Ring</td>
<td>DIAL=YES and ECLTYPE=LOGICAL</td>
</tr>
<tr>
<td>Frame Relay</td>
<td>DIAL=YES and FRELAY=LOGICAL or LOGICAL, PERIPHERAL</td>
</tr>
<tr>
<td>X25 (NPSI, ODLC)</td>
<td>PHYSRSC=puname and DIAL=YES</td>
</tr>
</tbody>
</table>

**VTAMTOPO=REPORT**

On **BUILD**: Specifies that the VTAM topology agent reports this NCP major node and all subordinate minor nodes, including connected switched PUs under switched logical lines. This does not override any value specified on the **GROUP** or **LINE**.

On **GROUP**: Specifies that the VTAM topology agent reports all subordinate lines and connected switched PUs. VTAMTOPO is valid only for a switched logical line group, see Table 33. This does not override any value specified on the **LINE**.

On **LINE**: Specifies that the VTAM topology agent reports this logical **LINE**. VTAMTOPO is valid only for a switched logical line, see Table 33.

**VTAMTOPO=NOREPORT**

On **BUILD**: Specifies that the VTAM topology agent does not report the switched logical lines or connected switched PUs. The topology agent still reports the major node and the other subordinate resources. This does not override any value specified on the **GROUP** or **LINE**.

On **GROUP**: Specifies that the VTAM topology agent does not report any subordinate lines and connected switched PUs. VTAMTOPO is valid only for a switched logical line group. This does not override any value specified on the **LINE**.

On **LINE**: Specifies that the VTAM topology agent does not report this logical **LINE**. VTAMTOPO is valid only for a switched logical line, see Table 33.
**VTAMTOPO=NOLLINES**

**On BUILD:** Specifies that the VTAM topology agent does not report the switched logical lines. The topology agent still reports the major node and the other subordinate resources, including the connected switched PUs. This does not override any value specified on the GROUP or LINE.

**On GROUP:** Specifies that the VTAM topology agent does not report any subordinate switched logical lines. The topology agent still reports the connected switched PUs under the subordinate switched logical lines. VTAMTOPO is valid only for a switched logical line group. This does not override any value specified on the LINE.

**VTAMTOPO=NOSWPUS**

**On BUILD:** Specifies that the VTAM topology agent does not report the connected switched PUs. The topology agent still reports the major node and the other subordinate resources, including the switched logical lines. This does not override any value specified on the GROUP.

**On GROUP:** Specifies that the VTAM topology agent does not report any connected switched PUs under the subordinate switched logical lines. VTAMTOPO is valid only for a switched logical line group. The topology agent still reports the subordinate switched logical lines.

**VTAMTOPO=INCLUDE**

**On BUILD:** Specifies that the VTAM topology agent reports this NCP major node and all subordinate minor nodes. VTAMTOPO=INCLUDE overrides any other value specified on the GROUP or LINE.

**VTAMTOPO=IGNORE**

**On BUILD:** Specifies that the VTAM topology agent does not report this NCP major node nor any subordinate minor nodes. VTAMTOPO=IGNORE overrides any other value specified on the GROUP or LINE.

**XID**
Specifies whether a physical unit on a non-switched SDLC line can identify itself to VTAM through XID.

**XID=NO**
- Indicates an XID is not sent to the physical unit.
- For non-switched frame relay PUs, code XID=NO.

**XID=YES**
- Indicates an XID is sent to the physical unit. Code for all physical units that have the ability to receive and respond to an XID in normal disconnect mode.
- XID=YES is required for type 2.1 peripheral nodes.

**XNETALS**

Determines whether the network ID of an attaching node or VTAM’s network ID is used for an adjacent nonnative node. Used to control how a nonnative node can attach to VTAM.

**Note:** The XNETALS option has changed in the way VTAM treats PUs that send in nonnative NETIDs in their XID. VTAM used to let the PU connect and treated it as if it sent in the same NETID that VTAM started with but that is no longer true. If the NETID that the PU sent in its XID is not the same as VTAM’s and XNETALS=NO or is defaulted, IST680I will be issued.

**XNETALS=NO**
- Indicates that VTAM’s network ID will be used to represent the connecting node's network ID.
  - During connection:
    - If the NETID operand is specified on the PU statement, then it must specify VTAM’s network ID.
    - If the NETID operand is not coded on the PU statement, then VTAM’s network ID is assumed and the adjacent node’s network ID is ignored.
  - During LU-LU session establishment:
If the network ID for the originating LU is specified (during connection time) and is not identical to VTAM's network ID, the session will not be established.

If the originating LU is not using a network qualified name, VTAM’s network ID is assumed and the session will be established.

**Note:** If the PU is a dependent LU server or requester, nonnative network connection is not allowed if XNETALS=NO is specified and the NETID operand is not coded.

**XNETALS=Yes**

Indicates that the network ID of the adjacent PU will be used for connection and session establishment.

During connection:

- If the NETID operand on the PU statement is coded, it will be used as the PU's network ID. In this case, the network ID specified on the PU statement must match the network ID specified by the attaching node during connection processing.
- If the NETID operand is not specified on the PU statement, then the network ID specified by the attaching node during connection processing will be assumed.

During LU-LU session establishment:

- If the network ID for the originating LU is specified and does not match the PU's network ID, the session will not be established.
- If the originating LU name is not a network qualified name, the session will be established. VTAM will qualify the LU’s name with the attaching node’s network ID.

**Note:** Providing that XNETALS=YES is coded, you can prohibit nonnative network attachment for LU-LU sessions for a particular PU by coding XNETALS=NO on the GROUP, LINE or PU statements.

When using XNETALS=NO, VTAM indicates to the NCP that this PU will use VTAM’s native network ID.

### Switched Major Node

To define a switched major node containing definitions for switched connections to subarea or peripheral nodes, code a switched line within a local switched line group, and code a corresponding remote physical unit in the switched major node.

The switched lines are defined within switched line groups within the following major nodes:

- External communication adapter (XCA)
- NCP

These switched connections can be made to a communication controller that is attached to a host processor. These connections can also be made to a host through an IBM 3172 Nways Interconnect Controller, an Open Systems Adapter, or through an IP network for Enterprise Extender.

The remote physical unit can be a type 1, 2, 4, or 5 physical unit that is defined within a switched major node.
PU and PATH definition statements for a switched subarea

The PU definition statements define subareas attached over switched lines that can be dialed into or dialed out from a communication controller using NCP.

PU and PATH definition statements for the XCA major node

The PU definition statement defines peripheral physical units on switched lines that can be dialed into or dialed out from the host through an IBM 3172 Nways Interconnect Controller, an Open Systems Adapter, or through an IP network using Enterprise Extender. The lines are defined in switched line groups in the external communication adapter major node. For dial-out operations, the PATH definition statement defines the paths used to connect the physical unit to the host through an IBM 3172 Nways Interconnect Controller, an Open Systems Adapter, or through an IP network using Enterprise Extender.

Table 34. Switched major node definition statements and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=SWNET CONFIGDS CONFIGPW VTAMTOPO</td>
<td>650</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>GROUP</td>
<td>VTAMTOPO</td>
<td>655</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td>ADDR</td>
<td>575</td>
<td>R</td>
<td></td>
<td></td>
<td>non-switched connections</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>ANS</td>
<td></td>
<td>576</td>
<td>G</td>
<td>STOP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>ASDP</td>
<td></td>
<td>576</td>
<td>G</td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>AUTHLEN</td>
<td></td>
<td>578</td>
<td>G</td>
<td>YES</td>
<td></td>
<td>Type 2.1; APPN only</td>
</tr>
<tr>
<td>name</td>
<td>CAPACITY</td>
<td></td>
<td>580</td>
<td>G</td>
<td>8K</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>name</td>
<td>CONNTYPE</td>
<td></td>
<td>583</td>
<td>G</td>
<td>CONNTYPE start option</td>
<td>APPN only; NODETYPE</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>COSTBYTE</td>
<td></td>
<td>584</td>
<td>G</td>
<td>0</td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>name</td>
<td>COSTTIME</td>
<td></td>
<td>584</td>
<td>G</td>
<td>0</td>
<td></td>
<td>APPN only; TOPO</td>
</tr>
<tr>
<td>name</td>
<td>CPCP</td>
<td></td>
<td>584</td>
<td>G</td>
<td>CPCP start option</td>
<td>APPN only</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>CPNAME</td>
<td></td>
<td>585</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>DATMODE</td>
<td></td>
<td>586</td>
<td>G</td>
<td>HALF</td>
<td></td>
<td>Type 2.1</td>
</tr>
<tr>
<td>name</td>
<td>DISCNT</td>
<td></td>
<td>589</td>
<td>G</td>
<td>(NO)</td>
<td></td>
<td>PUTYPE=2</td>
</tr>
<tr>
<td>name</td>
<td>DWACT</td>
<td></td>
<td>598</td>
<td></td>
<td>NO</td>
<td></td>
<td>Enterprise Extender only</td>
</tr>
<tr>
<td>name</td>
<td>DWINOP</td>
<td></td>
<td>598</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>DYNADJCP</td>
<td></td>
<td>599</td>
<td>G</td>
<td>DYNADJCP start option</td>
<td>APPN only; CPNAME</td>
<td></td>
</tr>
</tbody>
</table>
Table 34. Switched major node definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYNLU</td>
<td>600</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>DYNLU start option</td>
<td>PUTYPE=2; CDRDYN</td>
</tr>
<tr>
<td>EEVERIFY</td>
<td>601</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>EEVERIFY start option</td>
<td>Enterprise</td>
</tr>
<tr>
<td>ENCTYPE</td>
<td>603</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
<td>DES</td>
<td>Enterprise</td>
</tr>
<tr>
<td>ENCR</td>
<td>602</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
<td>NONE</td>
<td>Enterprise</td>
</tr>
<tr>
<td>HPR</td>
<td>608</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AppN only</td>
</tr>
<tr>
<td>HPREARBAR</td>
<td>609</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Enterprise</td>
</tr>
<tr>
<td>HPRPSDLY</td>
<td>609</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>HPRPSDLY start option</td>
<td>Enterprise</td>
</tr>
<tr>
<td>IDBLK</td>
<td>611</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CPNAME; IDNUM</td>
<td></td>
</tr>
<tr>
<td>INCLUDOE</td>
<td>613</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>PUTYPE=4,5</td>
</tr>
<tr>
<td>INCLUDSL</td>
<td>613</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>YES</td>
<td>AppN only</td>
</tr>
<tr>
<td>IRETRY</td>
<td>615</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>Putype=1,2</td>
</tr>
<tr>
<td>ISTATUS</td>
<td>616</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>ACTIVE</td>
<td></td>
</tr>
<tr>
<td>LANTEST</td>
<td>616</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>LOCAL</td>
<td></td>
</tr>
<tr>
<td>LIMRES</td>
<td>617</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>LIMQSINT; LU 6.2; PUTYPE=1,2; switched dynamics</td>
</tr>
<tr>
<td>LLERP</td>
<td>619</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NOTPREF</td>
<td>Type 2.1; HPR=YES</td>
</tr>
<tr>
<td>LUGROUP</td>
<td>621</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LUGROUP</td>
<td></td>
</tr>
<tr>
<td>LUSEED</td>
<td>622</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXDATA</td>
<td>622</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>261 (PU type 1)</td>
<td></td>
</tr>
<tr>
<td>MAXOUT</td>
<td>623</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>265 (PU type 2)</td>
<td></td>
</tr>
<tr>
<td>MLTGPRI</td>
<td>625</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MTU</td>
<td>626</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NATIVE</td>
<td>629</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NETID</td>
<td>630</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NN</td>
<td>632</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PASSLIMIT</td>
<td>633</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>PUTYPE=1,2</td>
</tr>
<tr>
<td>PDELAY</td>
<td>634</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>TERRESTR</td>
<td>AppN only</td>
</tr>
<tr>
<td>PRTCT</td>
<td>635</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PUTYPE=4,5</td>
</tr>
<tr>
<td>PUTYPE</td>
<td>635</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>SECNET</td>
<td>640</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>PUTYPE=1,2</td>
</tr>
<tr>
<td>SECURITY</td>
<td>641</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UNSECURE</td>
<td>AppN only</td>
</tr>
<tr>
<td>SPAN</td>
<td>644</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>SUBAREA</td>
<td>645</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>in_xid_value</td>
<td>PUTYPE=4,5</td>
</tr>
</tbody>
</table>

Chapter 2. Major nodes 553
Table 34. Switched major node definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>TGCONF</td>
<td></td>
<td></td>
<td>646</td>
<td></td>
<td></td>
<td>MULTI,NORMAL</td>
<td>PUTYPE=4, Not valid for X.25</td>
</tr>
<tr>
<td>TGN</td>
<td></td>
<td></td>
<td>647</td>
<td></td>
<td></td>
<td></td>
<td>see description</td>
</tr>
<tr>
<td>TGP</td>
<td></td>
<td></td>
<td>648</td>
<td>G</td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>TOPO</td>
<td></td>
<td></td>
<td>649</td>
<td></td>
<td></td>
<td>CONNECT</td>
<td>APPN only</td>
</tr>
<tr>
<td>U Parm1</td>
<td></td>
<td></td>
<td>650</td>
<td>G</td>
<td></td>
<td>128</td>
<td>APPN only</td>
</tr>
<tr>
<td>U Parm2</td>
<td></td>
<td></td>
<td>650</td>
<td>G</td>
<td></td>
<td>128</td>
<td>APPN only</td>
</tr>
<tr>
<td>U Parm3</td>
<td></td>
<td></td>
<td>650</td>
<td>G</td>
<td></td>
<td>128</td>
<td>APPN only</td>
</tr>
<tr>
<td>VERALSID</td>
<td></td>
<td></td>
<td>652</td>
<td>G</td>
<td></td>
<td>NO</td>
<td>APPN only</td>
</tr>
<tr>
<td>VTAMTOPO</td>
<td></td>
<td></td>
<td>655</td>
<td></td>
<td></td>
<td></td>
<td>XNETALS start option value</td>
</tr>
<tr>
<td>XNETALS</td>
<td></td>
<td></td>
<td>656</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PATH</td>
<td>CALL</td>
<td>579</td>
<td></td>
<td></td>
<td>INOUT</td>
<td>GRPNM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DIALNO</td>
<td>587</td>
<td></td>
<td></td>
<td></td>
<td>LINENM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DL CADDR</td>
<td>591</td>
<td></td>
<td></td>
<td></td>
<td>DLCADDR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HOSTNAME</td>
<td>607</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>IP ADDR</td>
<td>614</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LINENM</td>
<td>618</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PID</td>
<td>634</td>
<td></td>
<td></td>
<td></td>
<td>DIALNO</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RED DELAY</td>
<td>636</td>
<td></td>
<td></td>
<td></td>
<td>DLCADDR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>REDIAL</td>
<td>637</td>
<td></td>
<td></td>
<td>30</td>
<td>Enterprise Extender only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SAP ADDR</td>
<td>639</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SHM</td>
<td>642</td>
<td></td>
<td></td>
<td>NO</td>
<td>DIALNO</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SHMTIM</td>
<td>643</td>
<td></td>
<td></td>
<td></td>
<td>GRPNM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USE</td>
<td>651</td>
<td></td>
<td></td>
<td>YES</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>VERID</td>
<td>653</td>
<td></td>
<td></td>
<td></td>
<td>GRPNM; VERID</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VERIFY</td>
<td>653</td>
<td></td>
<td></td>
<td>NONE</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>LU</td>
<td>LOCADDR</td>
<td>619</td>
<td>R</td>
<td>G</td>
<td></td>
<td>ASLTAB PUTYPE=1,2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AS LENT</td>
<td>577</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASLTAB</td>
<td>577</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 34. Switched major node definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>BATCH</td>
<td></td>
<td></td>
<td>578</td>
<td>G,P</td>
<td>NO</td>
<td></td>
<td>pre-V4R3 NCPs PUTYPE=1,2</td>
</tr>
<tr>
<td>CERTIFY</td>
<td></td>
<td></td>
<td>580</td>
<td>P</td>
<td>NO</td>
<td></td>
<td>ENCR</td>
</tr>
<tr>
<td>CKEYNAME</td>
<td></td>
<td></td>
<td>581</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLRSESSQ</td>
<td></td>
<td></td>
<td>581</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLOGMOD</td>
<td></td>
<td></td>
<td>596</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENCR</td>
<td></td>
<td></td>
<td>602</td>
<td>G,P</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENCRETYP</td>
<td></td>
<td></td>
<td>603</td>
<td>G,P</td>
<td>DES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FASTPASS</td>
<td></td>
<td></td>
<td>604</td>
<td>G,P</td>
<td>YES</td>
<td></td>
<td>TSO only</td>
</tr>
<tr>
<td>FEATUR2</td>
<td></td>
<td></td>
<td>605</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISTATUS</td>
<td></td>
<td></td>
<td>616</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOGAPPL</td>
<td></td>
<td></td>
<td>620</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOCTAB</td>
<td></td>
<td></td>
<td>621</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXSESS</td>
<td></td>
<td></td>
<td>624</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDLENT</td>
<td></td>
<td></td>
<td>624</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDTAB</td>
<td></td>
<td></td>
<td>625</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODETAB</td>
<td></td>
<td></td>
<td>626</td>
<td>G,P</td>
<td>ISTINCLM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PACING</td>
<td></td>
<td></td>
<td>628</td>
<td>G,P</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGISTER</td>
<td></td>
<td></td>
<td>637</td>
<td>P</td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td>RESETLU</td>
<td></td>
<td></td>
<td>638</td>
<td>P</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESSCB</td>
<td></td>
<td></td>
<td>639</td>
<td></td>
<td>0</td>
<td></td>
<td>independent LU</td>
</tr>
<tr>
<td>SPAN</td>
<td></td>
<td></td>
<td>644</td>
<td></td>
<td></td>
<td></td>
<td>NetView</td>
</tr>
<tr>
<td>SSCPFM</td>
<td></td>
<td></td>
<td>644</td>
<td>G,P</td>
<td>FSS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TERM</td>
<td></td>
<td></td>
<td>646</td>
<td>G,P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USSTAB</td>
<td></td>
<td></td>
<td>651</td>
<td>G,P</td>
<td>ISTINCDT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VPACING</td>
<td></td>
<td></td>
<td>654</td>
<td>G,P</td>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R Required operand.

G Code this operand on the higher-level GROUP definition statement to take advantage of the sift effect, or code it on this definition statement.

P Code this operand on the higher-level PU definition statement to take advantage of the sift effect, or code it on this definition statement.
Restriction: The MAXDLUR, MAXGRP, and MAXNO keywords on the VBUILD definition statement and the MAXPATH keyword on the PU definition statement are obsolete. If these keywords are defined, the values are not checked and the keywords are ignored. No messages are issued.

Switched major node coding notes

This information contains coding notes for the switched major node.

Enterprise Extender connections

VBUILD definition statement

Code a VBUILD definition statement for each switched major node.

PU definition statement

Code a PU definition statement for each physical unit in the switched major node.

PATH definition statement

Code a PATH definition statement to define a path to a physical unit in a switched major node. You can code up to 256 PATH definition statements for each physical unit. The PATH definition statement must immediately follow the PU definition statement that defines the associated physical unit. VTAM searches the PATH definition statements for an available path in the order given in the configuration deck.

Use the DWACT operand to enable automatic dialing of the PU when it is activated by specifying DWACT=YES. Enable the PU to be redialed automatically in the event of a dial failure by use of the REDIAL and REDDELAY operands. If you also would like redial to be attempted (using the REDIAL and REDDELAY specifications) in the event of a connection INOP, specify DWINOP=YES. See "Switched major node operand descriptions" on page 575 for details.

The IPADDR keyword specifies the remote IP address for a dial-out connection. If the remote APPN node is a z/OS Communications Server node, the address must be the VIPA address the remote node is currently using for the Enterprise Extender connections. For example, the VIPA address specified or defaulted to on the IPADDR VTAM start option or modified by the F VTAMOPTS, IPADDR operator command.

The HOSTNAME keyword specifies the name of the remote (destination) host name for a dial-out connection. IPADDR and HOSTNAME keywords are mutually exclusive. Use of this keyword implies that the local z/OS CS node can perform TCP/IP name-to-address resolution, either by connectivity to a DNS name server, or by using a local hosts file such as HOSTS.SITEINFO or /etc/ipnodes. If the host name is specified and cannot be resolved, the dial will fail. For more information, see z/OS Communications Server: IP Configuration Guide. The DNS name servers and local hosts files must be configured to only return VIPA addresses if the name represents a z/OS CS APPN node, and the VIPA address returned must be one that is usable for Enterprise Extender connections on the remote VTAM node. For information about displaying information on networks and hosts, see z/OS Communications Server: IP User's Guide and Commands.

In order to use HOSTNAME, VTAM will use the system-wide resolver function. If you intend for the resolver to use a name server to resolve HOSTNAME, the
resolver will use the TCPIP.DATA file to determine the IP addresses of name server(s) to use. TCPIP.DATA also contains other necessary information such as the local domain name, local host name, and so on. If you intend to use a local hosts file instead of a name server, there is a specific search order used to find the correct data set. See z/OS Communications Server: IP Configuration Guide for information on the search order for local hosts files.

In order to use HOSTNAME, VTAM must be able to find one or more TCPIP definition files. This is done by way of the Native MVS Sockets resolver. Do not confuse this with the LE Resolver, sometimes referred to as the z/OS Resolver. If you intend to use a name server to resolve HOSTNAME, VTAM must be able to find the TCPIP.DATA file, which in turn, contains the IP addresses of name servers to use. It also contains other necessary information such as the local domain name, local host name, etc. If you intend to use a local HOST table instead of a name server, there also is a specific search order to find this data set. See z/OS Communications Server: IP Configuration Guide for information on the search order for Local Hosts tables.

Note: These data sets cannot be concatenated like many MVS data sets can.

The SAPADDR keyword specifies the remote service access point (RSAP) value.

For predefined Enterprise Extender connections, coding a PATH definition statement with GRPNM allows an incoming call to prefer one switched PU over another. When the GROUP name, associated with the line selected for the inbound call, matches the GRPNM on the switched PATH, the switched PU associated with the PATH is preferred over another switched PU that does not match, or does not have a PATH definition statement coded.

Note: The GRPNM keyword ties the switched major node PATH statement to the XCA major node. The name on the XCA GROUP statement must match the value on the GRPNM in the SMN PATH.

**LU definition statement**

Code an LU definition statement for each logical unit associated with a type 1 or 2 physical unit within a switched major node. The LU definition statement must follow the PU definition statement that defines the physical unit with which the logical unit is associated.

**Dynamic change of operands**

You can dynamically change the following switched major node operands by editing the VTAMLST and then issuing the VARY ACT,UPDATE=ALL command. You can specify the new operand value on a higher-level definition statement for sifting, if applicable.
In addition to the above VTAM operands, you can also dynamically change the following NCP operands, which are passed to the NCP during switched PU connection. For information on these operands, see the NCP, SSP, and EP Resource Definition Reference.

- COMRATE
- DLCI
- DYNWIND
- HPRQQLIM
- INTFACE
- IPQQLIM
- LADDR
- SNETMASK
- T1TIMER
- T2TIMER
To dynamically change an operand on the PU or LU statement, the resource to which it applies must be inactive. For instance, an operand on an LU definition statement that receives its value from a PU definition statement can be dynamically changed only when the LU is inactive; the PU can remain active.

You can dynamically change the operands associated with a PATH statement or tables (ASLTAB, LOGTAB, MDLTAB, MODETAB, and USSTAB) when the PU is active, connectable, or inactive. You cannot change the PATH operands when the PU is in pending state.

See z/OS Communications Server: SNA Operation and the z/OS Communications Server: SNA Network Implementation Guide for more information on the VARY ACT,UPDATE=ALL command.

Note: For a switched major node, you cannot add APPN capability or SDDLU support to a PU with the dynamic change function. If a PU has not been coded as APPN-capable, you cannot add the APPN operands to it. Likewise, if you have not coded LUGROUP (which allows SDDLU support) on a PU definition statement, you cannot add it with dynamic change.

**NODETYPE start option**

The NODETYPE start option allows APPN function. The combination of NODETYPE and the HOSTSA start option determines the various configurations of subarea node, interchange node, migration data host, network node, or end node.

Note that if you do not code the NODETYPE start option, any APPN operands you code in this major node are ignored.

**XCA and NCP connections**

**VBUILD definition statement**

Code a VBUILD definition statement for each switched major node.

**PU definition statement**

Code a PU definition statement for each physical unit in the switched major node. Operands specified on a PU definition statement override any specified under the NCP.

**PATH definition statement**

Code a PATH definition statement to define a path to a physical unit in a switched major node. You can code up to 256 PATH definition statements for each physical unit. The PATH definition statement must immediately follow the PU definition statement that defines the associated physical unit. VTAM searches the PATH definition statements for an available path in the order given in the configuration deck.

A PATH definition statement is required if data terminal equipment (DTE) ID verification or call selection is to be used on an incoming call.
LU definition statement

Code an LU definition statement for each logical unit associated with a type 1 or 2 physical unit within a switched major node. The LU definition statement must follow the PU definition statement that defines the physical unit with which the logical unit is associated.

Dynamic change of operands

You can dynamically change the following switched major node operands by editing the VTAMLST and then issuing the VARY ACT,UPDATE=ALL command. You can specify the new operand value on a higher-level definition statement for sifting, if applicable.

ADDR
ANS
ASDP
ASLENT
ASLTAB
BATCH
CALL
CAPACITY
CONNTYPE
COSTBYTE
COSTTIME
CPCP
DATMODE
DIALNO
DISCNT
DLCADDR
DLOGMOD
DLURNAME
DYNADJCP
ENCR
ENCRTYPE
FASTPASS
FEATUR2
GID
GRPNM
IDBLK
IDNUM
IPRESOLV
IRETRY
ISTATUS
LIMRES
LINENM
LOCADDR
LOGAPPL
LOGTAB
LUGROUP
LUSEED
MAXDATA
MAXOUT
MAXSESS
MDLENT
MDLTAB
NMODETAB
NN
PACING
PASSLIM
PDELAY
PID
PUTYPE
REDDELAY
REDIAL
REGISTER
SECURITY
SECNET
SHIM
SHMTIM
SSCPFM
TERM
TGP
TOPO
UPARM1
UPARM2
UPARM3
USE
USSTAB
VERALSID
VERID
VERIFY
VPACING

In addition to the above VTAM operands, you can also dynamically change the following NCP operands, which are passed to the NCP during switched PU connection. For information on these operands, see the NCP, SSP, and EP Resource Definition Reference.

- COMRATE
- DLCI
- DYNWIND
To dynamically change an operand on the PU or LU statement, the resource to which it applies must be inactive. For instance, an operand on an LU definition statement that receives its value from a PU definition statement can be dynamically changed only when the LU is inactive; the PU can remain active.

You can dynamically change the operands associated with a PATH statement or tables (ASLTAB, LOGTAB, MDLTAB, MODETAB, and USSTAB) when the PU is active, connectable, or inactive. You cannot change the PATH operands when the PU is in pending state.

See z/OS Communications Server: SNA Operation and the z/OS Communications Server: SNA Network Implementation Guide for more information on the VARY ACT,UPDATE=ALL command.

Note: For a switched major node, you cannot add APPN capability or SDDLU support to a PU with the dynamic change function. If a PU has not been coded as APPN-capable, you cannot add the APPN operands to it. Likewise, if you have not coded LUGROUP (which allows SDDLU support) on a PU definition statement, you cannot add it with dynamic change.

**NODETYPE start option**

Note that if you do not code the NODETYPE start option, any APPN operands you code in this major node are ignored.

The NODETYPE start option allows APPN function. The combination of NODETYPE and the HOSTSA start option determines the various configurations of subarea node, interchange node, migration data host, network node, or end node.

**XCA and NCP connections full syntax**

The full syntax for the switched major node follows. The syntax for each operand is repeated in “Switched major node operand descriptions” on page 575.

**VBUILD**
Connect to topology (TOPOCONNECT) with the following operands:

- **TOPO** = connection_cost
- **UPARM1** = user-defined_operand_value
- **UPARM2** = user-defined_operand_value
- **UPARM3** = user-defined_operand_value
- **VERALSID** = NO
- **XNETALS** = value_of_XNETALS_start_option
- **VTAMTOPO** = NOREPORT

**DISCNT** values:

- \( \text{DISCNT} \) values:
  \[ \text{DISCNT} \text{values} = \{ \text{YES}, \text{NF} \} \]

**DISCNT DELAY** values:

- \( \text{DISCNT DELAY} \) values:
  \[ \text{DISCNT DELAY values} = \{ \text{DELAY}, \text{NF}, \text{delay_time} \} \]

**Notes:**

1. Capacity defaults to 25M for ATM native connections.
2. **CONNTYPE**=APPN is the only valid value if the TRLE operand is coded.
3. **CPCP** defaults to YES for the model RTP PU.
CPCP defaults to the value of the CPCP start options for all PUs except the model RTP PU.

DYNADJCP=YES is the only valid value if the TRLE operand is coded.

LIMRES=NO is the only valid value if the TRLE operand is coded.

MAXDATA defaults to 261 when PUTYPE=1 is coded.

MAXDATA defaults to 265 when PUTYPE=2 is coded.

NEGLIGIB is the default if the TRLE operand is coded.

TGN defaults to 1 for subarea.

TGN defaults to ANY for APPN.

VERALSID=NO is the only valid value if the TRLE operand is coded.

**PATH**

**LU**
Notes:
1  REGISTER defaults to NETSRVR for dependent LUs.
2  REGISTER defaults to NO for independent LUs.

Enterprise Extender connections - full syntax
The full syntax for the switched major node follows. The syntax for each operand is repeated in Switched major node operand descriptions on page 575.

VBUILD

```
  VBUILD TYPE = SWNET
    name
    CONFIGS = configds_name
    CONFIGPW = configds_password
    VTAMTOPO = REPORT
```

PU
Notes:
1 Capacity defaults to 25M for ATM native connections.
2 NEGLIGIB is the default if the TRLE operand is coded.
3 TGN defaults to ANY for APPN.
VTAMTOPO = REPORT

XNETALS = value_of_XNETALS_start_option

PATH

name

CALL = INOUT
CALL = IN
CALL = OUT

DLURNAME = dlur_name

GID = group_identifier

GRPNM = group_name

HOSTNAME = remote_host_name (¹)
IPADDR = IP_address (¹)

IPRESOLV = (0)
IPRESOLV = (0-255)

LINENM = line_name

PID = path_identifier

REDDELAY = 30
REDDELAY = number_of_seconds_between_redials

REDIAL = 3
REDIAL = number_of_redials

SAPADDR = 8
SAPADDR = address

USE = YES

VERID = call_identifier

VERIFY = NONE
VERIFY = IN
VERIFY = INOUT
VERIFY = OUT

Note:
1. HOSTNAME and IPADDR are mutually exclusive, however, IPADDR is used if both are specified.
Local (source) and remote (destination) service access point determination

The local (source) service access points (SAPs) and remote (destination) SAPs are determined as follows:

<table>
<thead>
<tr>
<th>SAP type</th>
<th>Predefined TGs</th>
<th>Connection network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local SAP</td>
<td>XCA major node SAPADDR keyword on PORT statement. Default is 4.</td>
<td>XCA major node, SAPADDR on PORT statement.</td>
</tr>
<tr>
<td>Remote SAP</td>
<td>Switched major node, SAPADDR keyword on PATH statement. Default is 8.</td>
<td>Learned during connection.</td>
</tr>
</tbody>
</table>

Note:
1. These SAP pairs are used in LDLC signals.
2. For more information on SAP pairs see z/OS Communications Server: SNA Network Implementation Guide.

Switched major node operand descriptions

The following information describes the operands you can code for the switched major node.

ADDR
**ADDR**—link_station_address

*statements:*  
*dependencies:*

Specifies the hexadecimal SDLC station address for the physical unit. You can specify one or two hexadecimal digits for ADDR. For example, if the address is X'03', you can code ADDR=03 or ADDR=3.

This address is required for all physical units attached through SDLC lines and X.25 networks.

This operand is ignored for switched subarea connections.

**ANS**

*statements:*  
*dependencies:*

**Note:** This operand is valid for subarea and PU Type 2 connections.

Determines whether the physical unit continues to operate when the NCP enters automatic network shutdown. The value coded on this statement overrides whatever value you code on the ANS definition statement for the NCP.

For switched subarea connections, this operand specifies whether a switched subarea connection over lines attached to this physical unit should remain active or be dropped when the owning SSCP is lost. This is used only on a PU definition statement that defines a subarea physical unit that is connected to a link station within an NCP.

**ANS=CONTINUE** (or CONT)  
Indicates that the LU-LU sessions are to continue.

Code **ANS=CONTINUE** if the terminal participates in XRF sessions.

When a session is continued and takeover occurs, if the LU taken over is a dependent logical unit, it must support ACTLU(ERP) or the session is terminated during takeover processing. An independent logical unit does not receive ACTLU, so this restriction does not apply.

Note that there are several restrictions for session continuity in certain configurations. For information on these restrictions, see [z/OS Communications Server: SNA Network Implementation Guide](#).

**ANS=STOP**  
Indicates that the connection should be dropped.

**ASDP**
ASDP = NO
Indicates that an application cannot supply dial information.

ASDP = YES
Indicates that an application can supply dial information.

ASLENT

Specifies the name of the associated LU table entry to be used for this logical unit.

If you code the ASLENT operand, you must also code the ASLTAB operand. If you omit ASLENT and include ASLTAB, VTAM uses the first entry in the associated LU table by default.

ASLTAB
**ASLTAB**

As an associated LU table name.

**Statements:**

- GROUP, PU, LU

**Dependencies:**

- PUTYPE=1 or 2

Specifies the name of the associated LU table to be used for the logical unit. See "Associated LU table" on page 981 for a description of how to define an associated LU table.

If you omit the ASLTAB operand, VTAM will not provide the names of associated logical units to the application during LU-LU session initiation. However, you can provide the associated LU names in other ways. You can enter them from the terminal on the PRINTER1 and PRINTER2 operands of the LOGON command, or VTAM can send them to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the PRINTER1 and PRINTER2 operands of the LOGON command.

The operator can use the MODIFY TABLE command to dynamically replace an associated LU table (ASLTAB). See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**AUTHLEN**

- **AUTHLEN=NO**
  - VTAM will not pass the specified transmission priority, instead the default priority of medium will be passed.

- **AUTHLEN=YES**
  - VTAM will pass the specified transmission priority.

**BATCH**
Statements:

GROUP, PU, LU

Dependencies:

PUTYPE=1 or 2; pre-V4R3 NCPs only

Specifies the processing priority that the NCP uses for the logical unit.

**BATCH=NO**

BATCH=NO means a high priority (suitable for interactive application programs).

**BATCH=YES**

BATCH=YES means a low priority.

**Note:** This operand applies to pre-V4R3 NCPs only. If you code the BATCH operand and the NCP is V4R3 or later, the operand is ignored.

See the *NCP, SSP, and EP Resource Definition Reference* for more information about this operand.

**CALL**

**CALL=INOUT**

Specifies whether inbound and outbound calls are to be supported by this PATH definition statement. If all PATH definition statements for this physical unit specify CALL=INOUT, an inbound call is accepted from any switched line group. If you specify call selection (CALL=IN or CALL=OUT) on any PATH definition statement, an inbound call is accepted only if it is using a line group for which there is a PATH definition statement that specifies CALL=IN or CALL=INOUT. Therefore, if one PATH definition statement specifies call selection, all valid PATHs (inbound and outbound) must be defined.

If you specify CALL, GRPNM is required.

If CALL=OUT or CALL=INOUT, either DIALNO or LINENM is required.

**CALL=IN**

Specifies that inbound calls can be accepted but outbound calls cannot be made using this PATH definition statement.

**CALL=INOUT**

Specifies that inbound calls can be accepted and outbound calls can be made using this PATH definition statement.
CALL=OUT
Specifies that outbound calls can be made but no inbound calls can be accepted using this PATH definition statement.

Note: See the description of the VERIFY operand if you specify both CALL and VERIFY.

CAPACITY

Notes:
1. Capacity defaults to 25M for ATM native connections.

<table>
<thead>
<tr>
<th>statements:</th>
<th>GROUP, PU</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies:</td>
<td>APPN only</td>
</tr>
<tr>
<td>range:</td>
<td>1 - 1000Kb</td>
</tr>
<tr>
<td></td>
<td>1 - 1000Mb</td>
</tr>
<tr>
<td></td>
<td>1 - 100Gb</td>
</tr>
</tbody>
</table>

Specifies the effective capacity of the link that comprises the transmission group (TG). Specify the value in Kb per second (for example, 100K), Mb per second (for example, 100M), or Gb per second (for example, 10G). This number approximates the bits per second that the link can transmit (the transmission rate of the link, times the maximum load factor expressed as a percentage).

Note:
1. Because the value for capacity is represented as a single byte, the precision of the specified number of bits might be lost. For example, numbers that are close (such as 100K and 101K) can be interpreted by VTAM, and displayed, as the same value. See [Table 45 on page 722](#) for a list of CAPACITY values you can specify and their corresponding values when displayed.

2. The capacity specified for ATM native connections defined as reserved bandwidth should be related to the forward sustainable cell rate (DLCADDR subfield 7). To calculate the data throughput required for a reserved bandwidth connection, multiply the sustainable cell rate by the payload per cell (48 bytes). If the defined CAPACITY is greater than ten times the ATM reserved bandwidth connection capacity, excess storage might be used and the weighting of this TG with others might cause undesirable results.

CERTIFY
CERTIFY = NO

Certifies whether VTAM on behalf of the application program will verify authentication.

**CERTIFY=YES**
- Specifies SLU authentication (verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

**CERTIFY=NO**
- Specifies no SLU authentication (not verifying the SLU is using the same cryptographic key as the PLU) is to be performed by the PLU.

**CKEYNAME**

```
CKEYNAME = name
CKEYNAME = crypto_key_name
```

Specifies the name of the key-encrypting key (KEK) in the cryptographic key data set (CKDS) for the LU. Use CKEYNAME to encipher the session key that is sent in the BIND to the LU when a cryptographic session is established. CKEYNAME has the same format and restrictions as other VTAM names. For information on naming conventions, see "Format of definition statements" on page 8.

If you do not specify CKEYNAME, the cryptographic key name defaults to the name on the LU definition statement.

**Note:** Independent LUs will ignore the CKEYNAME keyword and use their default name.

**CLRSESSQ**
statements:  

Specifies whether a dependent LU is forced into an inhibited session state when the LU is disabled. When the LU is enabled, it is returned to normal status. Sessions, however, are not connected.

**CLRSESSQ=YES**
- Specifies that the LU is forced into an inhibited session state. Both active and pending sessions are removed from the session queue.

**CLRSESSQ=NO**
- Specifies that the LU is not forced into inhibited session state. This is the default.

**CONFGDS**

Specifies a 1–8 character data definition name that identifies the configuration restart data set defined by the user for this major node. Include a DD statement that has this data definition name in the VTAM start procedure.

**CONFGPW**
CONFGPW = confgds_password

Statements: VBUILD
Dependencies: CONFGDS

Specifies the 1–7 character alphanumeric password that VTAM uses to access the configuration restart data set. If you do not code CONFGPW, and it is required by VSAM, VSAM prompts the VTAM operator for the correct password when VTAM attempts to open the data set.

Code CONFGPW only if you also code CONFGDS.

CONNTYPE

Notes:
1. CONNTYPE=APPN is the only valid value if the TRLE operand is coded.

Specifies for a type 2.1 PU whether the connection is to be established as a LEN connection or attempted as an APPN connection.

If you do not code the NODETYPE start option, the CONNTYPE operand is ignored and the connection will be a LEN connection.

CONNTYPE=APPN

Specifies that this connection can support parallel TGs, CP-CP sessions, and CP name change support. CONNTYPE=APPN must be coded for CPCP=YES to be valid.

CONNTYPE=LEN

Specifies that this connection does not support parallel TGs, CP-CP sessions, or CP name change support.

Note: For LEN sessions between PUs when CONNTYPE=APPN is coded on each PU, the CPNAME of each PU must be unique. When CONNTYPE=LEN, the CPNAMEs can be the same.

Because CONNTYPE=LEN indicates that the PU is not APPN-capable, do not code any APPN-only operands on the same PU statement. If an APPN-only operand is coded with CONNTYPE=LEN, VTAM will issue a message warning of a parameter conflict.
Although you need to code CONNTYPE=APPN for a PU to be an APPN resource, this does not necessarily mean the PU will be APPN; a PU's characteristics are determined at activation. However, coding CONNTYPE=LEN means that the PU can only be a LEN resource, even if, at activation, the XID indicates the PU can be an APPN resource.

**COSTBYTE**

```
\[\text{COSTBYTE} = \text{cost\_per\_byte}\]
```

- **Statements:** GROUP, PU
- **Dependencies:** APPN only
- **Range:** 0–255

Specifies a cost-per-byte-transmitted to be associated with the transmission group (TG). A value of 0 is the least expensive cost per byte and 255 is the most expensive.

**COSTTIME**

```
\[\text{COSTTIME} = \text{cost\_per\_unit\_time}\]
```

- **Statements:** GROUP, PU
- **Dependencies:** APPN only, TOPO
- **Range:** 0–255

Specifies a cost-per-unit-of-time to be associated with the transmission group (TG). A value of 0 is the least expensive cost per unit of time and 255 is the most expensive.

If TOPO is coded with a connection cost specified, then the value specified through TOPO will be used as the cost per unit time until the connection is established. The COSTTIME value specified is used while the connection is active.

**CPCP**
Notes:
1 CPCP defaults to YES for the model RTP PU.
2 CPCP defaults to the value of the CPCP start options for all PUs except the model RTP PU.

Specifies whether CP-CP sessions are supported on this connection. If you do not code CPCP, the value defaults to YES only if you specify YES on the CPCP start option. Otherwise, the value defaults to NO.

CPCP=NO
   CP-CP sessions are not supported for this connection.

CPCP=YES
   CP-CP sessions are supported for this connection.

   Note: CPCP=YES is valid only when CONNTYPE=APPN.

CPNAME
CPNAME=control_point_name

statements: PU
dependencies: Type 2.1

Specifies the control point (CP) name of a type 2.1 peripheral node. If a type 2.1 peripheral node supports it, coding CPNAME is required if the peripheral node supports role negotiation (such as 3172, 3174, and Communications Manager).

To dial in, a type 2.1 peripheral node on a switched line requires either CPNAME or both IDBLK and IDNUM on the PU definition statement. However, you can code all three operands.

To dial out without knowing the partner's CPNAME or IDBLK and IDNUM, code CPNAME without a value (CPNAME=).

For dial-in or dial-out, if CPNAME is not in the XID or if you do not code CPNAME, VTAM uses IDBLK and IDNUM instead.

VTAM uses the CP name of a type 2.1 PU to identify the PU in any messages. For an APPN PU, its name (or label) must differ from control_point_name (specified here on CPNAME) to ensure that a CP-CP session establishment succeeds. If name and control_point_name are duplicates, the session establishment fails and sense code 08880001 is given.

When VTAM is defined as a type 2.1 node, code CPNAME the same as SSCPNAME on the other side of the connection. Ensure that SSCPNAME is unique within the host domain.

See the appropriate product manual for CPNAME, IDNUM, and IDBLK values.

**DLUR physical units:**
For DLUR-defined PUs, CPNAME may be specified for a single PU which is non-adjacent to the DLUS (non-adjacent means there is no APPN connection between the DLUS and the control point named through CPNAME). When the DLUR-defined PU is adjacent to the DLUS, or when more than one PU with the same CPNAME must be defined, a different means of identification dependent upon DLC support must be used (for example, IDBLK/IDNUM for internal PUs).

Additional information can be found in the [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/SSLTBK_22.2.0/com.ibm.zos.bks.savc22b.snavc22b.doc/guide/r706285.html) and in the publications for your DLUR implementation.

**DATMODE**
**DATMODE**

Specifies whether the physical unit communicates in half-duplex or full-duplex data mode. The associated line must support simultaneous transmission in both directions.

**DATMODE=FULL**
- Indicates that the physical unit communicates in full-duplex mode.

**DATMODE=HALF**
- Indicates that the physical unit communicates in half-duplex mode.

**DIALNO**

Specifies the dial information used to initiate a connection with a physical unit over a switched link.

If you do not code **LINENM**, **DIALNO** is required if this **PATH** definition statement is for a type 4 or 5 physical unit.

**DIALNO** is mutually exclusive with the **LINENM** operand for any single switched path definition statement.

**Public Telephone Networks:**
**DIALNO** consists of a combination of digits and special characters. If an auto-call unit is installed, the special characters consist of separator characters, dialing pause characters, and end-of-number characters, as follows:

- The separator character causes the auto-call unit to delay sending the next dial digit until it receives a secondary dial tone. For separator characters, use underscores (_) or the hexadecimal equivalent (X'6D').
- If the auto-dial unit is not equipped to use separator characters, a dialing-pause character can be used to allow time to receive a secondary dial tone. A dialing-pause character pauses for a predetermined interval before sending the next digit. You can insert as many vertical bars (|) or the hexadecimal equivalent (X'FA') as you need to denote dialing pauses.
- Some modems require end-of-number characters. For end-of-number characters, use
  - asterisks (*) or the hexadecimal equivalent (X'5C')
  - percent signs (%) or the hexadecimal equivalent (X'6C')
  - (@) character or the hexadecimal equivalent (X'7C')
For example, you could code DIALNO=8_5799*, where _ is the separator character and * is the end-of-number character.

- The maximum length of the operand is 242 characters, including vertical bars, the separator character, and the end-of-number character.

For more information on this operand, see the NCP, SSP, and EP Resource Definition Reference.

**NCP/Token-Ring interconnection (NTRI):**

Use the following format when coding DIALNO for a terminal that is called by a host through NCP/Token-Ring interconnection:

```
DIALNO=aabbcccccccccccc
```

*aa*  
The token-ring interface coupler (TIC) number of the communication controller (a decimal number in the range 00–99). The TIC number equates with the PORTADD keyword on the NCP LINE definition statement.

*bb or Xbb*  
The service access point (SAP) address of the terminal. The SAP address must be a multiple of 4, and can be a hexadecimal or decimal number.

*cccccccccccc*  
The last 6 bytes of the terminal’s ring-station address (the first digit must be in the range 4–7). The address must consist of 12 hexadecimal digits that range from X’4000 0000 0000’ to X’7FFF FFFF FFFF’ using the following rules:

- If bytes 0 and 1 are equal to either X’4000’ or X’7FFF’ then bytes 2 through 5 must be unique on the network.
- If bytes 0 and 1 are in the address range X’4001’ through X’7FFE’ then bytes 0 through 5 must be unique on the network.

**Note:** You cannot code 000000000000 for the last 12 digits of the DIALNO operand on the PATH definition statement.

This operand applies only if the physical unit is attached through NCP/Token-Ring interconnection. DIALNO is required if the physical unit is used as a dial-out physical unit.

**Frame Relay Devices:**

Use the following format when coding DIALNO for dialing out to a frame relay device:

```
DIALNO=aabb00000000cccc
```

or

```
DIALNO=aaXbb00000000cccc
```

*aa*  
The port number specified with the PORTADD keyword of the frame relay link, in decimal.

*bb or Xbb*  
The remote station’s service access point (SAP), in decimal (bb) or hexadecimal (Xbb).

*cccc*  
The data link connection identifier (DLCI) for the connection, in hexadecimal.
For frame relay BAN-attached devices, you must also specify the remote MAC address for the BAN station, in hexadecimal, as shown by `dddddddddddd` in the following format:

```
DIALNO=aabb00000000ccccdddddddddddd
```

or

```
DIALNO=aaXbb00000000ccccdddddddddddd
```

The MAC address must be in the range `1–X'7FFF FFFF FFFF'`.

**IBM 3172 Nways Interconnect Controller Switched Data Networks:**

Use the following DIALNO format for host-initiated connections to peripheral devices connected through the IBM 3172 Nways Interconnect Controller (`aa`, `bb`, and `cccccccccccc` are hexadecimal):

```
DIALNO=aabbcccccccccccc
```

- **aa** A 2 digit place-holder. It has no meaning in the IBM 3172 Nways Interconnect Controller environment.
- **bb** The service access point (SAP) address of the terminal. It must be a multiple of 4.
- **cccccccccccc** The medium access control (MAC) address for the peripheral device on the local area network.

**Note:**

1. You cannot code `000000000000` for the last 12 digits (MAC address) on the DIALNO operand on the PATH definition statement.
2. If the PU is attached to a LAN through the IBM 3172 Nways Interconnect Controller using an 8209 LAN bridge, the MAC address must be coded differently. Because the 8209 will flip the bits in every byte of the MACADDR, the MACADDR should be coded in inverted order in the DIALNO.
3. Certain levels of NCP do not support hexadecimal digits `X'A'–X'F' as part of the MAC address.
**DISCNT** values:

- **YES**
- **NF**
- **NO**

**DISCNT DELAY** values:

- **DELAY**
- **NF**
- **delay_time**
- **NO**

**Statements:**

GROUP, PU

**Dependencies:**

PUTYPE=2

Specifies when VTAM should terminate its SSCP-LU and SSCP-PU sessions and, when **DISCNT=**YES or **DISCNT=DELAY**, whether to indicate “final-use” status in the DACTPU request unit when it deactivates a PU.

**DISCNT=(YES)**

**DISCNT=(NO)**

Specifies whether VTAM physically disconnects the PU when the last LU ends its session with its application program (that is, when there are no more LU-LU sessions).

For a PU on a switched link, disconnection means that the dial connection is broken (in effect, the telephone is hung up, to save telephone charges) and the SSCP-PU session is terminated. Disconnection on a switched link does not involve deactivating the PU or its LUs (that is, sessions can be requested with those LUs causing the physical connection to be re-established).

See *z/OS Communications Server: SNA Network Implementation Guide* for more information about disconnection.

**DISCNT=(YES,F)**

**DISCNT=(YES,NF)**

Specifies whether VTAM indicates “final-use” status in the DACTPU request unit when it deactivates a physical unit as a result of **DISCNT=**YES. This value does not apply when **DISCNT=**NO, nor does it have any effect on the VARY INACT command.

If you code F or use the default, “final-use” status is indicated and the connection can be ended. If you code NF, “not-final-use” status is indicated and the connection should not be ended.

Each device has its own requirements regarding “final-use” status. To determine whether to code F or NF for a given device, consult the appropriate installation publication for the device.
**DISCNT=(DELAY,F,delay_time)**
**DISCNT=(DELAY,NF,delay_time)**
**DISCNT=(DELAY,,delay_time)**

Specifies that VTAM disconnects the physical unit if no LU-LU sessions exist after the value specified for delay_time expires. If delay_time is not specified, the current value for the DISCNTIM start option is used. delay_time specifies the number of seconds that VTAM delays disconnecting this PU if no LU-LU sessions exist. A decimal value in the range 5–65535 is valid.

The character value IMMED may be coded for delay_time to allow an immediate disconnect.

**Note:** A delay time value can also be assigned for this PU through the MODIFY DEFAULTS operator command.

**For HPR Sessions:** If this VTAM is a node in a High-Performance Routing (HPR) route and is connected to an IBM 37xx Communication Controller, an Open Systems Adapter or through an IP network using Enterprise Extender, NCP or VTAM monitors the session traffic on the link to determine whether to deactivate a PU. When DISCNT=(DELAY,,delay_time), NCP or VTAM disconnects the PU if no session traffic has occurred during the time interval specified by delay_time. When DISCNT=(DELAY) or DISCNT=(YES), NCP or VTAM disconnects the PU if no session traffic has occurred during the time interval specified on the DISCNTIM start option.

**DLCADDR**

```
DLCADDR=(subfield_id,,type,,string)
```

**statements:** PATH
**dependencies:** LINENM
**range:** See description

Specifies the signaling information used to initiate a connection with a physical unit over a switched link.

Multiple DLCADDR operands can be coded on a single PATH definition statement. For example, the following information can be coded on a single PATH statement.

```
PATH DLCADDR=(1,TYPE1,STRING1),
   DLCADDR=(2,TYPE2,STRING2)
```

**subfield_id**

Indicates the subfield. A value between 1–96 can be specified.

The first DLCADDR operand must have a subfield_id of 1. Also, each operand must have a unique subfield_id. If two DLCADDR operands—following a DLCADDR operand with a subfield of 1—have the same subfield_id, VTAM treats the two operands as duplicates. VTAM ignores the second operand and issues an error message. If a DLCADDR operand with a subfield of 1 immediately follows another DLCADDR operand with a subfield of 1, VTAM does not ignore the second operand. It treats the second operand as the beginning of another set of signaling information.
**Note:** If you receive an error message referring to the first DLCADDR operand coded on a PATH statement, limited syntax checking is done on the subsequent DLCADDR operands coded.

For DLUR PUs, refer to the documentation related to the DLUR implementer or to the device that supports the routing of the data link connection for information about what the specific values indicate. For ATM native connections, see [For ATM native connections](#) for information about what the specific values indicate.

**type**
Indicates the data format of the signaling information and is one of the following formats:

<table>
<thead>
<tr>
<th>Type</th>
<th>Data Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Compressed alphanumeric</td>
</tr>
<tr>
<td>BCD</td>
<td>Binary coded decimal</td>
</tr>
<tr>
<td>C</td>
<td>EBCDIC</td>
</tr>
<tr>
<td>D</td>
<td>Decimal</td>
</tr>
<tr>
<td>I</td>
<td>ASCII</td>
</tr>
<tr>
<td>X</td>
<td>Hexadecimal</td>
</tr>
</tbody>
</table>

Type C (EBCDIC) is the default.

**string**
Defines the signaling information in the data format defined by `type`. Multiple `string` values can be coded for a `subfield_id type` pair. For example:

```
DLCADDR=(1,TYPE1,STRING1,STRING2,STRING3)
```

You can specify up to 250 bytes of signaling information on a single DLCADDR operand. If coding multiple `strings` on a single operand, the total number of bytes available for signaling information is reduced by 2 for each additional DLCADDR `string`.

For DLUR PUs, `string` defines either the DLC type or the DLC signaling information in the data format defined by `type`.

Multiple sets of DLCADDR operands can be coded on the PATH definition statement. The first DLCADDR operand for each set of DLCADDR operands must have a `subfield_id` of 1 and it identifies the DLC type, which can be one of the following items:

- ATM
- Data link switching
- Ethernet
- FDDI
- Frame relay
- Internal PU
- SDLC leased
- SDLC switched
- Token-ring
- X.25
The remaining DLCADDR operands for each set of DLCADDR operands define the DLC signaling information for each DLC type. Refer to the data link control provider documentation (such as the NCP Resource Definition Reference) for information about how to code the remaining DLCADDR operands. The data link control provider is responsible for verifying the contents of the DLCADDR operand.

The maximum number of PATH statements in a switched major node that can have DLCADDR coded is 32767.

Note: DLCADDR is mutually exclusive with the LINENM operand for any single switched PATH definition statement.

Code DLCADDR, or DIALNO, or both for switched lines that can call out to a remote physical unit (that is, for links specified with CALL=OUT or CALL=INOUT on the LINE definition statement).

If both DLCADDR and DIALNO are coded on the PATH definition statement, both values are processed and stored. If the device that performs the routing of the data link connection supports expanded dial information, DLCADDR is used to make the connection. If the device does not support expanded dial information, DIALNO is used. Refer to the data link control provider documentation (such as the NCP Resource Definition Reference) for information about whether expanded dial information is supported, whether both DIALNO and DLCADDR are supported together, and how to code the DLCADDR contents. The data link control provider is responsible for verifying the contents of the DLCADDR operand.

For DLUR physical units:
Code DLCADDR for switched connections where the DLUS host can call out to a remote physical unit that is supported by a DLUR. The information that is coded on the DLCADDR operand is dictated by the DLUR being used. Refer to the DLUR implementer documentation (README file) or to the z/OS Communications Server: SNA Network Implementation Guide for information on how to code the DLCADDR contents. The DLUR implementer is responsible for verifying the contents of the DLCADDR operand.

For ATM native connections:
Code the following DLCADDR operands for an ATM native connection. Fields in italics are user-defined fields.

Note: You can code only one set of DLCADDR operands on the PATH definition statement for ATM native connections.

DLCADDR=(1,C,ATMSVC,tg_use)
DLCADDR=(7,BCD,format,best_effort_indicator,
               a,b,c,forward_traffic_management_options,
               d,e,f,backward_traffic_management_options)
DLCADDR=(8,X,format,outbound_QOS_class,inbound_QOS_class)
DLCADDR=(9,BCD,tms_identifier)
DLCADDR=(21,X,type,remote_ATM_address,remote_ATM_address)
DLCADDR=(31,D,remote_service_access_point_address)
DLCADDR=(51,X,format,a,b,c,d,e)
DLCADDR=(61,BCD,format,05,a,b)

The following list explains the information provided by each subfield:

<table>
<thead>
<tr>
<th>Subfield</th>
<th>Contents</th>
</tr>
</thead>
</table>
1 C indicates EBCDIC format. ATMSVC indicates that the connection provides native access to an ATM network.

Code tg_use to specify one of the following items:

**EXCLUSIVE**
- Indicates that the switched virtual channel (SVC) can be used only for this TG

**SHARE**
- Indicates that the SVC can be used for this TG and connection network TGs from this port to the same remote node through the same remote ATM address

7 BCD indicates binary coded decimal format.

Code format to specify one of the following items:

- 00 Indicates ITU-T defined
- 03 Indicates ATM network defined

Code best_effort_indicator to specify one of the following items:

- 00 Indicates that guaranteed bandwidth is wanted
- 01 Indicates that best effort is acceptable

Code a, b, c, d, e, and f as 4-byte decimal values to specify the following items:

- a Forward peak cell rate (cells per second) for cell loss priority 0+1 (CLP=0+1)
- b Forward sustainable cell rate (cells per second) for (CLP=0)
- c Forward maximum burst size (cells) for (CLP=0)
- d Backward peak cell rate (cells per second) for cell loss priority 0+1 (CLP=0+1)
- e Backward sustainable cell rate (cells per second) for (CLP=0)
- f Backward maximum burst size (cells) for (CLP=0)

Code forward traffic_management_options to specify one of the following items:

- 00 Indicates that tagging is not requested in the forward direction
- 01 Indicates that tagging is requested in the forward direction

Code backward traffic_management_options to specify one of the following items:

- 00 Indicates that tagging is not requested in the backward direction
- 01 Indicates that tagging is requested in the backward direction

**Note:** If best_effort_indicator is 00, code format, a, b, c, forward_traffic_management_options, d, e, f, and backward_traffic_management_options.

If best_effort_indicator is 01, code format, a, and d.

8 X indicates hexadecimal format.

Code format to specify one of the following items:
Indicates ITU-T defined

Indicates ATM network defined

Code `outbound_QOS_class` to specify the QoS class for outbound (forward) data traffic.

Code `inbound_QOS_class` to specify the QoS class for inbound (backward) data traffic.

9  BCD indicates binary coded decimal format.

Code `tns_identifier` to specify the carrier identification code for the carrier that will be used for this connection.

21  X indicates hexadecimal format.

Code `type` to specify one of the following items:

0101  Indicates public E164 address, which means the address is in a public ATM network. The address is variable in length.

0002  Indicates International Organization for Standardization (ISO) network service access point (NSAP), which means the address is in a private ATM network. The address is always 20 bytes long.

Code `remote_ATM_address` to specify the address of the remote node connected through this SVC.

**Note:** If the remote node is a VTAM node, see [zEnterprise System and System z10 OSA-Express Customer's Guide and Reference](#) for information about how to determine the ATM address through which the remote VTAM node can be reached. You can also determine the ATM address by starting both VTAMs, then activating the following resources in each VTAM:

- The TRL major node
- The XCA major node that defines the port through which VTAM can be reached
- Any line defined in the XCA major node

Then issue a DISPLAY ID command for the XCA major node in each VTAM. The resulting displays show the ATM addresses for each VTAM. These are the addresses that should be specified in the switched major nodes of the VTAMs.

Code a second `remote_ATM_address` if `type`=0002 and the port through which this call is made is attached directly to a public network. This second `remote_ATM_address` is a public gateway E164 address through which the private address can be reached.

51  X indicates hexadecimal format.

Code `format` to specify the following items:

00  Indicates ITU-T defined

Code `a`, `b`, `c`, `d`, and `e` to specify the following bearer capabilities:

- **a**  Bearer class.
a can be one of the following items:
01 Indicates class A
03 Indicates class C
10 Indicates class X

b Susceptibility to clipping.
b can be one of the following items:
00 Not susceptible
01 Susceptible

c User plane connection configuration.
c must be:
00 Point-to-point

d If bearer class is 10, indicates the traffic type.
d can be one of the following items:
00 No indication
02 Variable bit rate

e If bearer class is 10, indicates timing requirements.
e can be one of the following items:
00 No indication
01 End-to-end timing required

61 BCD indicates binary coded decimal format.

Code format to specify the following items:
00 Indicates ITU-T defined

Code 05 to indicate ATM adaptation layer (AAL) type 5.

Code a and b as 4-byte decimal values to specify the following AAL parameters:
a Forward CPCS-SDU maximum size
b Backward CPCS-SDU maximum size

Note: If an application using an ATM reserved bandwidth SVC connection sends RUs smaller than the forward CPCS-SDU size, the throughput for the session may be reduced. For example, if the forward CPCS-SDU size was defined or allowed to default to 8448 (Maximum RU size 8K + 256), and the data being transmitted over this ATM connection is in 1K RUs, the throughput (as defined by the connection’s forward cell rate) is greatly reduced. If this ATM SVC connection is used for 1K RU data transmission, the forward CPCS-SDU size should be set to 1280 bytes (Maximum RU size of 1K + 256).

DLOGMOD
**DLOGMOD** = **default_logon_mode_table_entry_name**

statements: GROUP, PU, LU
dependencies: PUTYPE=1 or 2

Specifies the name of the logon mode table entry used by default if one is not otherwise provided. If you do not code this operand and the name of a logon mode table entry is not otherwise provided, VTAM uses the first entry in the applicable logon mode table (specified on the MODETAB operand or used by default).

If you specify MODETAB, the entry must be in either the specified table or in ISTINCLM, an IBM-supplied default logon mode table. If you do not specify MODETAB, the entry must be in ISTINCLM. For more information on logon mode entries see "Logon mode table" on page 1026.

**DLURNAME**

**DLURNAME** = **dlur_name**

statements: PATH
dependencies: DLCADDR

Specifies the name of the dependent LU requester (DLUR) for this PU. You can code a network-qualified name for dlur_name.

If you do not code NETID, the network ID for the DLUR defaults to the network ID of the dependent LU server (DLUS). Note that NETID specifies a real network ID and that an alias network ID is not allowed for the DLUR.

The data link control (DLC) information specified on the DLCADDR operand is passed to the DLUR specified.

**Note:** If DLURNAME is specified, do not code the following operands:
- CALL
- GID
- GRPNM
- REDIAL
- SHMTIM
- VERID
- VERIFY

Note that the redial function is supported automatically for DLUR PUs, and the REDIAL operand is not involved. However, the following restrictions apply to the redial function for DLUR PUs:
- Redial applies only to PUs activated by dependent LU servers.
- Redial is attempted as many times as there are PATH definition statements for a PU.
• Redial is attempted for any sense code except 082C0002 (PU is already active) or 083B0002 (PCID is not unique). For 083B0002, VTAM deletes the older PCID and then resends the activation request with a new PCID.

• Redial is attempted for every active or pending active PU served by a DLUR when a protocol violation, topology database update error, or session outage signal is received for that DLUR.

**DWACT**

![Diagram of DWACT]

*statements:*

PU

*dependencies:*

none

Specifies whether a physical unit is dialed when it is activated. DWACT provides an alternative to using the VARY DIAL command on each physical unit after it is activated.

For more information on the use of DWACT with a switched physical PU which has an associated logical hierarchy, see the [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/en/SSLVMB_2.2.0/com.ibm.zos.v2r2.snah.00/zoscommnsr/sna_guide.html)

**DWACT=NO**

Specifies that the physical unit is not dialed after it is activated.

**DWACT=YES**

Specifies that the physical unit is dialed after it is activated.

*Note:* DWACT=YES has no effect on VARY INACT,R command processing; VARY INACT,R causes the specified resource to be deactivated and reactivated but no dial operation will be performed.

**DWINOP**
Specifies whether a switched physical unit used for an Enterprise Extender connection is redialed when the connection is broken by INOP. The values specified by the REDIAL and REDDELAY operands on the PATH statement determine if any additional redial attempts are made and the interval between each retry. Coding DWINOP=YES can reduce the need for operator intervention in reestablishing Enterprise Extender connections.

**DWINOP=NO**
Specifies that the physical unit is not redialed after it has INOPed.

**DWINOP=YES**
Specifies that the physical unit is redialed after it has INOPed.

**Note:** Specification of DWINOP=YES will cause this host to redial the partner when the connection has been broken because of a VARY HANGUP command on the remote host. If the VARY HANGUP command successfully placed the switched PU on the remote host in connectable state, then the dial-in from this host will be accepted and the connection will be reestablished. Issuing VARY INACT for the PU on the remote host will prevent the connection from being made unless VARY ACT is issued for the PU on the remote host before either all dial retries are exhausted on this host or VARY INACT is issued for the switched PU on this host.

In addition, coding CALL=OUT on the Enterprise Extender PATH statement under the PU on the remote host, or ANSWER=OFF on the GROUP statement in the XCA Major node on the remote host to which the PATH statement applies, will prevent inbound dial attempts from being successfully directed to that PU.

**DYNADJCP**
Notes:

1  DYNADJCP=YES is the only valid value if the TRLE operand is coded.

Statements:
GROUP, PU

Dependencies:
APPN only; ignored if CPNAME coded

Specifies whether an adjacent control point can be created dynamically for the adjacent node.

**DYNADJCP=NO**
Indicates that an adjacent control point cannot be dynamically created for the adjacent node.

**DYNADJCP=YES**
Indicates that an adjacent control point can be dynamically created for the adjacent node.

**Note:** A type 2.1 peripheral node in the switched major node requires both IDBLK and IDNUM, or CPNAME. DYNADJCP=YES (the operand or the start option) is valid when you code IDBLK and IDNUM. However, DYNADJCP is ignored when you code CPNAME.

**DYNLU**
Specifies whether dynamic allocation of CDRSC definitions is supported. If you do not code this operand, the default is the value you coded on the DYNLU start option.

If dynamic allocation of CDRSC definitions is not supported for a particular PU, a resource that was dynamically defined as a CDRSC cannot use that PU to connect to a session.

This operand is valid only when PUTYPE=2, and it does not apply to SSCP-SSCP CDRSCs.

**DYNLU=NO**

Specifies that dynamic allocation of CDRSC definitions is not supported. If you code DYNLU=NO and plan to connect to other nodes using CP-CP sessions, then predefine each node in the CDRSC major node. If you do not, CP-CP session setup fails.

**DYNLU=YES**

Specifies that dynamic allocation of CDRSC definitions is supported.

For subarea nodes (HOSTSA is specified), DYNLU=YES is meaningful only when the host CDRM is activated and CDRDYN=YES is specified on the CDRM definition statement.

For APPN-only nodes (HOSTSA is not specified), DYNLU=YES is meaningful even though a host CDRM cannot be activated.

**Note:** If you have coded a DYNLU value in the adjacent control point major node for the adjacent control point to which this physical unit is linked, that value overrides the DYNLU value you code on this PU definition statement. For additional DYNLU considerations, see "DYNLU" on page 23.

**EEVERIFY**
Specifies whether VTAM should automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the EE connection. This option might also specify the time interval during which VTAM should send the probe on active EE connections.

This operand is valid only for the EE connection attached to a predefined switched PU. If you do not code this operand, VTAM uses the value coded on the EEVERIFY start option.

**EEVERIFY=ACTIVATE**

Specifies that VTAM should automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the EE connection. If one or more ports are not reachable or are blocked, then VTAM issues an error message and does not activate the EE connection.

**EEVERIFY=NEVER**

Specifies that VTAM should never automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the EE connection.

**EEVERIFY=Number_of_minutes**

Specifies that VTAM should automatically send an LDLC probe during the activation of the EE connection and that VTAM should continue to periodically send LDLC probes at intervals specified by the `Number_of_minutes` value while the connection is active. Valid values for `Number_of_minutes` are in the range 1 - 1440 minutes (1440 minutes specifies 1 day). If one or more ports are not reachable or are blocked at activation, then VTAM issues an error message and does not activate the EE connection. If one or more ports are not reachable after the connection is active, VTAM issues an error message, but the connection remains active.

For EE connection networks, define this parameter on the connection network GROUP definition statements in the EE XCA major node. For dial-in EE connections that have their associated PUs dynamically created, define this parameter on the model major node (DYNTYPE=EE) PU definition statement.

**ENC**
ENCRTYPE

Indicates whether this logical unit has any special requirements for enciphering and deciphering messages.

ENC=NONE
  Specifies that this logical unit cannot engage in cryptographic sessions.

ENC=OPT
  Specifies that this logical unit can engage in cryptographic sessions, but allows the application program to determine whether to use cryptography.

ENC REQD
  Specifies that VTAM must encipher all messages to and from this logical unit. If ENCR=REQD, no sessions can be established with this logical unit unless the host with which it is associated can handle cryptographic sessions.

ENC=SEL
  Has no meaning for logical units. If you specify ENCR=SEL, ENCR=OPT is used instead.

ENCRTYPE
ENCRTYPE = DES

Specifies that VTAM must use a minimum of DES encryption with an 8-byte key when performing session level encryption. This is the default.

ENCRTYPE = TDES24

Specifies that VTAM must use a minimum of Triple_DES encryption with a 24-byte key when performing session level encryption.

Note: The ENCRTYPE value coded on the GROUP or PU will be sifted down to the appropriate resource (LU).

The SYSDEFed ENCRTYPE of a LU can be changed at runtime through the MODIFY SECURITY command. See z/OS Communications Server: SNA Operation for specific details. However, the ENCRTYPE value will not be able to be downgraded. In other words, if the application is currently operating at ENCRTYPE=TDES24, then the encryption level cannot be changed to ENCRTYPE=DES using the MODIFY SECURITY command.

Note: If downgrading the security level is preferred, the current major node being used must be deactivated and another major node value must be coded.

In addition, a user may use the LOGMODE TABLE to override the ENCRTYPE value of the SLU based on the logmode entry being used. The ENCRTYPE value will not be able to be downgraded from TDES24 to DES using the logmode table. Only TDES24 can be specified for the encryption level (ENCRTYP). See “MODEENT macroinstruction” on page 1027 for more information about MODEENT macroinstructions.

FASTPASS
Determines how session establishment is performed for application programs that issue the CLSDST macroinstruction with the PASS option as part of their session establishment procedure. The application program acts as a secondary logical unit (SLU).

The PLU acknowledges the capability of the SLU during session establishment. If you have coded the logon performance enhancement in the PLU’s application program, only a single bind is issued to establish the session.

Some SLUs are incompatible with this enhancement. If your SLU is incompatible with this enhancement, code FASTPASS=NO.

**Note:** FASTPASS applies only to application programs (acting as SLUs) that establish sessions with TSO.

**FASTPASS=NO**

Specifies that an unenhanced logon procedure requiring multiple binds is performed for application programs that issue the CLSDST PASS macroinstruction with the PASS option as part of their session establishment procedure.

**FASTPASS=YES**

Specifies that the logon performance enhancement is performed for application programs that issue the CLSDST macroinstruction with the PASS option as part of their session establishment procedure.
Contains certain device information for virtual logical units that are supported by network terminal option (NTO).

**FEATUR2=EDATS**  
**FEATUR2=NOEDATS**  
Specifies whether this terminal has the extended data stream feature. You cannot use this operand for terminals attached by SDLC lines.

**FEATUR2=DUALCSE**  
**FEATUR2=LOWERCSE**  
Specifies how VTAM sends alphabetical characters coded with the TEXT operand on a USSMSG macroinstruction to a non-SNA terminal over the SSCP-LU session. This value does not affect non-alphabetical characters, or any characters coded on the BUFFER operand of a USSMSG macroinstruction.

Code LOWERCSE to indicate that alphabetical characters are sent to the terminal over the SSCP-LU session in lowercase. Code DUALCSE to indicate that VTAM sends all characters as they are coded in the USSMSG macroinstruction.

**FEATUR2=MODEL1**  
**FEATUR2=MODEL2**  
Identifies the specific model number (Model 1 or 2) for this 3275, 3277, 3284, or 3286 component. Code MODEL1 for those devices that have a default screen or buffer size of 480 bytes. Code MODEL2 for those devices that have a default screen or buffer size of 1920 bytes.

This information is available to an application program as part of the device characteristics pertaining to this terminal. You can obtain those characteristics by using the INQUIRE macroinstruction. For more information on using the INQUIRE macroinstruction, see [z/OS Communications Server: SNA Programming](https://www.ibm.com/support/knowledgecenter/S590000/SM630000/)

**FEATUR2=NOPRINTR**  
**FEATUR2=PRINTR**  
Specifies whether this terminal has an attached IBM 3284 Model 3 printer. This operand is valid only if TERM=3275.

**FEATUR2=NOSELPEN**  
**FEATUR2=SELPEN**  
Specifies whether this terminal supports a selector pen.
**GID** = *group_identifier*

**statements:** PATH  
**dependencies:** none  
**range:** 1–255

Specifies the group identifier of a group of paths across all physical units in the switched major node.

Group identifiers can be assigned to allow the VTAM operator to regulate the use of switched network services. For example, if GID=6 is assigned to all paths in a switched major node that use direct distance dialing, the VTAM operator can make all of the paths usable or not usable with a single command.

**GRPNM**

---

**GRPNM** = *group_name*

**statements:** PATH  
**dependencies:** CALL; VERID; VERIFY

Provides the name of a GROUP definition statement in the NCP or channel-attachment major node that defines a group of SDLC switched links. The line group must have all the characteristics necessary to process the telephone number and must be compatible with the type of physical unit.

GRPNM is required when one of the following items is specified:
- VERIFY=IN, OUT, or INOUT
- VERID
- CALL

A dial-out connection will be made to the PU associated with this PATH definition statement.

For predefined Enterprise Extender connections, coding a PATH definition statement with GRPNM allows an incoming call to prefer one switched PU over another. When the GROUP name, associated with the line selected for the inbound call, matches the GRPNM on the switched PATH, the switched PU associated with the PATH is preferred over another switched PU that does not match, or does not have a PATH definition statement coded.

**Note:** When defining an NCP/Token-Ring Interconnection (NTRI), an IBM 3172 Nways Interconnect Controller, or an Open Systems Adapter group, *group_name* should be the name of the logical group definition.

**HOSTNAME**
Specifies the host name used by the TCP/IP stack on the remote end of the Enterprise Extender connection. The HOSTNAME you code may be qualified with its domain name. The local host must be able to perform TCP/IP name-to-address resolution, either by connectivity to a DNS name server, or by using a local host file such as HOSTS.SITEINFO or /etc/ipnodes. For more information see z/OS Communications Server: IP Configuration Guide. Furthermore, if the host name represents a z/OS Communications Server APPN node (or NAT address associated with a firewall), the address returned must be a static VIPA address.

Note:
1. Remote_host_name may be up to 64 characters in length.
2. HOSTNAME and IPADDR are mutually exclusive; however, IPADDR is used if both are specified.
3. HPR must equal YES if the PU is used for Enterprise Extender.

HPR
**HPR**

Specifies whether High-Performance Routing (HPR) support is enabled for a PU.

- **HPR=YES**
  - Indicates that this PU provides HPR support.

- **HPR=NO**
  - Indicates that this PU does not provide HPR support.

**HPREEARB**

This operand specifies the algorithm to calculate adaptive-rate-based (ARB) pacing for the High-Performance Routing (HPR) pipes that traverse predefined Enterprise Extender (EE) connections.

- **HPREEARB=PROGRESS**
  - Specifies that the progressive-mode ARB flow control algorithm is used for the HPR pipes that traverse this predefined EE connection. Progressive mode is used only when the HPR pipe is a single-hop pipe (which includes a single physical hop across a 2-hop EE virtual routing node [VRN]) and the RTP partner supports progressive mode.
  - For EE connection networks, define this parameter on the connection network GROUP definition statements in the EE XCA major node. For dial-in EE connections associated with dynamically created PUs, define this parameter on the model major node (DYNTYPE=EE) PU definition statement.

- **HPREEARB=HPRARB**
  - Specifies that the HPRARB start option value is used to determine the ARB flow control algorithm for the HPR pipes that traverse this predefined EE connection.
  - **Tip:** The start option value of HPRARB=BASE is not recommended for use with HPR pipes that traverse EE connections.

---

Chapter 2. Major nodes 609
Specifies the time that lapses before RTP pipes enter the path switch state. The ARB flow control algorithm is sensitive to minor variations in the round-trip time and to unpredictable response times from the partner. This sensitivity might cause RTP endpoints to prematurely enter the path switch state. Although this situation does not affect availability, it does consume CPU cycles and it also causes a significant number of path switch messages to be written to the console log.

This operand is valid only for RTP pipes directly attached to a predefined EE connection. It affects only path switches that are the result of an unresponsive partner (path switches that generate the message IST1818I PATH SWITCH REASON: SHORT REQUEST RETRY LIMIT EXHAUSTED). The HPRPSDLY operand does not control path switches that are initiated as a result of a TG INOP, F RTP commands, or the PSRETRY function. If you do not code this operand, the default is the value that you coded on the HPRPSDLY start option.

**HPRPSDLY=ps_delay**

Specifies the number of seconds that RTP pipes must delay prior to entering the path switch state. During this time, the RTP endpoint periodically tries to contact the partner in an effort to avoid switching paths. The value 0 indicates that the RTP nodes enter the path switch state when a predetermined number of retry attempts have been unsuccessful.

**HPRPSDLY=EEDELAY**

Specifies that VTAM calculates the number of seconds that RTP pipes must delay prior to entering the path switch state. The value that is calculated allows enough time for the EE keep-alive mechanism to cause the EE connection to become inoperative if connectivity is lost to the partner. Unnecessary path switches are avoided while EE determines whether there is a loss of connectivity to the partner.

For EE connection networks, define this parameter on the connection network GROUP definition statements in the EE XCA major node. For dial-in EE connections with associated PUs that are dynamically created, define this parameter on the model major node (DYNTYPE=EE) PU definition statement.

**Result:** When the EEDELAY value is specified, the value calculated for the HPR path-switch delay might be as long as 253 seconds.

**Tips:**
- This operand can be useful when the RTP partner is on a node that is CPU-constrained or running in a virtualized environment. In both of these situations, allowing additional time for the RTP partner to respond might avoid unnecessary processing associated with a path switch.
- If alternate routes exist, specifying a long delay time might cause unnecessary delays for the sessions that are using this RTP pipe.
**IDBLK**

Specifies a 3-digit hexadecimal number that identifies the device type. You can get the number from the component description manual for the device. All devices of the same type, such as all 3274s, will have the same IDBLK.

Valid values are X'001'–X'FFE'. Note that if the value is X'nnn', you code IDBLK=nnn. The IDBLK number occupies bits 16–27 of the 48-bit station ID constructed by VTAM for switched network operation. See the IDNUM operand for a description of the station ID.

You should not code IDBLK for switched subarea connections (type 4 and 5 physical units).

For any switched peripheral node (PUTYPE=1 or 2):
- If the node is not a type 2.1 peripheral node, you must code both IDBLK and IDNUM.
- For a type 2.1 peripheral node, coding CPNAME is required if the peripheral node supports role negotiation (such as 3172, 3174, and Communications Manager).
  
  For external communication adapter type 2.1 peripheral nodes, you must code CPNAME. IDBLK and IDNUM are ignored if you code them.
  
  For an APPN connection, if you code IDBLK and IDNUM, code DYNADJCP=YES instead of CPNAME.
- If VTAM is defined as a type 2.1 node, you must code CPNAME.
- See the appropriate product manual for CPNAME and IDNUM and IDBLK values.

**IDNUM**
SM590000

IDNUM = id_number

**statements:**

PU

dependencies:

CPNAME; IDBLK; PUTYPE=4,5

Specifies a 5-digit hexadecimal number that identifies the specific device (if PUTYPE=1 or 2) or the specific connection (if PUTYPE=4 or 5). The IDNUM for the device is usually the serial number of the device. The device’s serial number is frequently used for this purpose. You can get this number from the component description manual for the device.

For any switched peripheral node (PUTYPE=1 or 2):

- If the node is not a type 2.1 peripheral node, you must code both IDBLK and IDNUM.
- For a type 2.1 peripheral node, coding CPNAME is required if the peripheral node supports role negotiation (such as 3172, 3174, and Communications Manager). For external communication adapter type 2.1 peripheral nodes, you must code CPNAME. IDBLK and IDNUM are ignored if you code them. For an APPN connection, if you code IDBLK and IDNUM, code DYNADJCP=YES instead of CPNAME.
- If VTAM is defined as a type 2.1 node, code CPNAME.
- See the appropriate product manual for CPNAME and IDBLK and IDNUM values.

For peripheral physical units, VTAM uses IDBLK and IDNUM to build a 48-bit station ID that is used in XID exchange during the dial procedure. That station ID must be unique for each station within the network, not only within the major node.

The station ID is structured as follows:

<table>
<thead>
<tr>
<th>Bits</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0–3</td>
<td>Reserved</td>
</tr>
<tr>
<td>4–7</td>
<td>PUTYPE</td>
</tr>
<tr>
<td>8–15</td>
<td>X'00'</td>
</tr>
<tr>
<td>16–27</td>
<td>IDBLK</td>
</tr>
<tr>
<td>28–47</td>
<td>IDNUM</td>
</tr>
</tbody>
</table>

For subarea physical units (PUTYPE=4 or 5), code either IDNUM or CPNAME. VTAM uses the IDNUM value to correlate the PU definitions between the two SSCP's that own each end of the switched line. This IDNUM value must be unique for all PU definitions within this SSCP that contain the same subarea number and NETID.

**Note:** Specify the same IDNUM on the PU definitions in the two SSCP’s that own each end of the switched lines.

The node identifier field of an XID contains IDNUM along with a value of X’FFF’ as the IDBLK.
For subarea physical units, VTAM uses the subarea number specified on SUBAREA and the IDNUM value to build a station ID that is used in some operator messages. That station ID must be unique for each station within the network, not only within the major node.

The station ID is structured as follows:

<table>
<thead>
<tr>
<th>Bits</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0–15</td>
<td>Subarea number</td>
</tr>
<tr>
<td>16–27</td>
<td>X’FFF’</td>
</tr>
<tr>
<td>28–47</td>
<td>IDNUM</td>
</tr>
</tbody>
</table>

**INCLUD0E**

Specifies whether VTAM provides LU names to the logical units defined under this PU. The names are provided by passing a CV0E, Network Name control vector on the ACTLU requests.

- **INCLUD0E=NO**
  - Indicates that VTAM will not provide LU names to the logical units defined under this PU.

- **INCLUD0E=YES**
  - Indicates that VTAM will provide LU names to the logical units defined under this PU.

**INCLUDSL**
Specifies whether VTAM includes the Network Services (NS) Secondary LU Name Field on a non-negotiable, non-extended (control vectors are not included) BIND, destined to a dependent LU under this switched XCA PU.

**INCLUDSL=**YES

Indicates that VTAM will provide the Secondary LU Name Field.

**INCLUDSL=**NO

Indicates that VTAM will not provide the Secondary LU Name Field.

**IPADDR**

Specifies the remote IP address for the connection.

**Note:**
1. The address must be specified in dotted decimal format for an IPv4 address (for example, IPADDR=4.6.41.7) or in colon-hexadecimal format for an IPv6 address (for example, IPADDR=2000:67:1:2).
2. HOSTNAME and IPADDR are mutually exclusive. However, IPADDR is used if both are specified.

**IPRESOLV**
IPRESOLV = (0)

Specifies the number of seconds VTAM waits for name-to-address resolution requests to complete before canceling the request. The default is 0 seconds, meaning VTAM will wait indefinitely for resolution to be performed. Any value from 0 to 255 seconds can be specified.

If a nonzero value is specified for IPRESOLV, the value should be chosen after giving consideration to the values defined for resolver timeouts on the TCP/IP stack (RESOLVERTIMEOUT directive in the TCPIP.DATA configuration file), the number of retries attempted (RESOLVERUDPRETRIES directive in the TCPIP.DATA configuration file), and the number of domain names that might be appended to the host name while performing name-to-address resolution (SEARCH or DOMAINORIGIN directive in the TCPIP.DATA configuration file). If a large timeout is specified for RESOLVERTIMEOUT, a number of retries is specified for RESOLVERUDPRETRIES, and/or a large number of domains is specified on SEARCH or DOMAINORIGIN, the resolver might take some time to complete the name-to-address resolution attempt, so the value of IPRESOLV should be set large enough to allow the resolver to complete its processing. See z/OS Communications Server: IP Configuration Reference for details on the resolver directives.

Alternatively, you may want to ensure that the host name being resolved (the HOSTNAME operand on the GROUP statement, or the HOSTNAME start option value) is defined in the local hosts files on this system, and then specify LOOKUP LOCAL DNS in the TCPIP.DATA configuration file to ensure that local hosts files are searched first (See z/OS Communications Server: IP Configuration Reference for details about the LOOKUP resolver directive). Coding LOOKUP LOCAL DNS will affect all resolver searches, not just Enterprise Extender searches initiated by VTAM, so this option should be chosen with caution.

Note:

1. The IPRESOLV keyword has been set to a default value of zero because in most cases it is desirable to establish the amount of time allowed before hostname resolution timeout by using the value in the TCP/IP configuration statements. Reset this value only when a VTAM hostname resolution time limit is required.
2. The RESOLVERUDPRETRIES option is applicable only when the resolver is communicating with the name server using the UDP protocol. (This is the default communication method and is specified with the RESOLVEVIA TCPIP.DATA statement.)
Specifies whether the boundary NCP (the NCP to which the switched physical unit becomes connected) retries a polling operation immediately for the device if an idle detect timeout follows a polling operation. For more information on this operand, see the NCP, SSP, and EP Resource Definition Reference.

**IRETRY=NO**
- Specifies that the boundary NCP does not retry the polling operation.

**IRETRY=YES**
- Specifies that the boundary NCP retries the polling operation.

**ISTATUS**

Determines whether the resource is to be activated after the first activation of the switched major node.

**ISTATUS=ACTIVE**
- Indicates that the resource becomes active when its major node becomes active.

**ISTATUS=INACTIVE**
- Indicates the resource is to remain inactive until the operator activates it. The operator can override this value by specifying the SCOPE operand on the VARY ACT command when the major node is activated. See z/OS Communications Server: SNA Operation for information about using the VARY ACT command.

**LANTEST**
LANTEST = LOCAL

LANTEST = GLOBAL

Restriction: If you specify MEDIUM=CSMACD (Ethernet) on the associated XCA PORT definition statement, the parameter has no meaning and is ignored.

This parameter allows the user to define how the XCA peripheral nodes connected on the token ring will be located. There are two types of TEST commands sent: LOCAL and GLOBAL. The LOCAL will flow only on the local token ring. The GLOBAL will flow with a routing information field. This routing information field has the broadcast bit on, meaning to route the test into the entire network.

Note: The LANTEST parameter under the switched PU definition has meaning only for switched PUs connecting in through an XCA Major Node.

LANTEST=LOCAL
- Specifies that a TEST frame will first be sent out on the LOCAL token ring. If no response is received within the PORT TIMER interval, a GLOBAL TEST frame will be sent. Each time the GLOBAL TEST receives no response within the PORT TIMER interval, another GLOBAL TEST will be sent. The GLOBAL test will be repeated up to four times, at which time the connection setup is terminated if no response is received.

LANTEST=GLOBAL
- Specifies that only GLOBAL TEST frames will be used to locate the device. Each time the GLOBAL TEST receives no response within the PORT TIMER interval, another GLOBAL TEST will be sent. The GLOBAL test will be repeated up to five times, at which time the connection setup is terminated if no response is received.

LIMRES
Notes:

1  LIMRES=NO is the only valid value if the TRLE operand is coded.

*statements:*  GROUP, PU
*dependencies:*  LIMQSINT; LU 6.2 only; PUTYPE=1 or 2; Switched dynamics

Specifies whether the physical unit is to be treated as a limited resource. A limited resource allows you to limit the use of some network connections. It also causes any sessions that traverse the resource to be deactivated if no conversation is active. A session can also be deactivated if the time limit you specify for the LIMQSINT operand on the APPL definition statement expires.

This operand applies only to LU 6.2 conversations and PU types 1 and 2.

If you do not code a value for LIMRES, resources are not considered as limited resources. Furthermore, sessions will remain active even after all conversations have ended.

You can also use LIMRES in conjunction with the LIMINTCP start option to specify the interval to retain a free session using a limited resource for some network management flows (excluding flows using the APPN control point sessions or dependent LU server sessions). If you do not code a value for the LIMINTCP start option, resources are not considered to be limited resources, and the sessions will remain active even after all conversations have ended.

**LIMRES=NO**

- Specifies that this physical unit is not to be treated as a limited resource.

**LIMRES=YES**

- Specifies that this physical unit is to be treated as a limited resource.

**LINENM**
Names a line for the direct call function.

LINENM is mutually exclusive with the DIALNO and DLCADDR operands for any single switched path definition statement.

Although not required, the PATH definition statement for a direct call line should be the first statement after the switched PU definition statement. This ensures that VTAM attempts to call the switched physical unit using the direct call line first.

Note: The direct call function is not available for switched subarea connections.

**LLERP**

Specifies the link-level error recovery procedures (LLERP) preference for HPR connections. The LLERP value will be overridden to NOTPREF for PUs in a switched major node that will be mapped to LSA (XCA major node).

**LLERP=REQUIRED**

Specifies that link-level error recovery procedures are required.

**LLERP=NOTPREF**

Specifies that link-level error recovery procedures are required by this PU only when the adjacent link station requires it.
LOCADDR — address

statements: LU
dependencies: none
range: 0–255 (PU type 2) 0–63 (PU type 1)

Specifies the logical unit’s local address at the physical unit. For dependent logical units, the range of valid local addresses depends on the PUTYPE of the associated physical unit. All dependent logical units under a given physical unit must have unique LOCADDR values.

An LU definition statement is not required for every possible local address, and LOCADDR values need not be consecutive. Unused local addresses smaller than the largest local address at a station are not assigned network resources.

Note:
1. The value of LOCADDR depends upon the requirements of the device being defined. Consult the appropriate component description manual for these restrictions.
2. Although you can use this operand to define independent LUs, it is recommended that you either use dynamic definition or predefine the independent LUs as CDRSCs.

LOGAPPL

LOGAPPL — controlling_primary_lu

statements: GROUP, PU, LU
dependencies: PUTYPE=1 or 2

Identifies the name of the primary logical unit that this logical unit will automatically log on to as a secondary logical unit when this logical unit is activated. The name you specify can also be a USERVAR that will be resolved into its current value before being used to establish or redrive the session with the controlling primary logical unit.

For independent logical units, the LOGAPPL operand establishes a session with the primary logical unit designated on the LOGAPPL operand if a session does not currently exist with this logical unit. The session is established regardless of whether sessions with other logical units exist. The LOGAPPL operand cannot contain the name of an independent logical unit because independent logical units cannot be the controlling logical units for an automatic logon session.

To ensure retry of a pending automatic logon, code the AUTORTRY or AUTOTI start option.

You can specify a network-qualified name for LOGAPPL. You cannot specify an LUALIAS name for LOGAPPL.

Note:
1. If an error occurs for the switched connection (an INOP), VTAM reallocates the LOGAPPL session and attempts to redial the logical unit to recover the connection. The original session is ended by disconnecting the dial connection instead of ending the session normally using LOGOFF and TERMSELF. If you do not code the LOGAPPL operand, the PU and LU must be activated manually.

2. If you specify the NetView or TSO program as the application program on the LOGAPPL operand, you will get unpredictable results in recovery situations.

For more information about establishing sessions, see z/OS Communications Server: SNA Network Implementation Guide.

**LOGTAB**

```
LOGTAB --interpret_table_name
```

*statements:* GROUP, PU, LU
*dependencies:* PUTYPE=1 or 2

Specifies the name of an interpret table that VTAM uses when processing logon requests originating from the logical unit. VTAM uses the interpret table to interpret the name received from an initiation request. See "Interpret table" on page 1017 for a description of how to define an interpret table.

LOGTAB is valid for dependent logical units only.

**LUGROUP**
LUGROUP

Specifies the name of the model LU group that VTAM uses to select a model LU
definition when dynamically defining an LU. This operand also indicates that the
device supports dynamic definition of LUs.

LUSEED

Provides a 1–8 character pattern name that the SDDLU exit routine uses to
generate an LU name when dynamically defining an LU.

If you use the IBM-supplied SDDLU exit routine, this operand is required and
must have the following format:

• The first character must be either alphabetical or one of the national characters
  (@ or $).
• The remaining characters can be either alphabetical, numerical, or national.
• There must be either two or three contiguous national # characters in the name.
  These characters represent holders, in which the logical unit’s local address is
  substituted when an LU’s name is generated. If you specify three # characters,
  the LU’s logical address is substituted in decimal for the # characters. If you
  specify two # characters, the LU’s logical address is substituted in hexadecimal
  for the # characters.

When used with a user-written SDDLU exit routine, this operand is optional and
there are no restrictions on its format. For more information on the SDDLU exit,
see z/OS Communications Server: SNA Customization.

This operand is valid only when you code the LUGROUP operand.

MAXDATA

For PU type 1:

<table>
<thead>
<tr>
<th>MAXDATA = 261</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAXDATA = number_of_bytes</td>
</tr>
</tbody>
</table>
For PU type 1:

| statements: | GROUP, PU |
| dependency: | see notes |
| range: | 20-65535 |

For PU type 2:

| statements: | PU |
| dependency: | see notes |
| range: | 20-65535 |

Specifies the maximum number of bytes that the physical unit can receive in one PIU or PIU segment, including the transmission header (TH) and the request or response header (RH). To determine the maximum PIU (or PIU segment) size that the physical unit can receive, consult the component publications for the specific type of SDLC station represented by this PU definition statement.

The maximum amount of user data that VTAM sends to the physical unit in one PIU is:
- MAXDATA minus 2 bytes (for a PU type 1)
- MAXDATA minus 6 bytes (for a PU type 2)

**Note:** The RU part of the PIU contains the user data. When VTAM segments the PIU, the middle and last PIUs will contain three more bytes of user data in the RU because there is no RH field. The value for the PIU varies with the device; for example, the 3276 requires MAXDATA=262.

You must specify a MAXDATA value for a type 4 or 5 physical unit; there is no default for these physical units.

**Note:**
1. In a local area network, the value for the MAXDATA operand on the PU definition statement must be less than or equal to the value for the MAXDATA operand on the PORT definition statement for the LAN major node to which the switched physical unit is connected.
2. NCP V4R3 changed the MAXDATA calculation for a type 1 physical unit. See *NCP, SSP, and EP Resource Definition Reference* for information about calculating a MAXDATA value for physical units on non-switched links.
3. MAXDATA is ignored for type 2.1 nodes attached through an NCP, an IBM 3172 Nways Interconnect Controller, or through an IP network for Enterprise Extender. The type 2.1 node supplies the actual value used when the connection is established.

**MAXOUT**
MAXOUT

- \texttt{MAXOUT}---1
- \texttt{MAXOUT}---\textit{max\_number\_of\_pius}

\begin{itemize}
  \item \textbf{statements:} \textit{GROUP, PU}
  \item \textbf{dependencies:} ignored for type 2.1 if attached through an IBM 3172 Nways Interconnect Controller
  \item \textbf{range:} 8–127
\end{itemize}

Specifies the maximum number of PIUs or PIU segments (if the session parameters allow segmenting of data) that VTAM sends to the physical unit represented by this definition statement before requesting a response.

For performance reasons, you should code MAXOUT=7 if the physical unit represented by this definition statement can handle more than one PIU before sending out a link-level response. To determine this, consult the component publications for the specific type of SDLC station.

\textbf{Note:} MAXOUT is ignored for type 2.1 nodes attached through an IBM 3172 Nways Interconnect Controller or through an IP network for Enterprise Extender.

MAXSESS

\begin{itemize}
  \item \texttt{MAXSESS}---\textit{max\_number\_LU-LU\_sessions}
\end{itemize}

\begin{itemize}
  \item \textbf{statements:} \textit{GROUP, PU, LU}
  \item \textbf{dependencies:} independent LUs attached to an NCP
  \item \textbf{range:} 1–65535
\end{itemize}

Specifies the maximum number of concurrent LU-LU sessions in which an independent LU can participate. The MAXSESS value applies to each connection the LU is using to establish LU-LU sessions. For example, if MAXSESS=2 was specified for LUA and LUA appears on three PUs, LUA can have two sessions on each of the three PUs for a total of six sessions.

If you do not code MAXSESS, or code a value outside the range, the MAXSESS value set on the BUILD definition statement (NCP major node) will be used. Reference the \textit{NCP, SSP, and EP Resource Definition Reference} for more information about this operand.
**MDLENT**

```plaintext
MDLENT = model_name_table_entry_name
```

**Statements:** GROUP, PU, LU  
**Dependencies:** MDLTAB; PUTYPE=1 or 2

Specifies the name of the model name table entry to be used for this logical unit. If you code the MDLENT operand, you must also code the MDLTAB operand. If you omit MDLENT and include MDLTAB, VTAM uses the first entry in the table named in MDLTAB by default.

**MDLTAB**

```plaintext
MDLTAB = model_name_table_name
```

**Statements:** GROUP, PU, LU  
**Dependencies:** PUTYPE=1 or 2

Specifies the name of the model name table to be used for the logical unit. See "Model name table" on page 1054 for a description of how to define a model name table.

The model name is passed only when VTAM initiates the session with the application program, for example, only when the LOGAPPL operand is also coded.

If you omit the LOGAPPL operand or the MDLTAB operand, VTAM will not provide the name of the model name table to the application during LU-LU session initiation. However, you can provide the model name in other ways. You can enter it from the terminal on the MODEL operand of the LOGON command, or VTAM can send it to the application in a formatted session-initiation request. See "LOGON command" on page 1103 for more information on the MODEL operand of the LOGON command.

You can use the MODIFY TABLE command to dynamically replace a model name table. See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

**MLTGPR**
Assigns a priority to a line in a MLTG. The lowest priority is 0 and the highest priority is 255. NCP selects higher priority lines before lower priority lines when you code MLTGORDR=MLTGPI on the BUILD definition statement. If you code MLTGORDR=FIFO on the BUILD definition statement or if you do not code MLTGORDR, the MLTGPI keyword is ignored.

**MODETAB**

Specifies the name of a logon mode table to be used to correlate each logon mode name with a set of session parameters for the logical unit. The name you code must be the name of a logon mode table created as described in [“Logon mode table” on page 1026](#). If you do not supply a logon mode table for the logical unit on the MODETAB operand, an IBM-supplied default logon mode table (ISTINCLM) is used. If you specify a table, both the table you specify and the default table are used.

**MTU**
MTU = \texttt{mtu\_size}

\begin{itemize}
  \item \textbf{statements}: \texttt{PU}
  \item \textbf{dependencies}: Enterprise Extender only
  \item \textbf{range}: 1024 - 32767
\end{itemize}

Defines the maximum transmission unit (MTU), in bytes, that is used for predefined Enterprise Extender connections. This MTU size is reduced by VTAM to account for header lengths such as the IP, UDP, and LLC headers that are necessary for EE traffic. If MTU is not specified, VTAM uses the value learned from the TCP/IP stack. This learned value is the minimum MTU size (minus IP and UDP header sizes) associated with all the defined TCP/IP interfaces that can use this local static VIPA. If an MTU size with a value is specified, VTAM uses the lesser of this \texttt{mtu\_size} value and the value learned from TCP/IP. If the TCP/IP stack presents a value less than 768 bytes, VTAM sets the MTU to 768 bytes because this is the smallest packet size allowed by the HPR architecture. Typically, the MTU size for an EE connection is constant after the EE connection is established. However, if the MTU size of the TCP/IP stack changes, RTP pipes with endpoints in the same node as the TCP/IP stack dynamically detect these changes when their outbound packets are being transmitted. If the learned value is less than the \texttt{mtu\_size} value, VTAM uses the learned value. Reasons for MTU size changes include the following items:

\begin{itemize}
  \item New IP routes with different local MTU sizes become available
  \item Existing IP routes become unavailable
  \item Path MTU discovery is enabled for IPv4 or IPv6 EE connections (see \textit{PMTUD start option} on page 911 for more information), and path MTU changes are discovered in the IP network
\end{itemize}

You can enable path MTU discovery for IPv4 and IPv6 EE connections (see \textit{PMTUD start option} on page 911 for more information). If path MTU discovery is enabled, VTAM can dynamically learn about any MTU size changes that occur in the IP network. RTP pipes traversing Enterprise Extender connections can reduce their packet size to avoid IP fragmentation. In some environments, you might not be able to use path MTU discovery because firewalls must allow ICMP messages to be forwarded back to the host. In these cases, specify the MTU operand to control the maximum packet size that VTAM sends out on an Enterprise Extender connection.

\textbf{Tips:}

\begin{itemize}
  \item If the \texttt{mtu\_size} value is specified and IPv4 Enterprise Extender packets are being fragmented in the IP network, you might need to reduce the \texttt{mtu\_size} value to account for optional headers that might be present.
  \item For EE connection networks, you can define the connection network GROUP definition statements in the EE XCA major node.
  \item For dial-in Enterprise Extender connections with associated PUs that are dynamically created, you can define the MTU parameter in the model major node (DYNTYPE=EE) PU definition statement.
\end{itemize}
Specifies the name associated with the resource being defined.

**On VBUILD:**
name is optional. If a syntax or definition error is detected with this statement during activation of the major node, name appears in the message issued to identify the error.

The optional name you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node’s status.

**On PATH:**
Provides an optional name for the PATH statement.

**On PU:**
Provides the required minor node name of the physical unit represented by this definition statement.

For an APPN PU, name must differ from control_point_name (specified on CPNAME) to ensure that a CP-CP session establishment succeeds. If name and control_point_name are duplicates, the session establishment fails and sense code 08880001 is given.

**On LU:**
Provides the required minor node name of the logical unit represented by this definition statement.

**NATIVE**
Specifies whether this link station represents a connection to a native node.

If NATIVE is not specified on the ADJCP definition statement or on the PU
definition statement, the two nodes negotiate their subnetwork affiliation during
connection establishment.
- If the NETIDs match, the connection defaults to a native connection.
- If the NETIDs are different, the connection defaults to a nonnative connection.

No subnetwork negotiations will occur if NATIVE has been specified; in this case
the value between the two nodes must match for connection to occur.

If you code NATIVE on both the ADJCP and PU definition statements, the values
must match.

For transmission groups (TGs) between nodes, the value specified or negotiated for
NATIVE must match. If a TG is already active, subsequent TG activations must not
have a specified or negotiated value for NATIVE that conflicts with the value of
NATIVE for the active TG. If there is a conflict, the TG attempting activation is not
allowed to become active. If the activation is for a virtual-route-based transmission
group then the SSCP-SSCP session will also fail.

**NETID**
Specifies a 1–8 character network identifier.

**Note:** The use of the national character, #, is discouraged, because it might not be on keyboards of terminals produced in some countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

**NETID=network_id**

For **type 4 and 5 physical units**: NETID identifies the network in which this physical unit resides. It is required for a connection to another network.

If you do not specify a value, the network specified as NETID in the VTAM start options is used.

For **type 2 physical units**: This operand is used in conjunction with the XNETALS start option in support of nonnative network connection. See “Chapter 4, “VTAM start options,” on page 755” for information about XNETALS.

If you specify NETID on the PU definition statement, VTAM ensures that when the physical unit is active, the connecting resource is within the network specified by NETID. If you omit NETID, VTAM dynamically processes the network ID when connections are established. If you code NETID, and nonnative network connection is not supported, the definition is rejected unless the operand specifies VTAM’s network id.

**NETID=(,NOXNETLS)**

Specifies that VTAM does not allow nonnative network attachment for LU-LU sessions for this PU. When NETID=(,NOXNETLS), VTAM indicates to the NCP that this PU uses VTAM’s native network ID.

LU-LU sessions might fail if the NETID sent in the BIND from the LU is different than the NETID specified in VTAM (even though PU-PU connections can be established). If no NETID is included in the BIND, or if the NETID is identical to that specified by VTAM, the LU-LU session should activate.

If a dynamic nonnative dial-out connection is required (that is, a session request drives the dial), and it is not an APPN connection, then NETID must be predefined. This ensures that subsequent session requests do not fail because of VTAM’s inability to locate dependent logical units by using the dynamic nonnative NETID of the physical unit. This is also true when establishing a DLUS pipe, which is an APPN connection.

For dependent LU server PUs, VTAM will not dynamically process the network ID when CPSVRMGR sessions are established. To correctly define the NETID of a dependent LU server, if the NETID coded on the DLURNAME operand on the PATH statement is different from VTAM’s network ID, the NETID parameter on the PU statement must be coded with the NETID coded on the DLURNAME otherwise the VTAM’s network ID will be used for the PU.
NN

<table>
<thead>
<tr>
<th>NN = NO</th>
<th>NN = YES</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO</td>
<td>YES</td>
</tr>
</tbody>
</table>

Specifies whether the adjacent node is expected to be a network node. If a value for the NN operand is specified, it will be validated when a connection is attempted to the adjacent node. If the adjacent CP is not the type of node that is expected, then connection setup fails. If the NN operand is not specified, then the APPN capabilities of the adjacent node are identified and accepted when a connection is established.

**NN=NO**

Specifies that the adjacent node is expected to be an end node.

**NN=YES**

Specifies that the adjacent node is expected to be a network node.

**Restriction:** In a peripheral subnetwork boundary configuration between a border node and a network node without border node function, the border node appears to the non-border node as an end node. In this case, coding NN=YES on the definition statement that represents the border node, a connection failure will occur. Avoid the failure by not coding the NN operand or by coding NN=NO.

**PACING**
PACING = pacing_number

**statements:** GROUP, PU, LU  
**dependencies:** PUTYPE=1 or 2  
**range:** 0–63

Specifies how VTAM paces the flow of data from the boundary node, which performs pacing for a channel-attached SNA device, to the SLU. The PACING value is not used in all configurations. See z/OS Communications Server: SNA Network Implementation Guide for more information.

**Note:** You can also use the FIXED option on this operand. FIXED is an NCP option that allows you to override adaptive session pacing and use fixed pacing instead.

To use the FIXED option, you must code (N,,FIXED). The second comma represents a minimum value that is used by some pre-ENA nodes.

For more information about pacing, see z/OS Communications Server: SNA Network Implementation Guide.

**PASSLIM**
Determines, for switched physical units connected through an NCP, the maximum number of contiguous PIUs that the NCP sends to the physical unit at one time.

If the switched PU is a type 4 or 5 PU, this operand is ignored. In a switched subarea environment, the value on PASSLIM is overridden with the value of MAXOUT.

**PDELAY**

Specifies the maximum propagation delay of the link for the transmission group. Propagation delay represents the time needed for a signal to travel from one end of the link to the other.

**PDELAY=NEGLIGIB**
- Indicates a local area network delay (less than .48 milliseconds).

**PDELAY=TERRESTR**
- Indicates telephone network delay (between .48 and 49.152 milliseconds).

**PDELAY=PACKET**
- Indicates a packet-switched network delay (between 49.152 and 245.76 milliseconds).

**PDELAY=LONG**
- Indicates a satellite delay (greater than 245.76 milliseconds).
**PID**

```
[PID=n__path_identifier]
```

**statements:** PATH  
**dependencies:** none  
**range:** 1–255

Specifies an identifier for the path being defined. The operator uses this identifier to change the status of the path.

This identifier is unique for a given physical unit.

---

**PRTCT**

```
[PRTCT=n__password]
```

**statements:** PU  
**dependencies:** PUTYPE=4 or 5

Specifies a 1–8 EBCDIC character password used to verify the identity of a caller or receiver when switched subarea connections are being established. If you do not specify this operand, the identity of the caller or receiver is not verified when a connection is established.

The password value must be the same at the SSCP of the caller and the SSCP of the receiver. When one side has the password defined, but the other side does not, or if the caller, receiver, or other SSCP does not support call security verification, the connection fails regardless of which side initiates the call. The NCP must support call security verification.

This operand is valid only when PUTYPE=4 or PUTYPE=5.

---

**PUTYPE**
PUTYPE = 2

PUTYPE = 1

PUTYPE = 4

PUTYPE = 5

statements: GROUP, PU
dependencies: none

Specifies the physical unit type. The physical unit type depends on the type of cluster controller (such as 3767, 3174, or AS/400). To determine the physical unit type for a given device, see the component description manual for that device.

For a type 2.1 physical unit, code PUTYPE=2.

For switched subarea connections, code PUTYPE=4 or PUTYPE=5. PUTYPE=4 defines the physical unit as an NCP. PUTYPE=5 defines the physical unit as a VTAM.

Note: For PU types 4 and 5, an auto call unit (ACU) is required for subarea dial connections to be established.

The physical unit type for a LAN switched station must be 2.

For a PU type 1 or 2, or type 2.1, the value of PUTYPE occupies bits 4–7 of the 48-bit station ID constructed by VTAM for switched network operation. See "IDNUM" on page 611 for a description of the station ID.

Note: The only valid value for Enterprise Extender is PUTYPE=2.

REDDELAY
The REDDELAY statement specifies the number of seconds between each attempt to redial an Enterprise Extender connection. The default is 30 seconds. The minimum value that can be coded is one second. Customers with a very large number of Enterprise Extender connections should not use a small value for REDDELAY, as the amount of activity generated during sustained outages could have significant CPU impact.

**REDIAL**

- **statements:** `PATH`
- **dependencies:** `MEDIUM=HPRIP`
- **range:** `1–1200`

Specifies the number of times dialing is to be retried before returning a dialing error to VTAM. The interval specified by the REDDELAY operand will be observed between redials for Enterprise Extender connections.

If you code 0, dialing is not retried. If you code FOREVER and the switched PU is used for an Enterprise Extender connection, dialing is retried as necessary until either a connection is established or the PU is deactivated. If you code FOREVER and the PU is not used for an Enterprise Extender connection, the value 254 is used.

**REGISTER**

For independent LUs:
Specifies how a resource should be registered. The default depends on whether you code REGISTER for dependent or independent LUs.

**REGISTER=CDSERVR**
Indicates that an end node resource is registered to a network node server and directory resource registration is requested for it. A network node resource is registered at the central directory server. If the node is configured as a central directory server, this value has the same effect as NO.

**REGISTER=NETSRVR**
Indicates that the end node resource should be registered to its network node server, but that directory registration should not be requested for it.

For dependent LUs, NETSRVR is the default, and LOCADDR must be specified as a nonzero value.

Dependent LUs are not usually the targets of LU-LU session setup attempts and are therefore not likely to be the targets of an APPN LOCATE search. However, because end nodes cannot be searched, you can register their dependent LUs with their network node server.

**REGISTER=NO**
Indicates that the resource should not be registered.

For independent LUs, NO is the default, and LOCADDR must be specified as 0.

**RESETLU**
**RESETLU**

Specifies whether a DACTLU request followed by a ACTLU request will be sent during clean-up processing.

**RESETLU=YES**

Specifies that during clean-up processing for dependent LUs, VTAM will issue an ACTLU request after the DACTLU request.

**RESETLU=NO**

Specifies that during clean-up processing for LU-LU sessions, RESETLU=NO prevents a DACTLU-ACTLU from flowing to an LU.

**RESSCB**

Specifies the number of session control blocks reserved by the NCP for the logical unit. This operand is valid for independent logical units only. If you code it for other logical units, VTAM ignores it.

This operand applies only to type 2.1 peripheral nodes.

For information about NCP restrictions on this operand, see *NCP, SSP, and EP Resource Definition Reference*.
SAPADDR = 8

statements: PATH
dependencies: MEDIUM=HPRIP

Specifies the service access point for the switched connection through an Enterprise Extender network.

Note:
1. The SAP values for Enterprise Extender are multiples of 2 and for all other mediums a multiple of 4 applies. A null SAPADDR value has the same meaning as not coding the SAPADDR operand.
2. The remote SAP value specified by SAPADDR must be unique. For more information, see the z/OS Communications Server: SNA Network Implementation Guide.
3. Exercise caution when specifying a Switched Major Node which will be active when the connection network function is in effect for HPR/IP. The remote SAP values specified with SAPADDR on the PATH statement must not match local SAP values specified in a node in the HPR/IP Shared Access Transport Facility (SATF) if connections will be attempted through the defined PATH (and PU) while connections through the connection network are active to that node.
4. Use the defaults for SAPADDR on the PATH definition statements.

SECNET
Specifies whether this physical unit is associated with a secondary network containing resources whose connectivity is not defined to VTAM.

This operand applies only if you have IBM 3710 Network Controllers or IBM 3174 token-ring attachments in your network.

**SECNET=NO**
Data received by VTAM from this physical unit is not flagged as requiring special problem determination considerations when routed to a communication network application.

**SECNET=YES**
Data that is received by VTAM from this physical unit is flagged as requiring special problem determination considerations when routed to a communication network application.

**SECURITY**
Specifies the security level of the transmission group. The following options range from the least secure level to the most secure level.

**SECURITY=UNSECURE**
- Specifies no security level.

**SECURITY=PUBLIC**
- Specifies a public switched network.

**SECURITY=UNDERGRO**
- Specifies an underground cable, not guarded.

**SECURITY=SECURE**
- Specifies a secure conduit, not guarded.

**SECURITY=GUARDED**
- Specifies a guarded conduit, physical only.

**SECURITY=ENCRYPT**
- Specifies link encryption.

**SECURITY=SHIELDED**
- Specifies a guarded conduit, physical and radiation shielded.

**SHM**
specifies whether the switched subarea connection made using this PATH
definition statement should be a short-hold mode connection.

SHM is mutually exclusive with the LINENM operand. It is valid only for
connections established through X.25 NCP Packet Switching Interface (NPSI).

**SHM=NO**

Specifies that short-hold mode is not to be used.

**SHM=YES**

Specifies that short-hold mode is to be used, and the value specified by the
SHMTIM operand should be used as the short-hold mode timer.

When SHM=YES, the DIALNO and GRPNM operands are required. The
GRPNM operand is used to specify which group is appropriate for this PATH
definition statement.

**SHMTIM**
### SHMTIM

**statements:** `PATH`  
**dependencies:** `NPSI connections`  
**range:** `0.0–6553.5 seconds`

Specifies the time, in seconds and tenths of seconds, to be used as the short-hold mode timer for a short-hold mode connection that uses the SHM operand.

If `SHMTIM=0.0`, a default timer value associated with the line is used. Consult the documentation of the product providing the short-hold mode support for the range of timer values that it supports.

This operand is valid only for connections established through NPSI.

### SPAN

**statements:** `PU, LU`  
**dependencies:** `NetView`

Defines a span of control for VTAM minor node resources. Code this operand if you are using the NetView program. For a full description, see *IBM Tivoli NetView for z/OS Installation: Configuring Graphical Components*.

The NetView program checks the SPAN value, but VTAM ignores it.

### SSCP FM
statements:  GROUP, PU, LU
dependencies:  PUTYPE=1 or 2

Determines what type of RUs the logical unit can support in its communications with the SSCP.

**SSCPFM=FSS**
Specifies that field-formatted RUs are supported.

*Note:* You might receive character-coded messages when SSCP FM=FSS.

**SSCPFM=USSNTO**
Specifies that this logical unit is supported by NTO in the boundary node, and character-coded RUs are supported over its SSCP-LU sessions.

**SSCPFM=USSSCS**
Specifies that character-coded RUs are supported.

**SSCPFM=USS3270**
Allows printing of a user-defined logon message (USSMSG10) on BSC printers.
Code SSCP FM=USS3270 for terminals attached to a controller defined as an SDLC 3271 (PUTYPE=1) or SDLC 3275 (PUTYPE=1).

**SSCPFM=USS3275**
Prevents printing of a user-defined logon message (USSMSG10) for a terminal defined as an SDLC 3275 (PU type 1) that might have an attached printer.
Code SSCP FM=USS3275 to prevent the user-defined logon message from printing on BSC printers.

**SSCPFM=USS3780**
Coded for non-SNA devices supported with NTO that use the 3780 protocol. These devices use USS command facilities such as LOGON and IBMTEST and are supported only by Release 2 of NTO. See *NTO Planning, Migration, and Resource Definition* for more information.
SUBAREA—"inbound_xid_value"

SUBAREA—"suba_addr"

specifies the subarea address assigned to the physical unit (NCP or VTAM) represented by this PU definition statement. VTAM compares the subarea number in the partner’s XID with the defined subarea number to verify that the node on the other end of the switched subarea connection is the one that was expected. This operand is required when PUTYPE=4 or PUTYPE=5.

The SUBAREA value must be the same as the subarea address defined for this physical unit in the other domains of the network. This operand is for switched subarea connections.

**TERM**

TERM—"type"

specifies the device data stream compatible characteristics. An application program can find these characteristics by executing an INQUIRE DEVCHAR macroinstruction that names this virtual logical unit. The information is placed in the DEVAUXTP field of the DEVCHAR DSECT. (DEVAUXTP is an 8-byte storage area provided by the application program.) For more information about the DEVCHAR DSECT, see z/OS Communications Server: SNA Programming.
TGCONF = ( MULTI , NORMAL )

TGCONF = ( , )

MULTI NORMAL

SINGLE STANDBY ,

nn

NONE

specifies whether this subarea link station is in a multilink or single-link transmission group. TGCONF is valid only if you specify PUTYPE=4.

**Multilink Transmission Groups:**
Code TGCONF=MULTI or omit TGCONF for all subarea link stations in a multilink transmission group.

**Single-Link Transmission Groups:**
You do not need to code TGCONF for a single-link transmission group. However, coding TGCONF=SINGLE for a transmission group that contains a single line can improve NCP performance for the transmission group.

**Standby Stations:**
TGCONF=STANDBY designates if a subarea station is to be used in hot-standby mode. TGCONF=STANDBY is not valid when TGCONF=SINGLE is coded.

**Congestion Relief:**
TGCONF=(MULTI,STANDBY,nn) designates that a subarea station is to be used for hot-standby and congestion relief. The subarea transmits data to relieve congestion in the transmission group configuration if queued data reaches the value specified by nn. nn indicates the percentage reached of a threshold specified by the totthresh suboperand of the ERn keyword on the associated PATH definition statement. The totthresh suboperand defines the total number of bytes that can be queued for transmission to an MLTG before NCP takes control measures to slow down the TG traffic. Valid values for nn are 3, 6, 12, 25, 50, 75, 87, 93, 97, and 100. NONE designates that the station is used only for hot-standby and not for congestion relief.

**TGN**

<table>
<thead>
<tr>
<th>For subarea TGN:</th>
<th>For APPN TGN:</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ TGN = \text{any} ]</td>
<td>[ TGN = \text{any} ]</td>
</tr>
<tr>
<td>[ TGN = 1 ]</td>
<td>[ TGN = 1 ]</td>
</tr>
<tr>
<td>[ TGN = \text{any} ]</td>
<td>[ TGN = \text{any} ]</td>
</tr>
<tr>
<td>[ \text{tg} _\text{number} ]</td>
<td>[ \text{tg} _\text{number} ]</td>
</tr>
</tbody>
</table>
Subarea:
Specifies the transmission group (TG) of the switched subarea connection associated with this physical unit. For subarea, this operand is valid only when PUTYPE=4 or PUTYPE=5.

APPN:
Specifies the transmission group (TG) that represents the connection between this node and the destination APPN node. For APPN, this operand is valid when PUTYPE=2.

By specifying a `tg_number`, you indicate a preferred TG which might or might not be used depending on how the partner nodes negotiate the number.

For APPN-capable nodes, the CPNAME and NETID operands are required with the TGN operand. Note that you can specify the CPNAME operand without the TGN operand and the transmission group number will be negotiated. However, if a value for TGN is specified, then a value for CPNAME and NETID must also be specified.

**TGN=any**
For subarea, allows the adjacent node to accept whatever valid transmission group number it receives on the XID. You can specify TGN=ANY for only one end of the transmission group. For the other end, specify a valid TGN number or let it default to 1. If you specify TGN=ANY for both ends, an error occurs. For APPN, coding TGN=ANY, or not coding TGN, allows the transmission group number to be negotiated. For connections to LEN nodes, or to any node that does not support parallel TGs, TGN=0 is always assumed.

**Note:** VTAM does not validate TGN when coded for subarea resources.

**TGP**
TGP = tgp_name

**statements:**  
GROUP, PU

**dependencies:**  
APPN only

Specifies the name of a transmission group (TG) profile. The characteristics of the TG profile (along with any modifiers) become the characteristics of the PU. If TGP is not specified or has not been activated when the PU becomes active, default TG characteristics are assigned.

See "APPN transmission group profile" on page 720 for more information on transmission group profiles.

**TOPO**

**TOPO = CONNECT**

**TOPO = connection_cost**

**TOPO = NONE**

**controls the way APPN connections using this link station will be reported to APPN topology and routing services for inclusion into APPN functions.**

If you change a VTAM's node type from network node to end node by deactivating the VTAM and then reactivating it with NODETYPE=EN, you might need to change the switched PU definitions in partner VTAMs to use TOPO=None. This is suggested only if the switched PU connection is predefined and you have been using this VTAM as an intermediate routing node.

**TOPO=CONNECT**

APPN topology and routing services receives information about the connection and its characteristics when the connection is activated. When the connection is deactivated, APPN topology and routing services receives notification that the TG is inactive.

**TOPO=connection_cost**

Indicates that the connection is to be treated as auto-active with a connection cost override. Auto-active allows a switched physical unit to be selected for routing prior to the actual establishment of the connection.

If a connection_cost is coded, then CPNAME, NN, and TGN=tg_number must also be coded. If one or more of these operands is not coded, then the connection cannot be reported prior to activation and the TOPO support is forced to CONNECT.

**Note:** connection_cost is valid only for switched major nodes.
TOPO=None

APPN topology and routing services receives no information concerning
connections established with this link station. This means that connections
using this link station are not available for APPN network services.

TYPE

 statements: VBUILD
dependencies: none

Means that the VBUILD definition statement defines a switched major node to
VTAM. All physical units defined in this major node can be connected only by a
switched link. This operand is required.

UPARM1

 statements: GROUP, PU
dependencies: APPN only
range: 0–255

Specifies a user-defined operand value.

See z/OS Communications Server: SNA Network Implementation Guide for an
example of using the UPARM operands.
UPARM2 = user-defined_operand_value

Statement: GROUP, PU
Dependencies: APPN only
Range: 0–255

Specifies a user-defined operand value.

UPARM3 = user-defined_operand_value

Statement: GROUP, PU
Dependencies: APPN only
Range: 0–255

Specifies a user-defined operand value.

USE = YES

Statement: PATH
Dependencies: none

Specifies whether the path is initially usable or not usable. This attribute of the path can be modified by the VTAM operator, using the VARY PATH command. The effect of USE=YES and USE=NO for a path is similar to the effect of ISTATUS=ACTIVE and ISTATUS=INACTIVE for a minor node.

USE=NO

Specifies that the path is initially not usable.

USE=YES

Specifies that the path is initially usable.
USSTAB = ISTINCDT
USSTAB = uss_table_name

Specifies the name of a USS table that VTAM uses to process character-coded input that it receives from the logical unit.

A terminal user can issue a USS command with the LANGTAB operand. This causes a second USS table to be associated with the logical unit, which overrides the table specified with USSTAB. If you do not code USSTAB and a LANGTAB USS table is not in use, the IBM-supplied USS table (ISTINCDT) is used. For more information on USS tables, see “Unformatted system services tables” on page 1069.

USSTAB is valid for dependent logical units only.

VERALSID

Notes:
1 VERALSID=NO is the only valid value if the TRLE operand is coded.

Specifies whether the adjacent link station name should be used to determine the link station. When specified as YES, the connection endpoints must supply the same name in the CV0EF7 vector that is appended to the XID3. The name supplied in CV0EF7 is the name of a switched PU or the name of a PU under a non-switched line.

VERALSID=NO
Specifies that the adjacent link station name is not used to determine the link station.

VERALSID=YES
Specifies that the adjacent link station name is used to determine the link station.

When specified as YES, the connection endpoints must supply the same name in the CV0EF7 vector that is appended to XID3s. The name supplied in CV0EF7 is the name of a switched PU or the name of a PU under a non-switched line.
VERID

```
VERID = call_identifier (call_identifier)
```

**statements:** PATH
**dependencies:** GRPNM

Specifies one or more call identifiers to be used by call ID verification.

*Call identifier* is a general term for an identifier provided by a switched network that identifies the telephone number of the device at the other end of the switched line. For example, the calling DTE address is such an identifier in X.25 networks. Each call identifier is specified as a character string and can be up to 251 bytes long.

If you code VERID:
- GRPNM is required.
- VERIFY=NONE is not valid.
- The list form is required if you specify more than one call identifier. For example, to specify call identifiers *id1* and *id2*, code VERID=(id1,id2).

**VERIFY**
Specifies when call ID verification is to be performed if this PATH definition statement is used.

If the VERIFY operand is specified or defaults to NONE and all PATH definition statements for this physical unit specify or default to CALL=INOUT, an inbound call is accepted from a switched line group.

If call ID verification (VERIFY=IN, VERIFY=OUT, or VERIFY=INOUT) is specified on any PATH definition statement, an inbound call is accepted only if it is using a line group for which there is a PATH definition statement that specifies or defaults to CALL=IN or CALL=INOUT. That is, if one PATH definition statement specifies call ID verification, all valid paths (inbound and outbound) must be defined, including those paths for which call ID verification is not to be performed.

If you code the VERIFY operand with any value other than VERIFY=NONE, GRPNM is required.

**VERIFY=IN**
Specifies that call ID verification is to be performed for inbound calls using this PATH definition statement. If VERIFY=IN, CALL=OUT is not valid. If VERIFY=IN, VERID and GRPNM are required.

**VERIFY=INOUT**
Specifies that call ID verification is to be performed for inbound and outbound calls using this PATH definition statement. If VERIFY=INOUT, CALL=INOUT is required. If VERIFY=INOUT, VERID and GRPNM are required.

**VERIFY=NONE**
Specifies that call ID verification is not to be performed for inbound or outbound calls using this PATH definition statement. If VERIFY=NONE, VERID is not valid.

**VERIFY=OUT**
Specifies that call ID verification is to be performed for outbound calls using this PATH definition statement. If VERIFY=OUT, CALL=IN is not valid. If VERIFY=OUT, VERID and GRPNM are required.

**VPACING**
VPACING = \text{number\_of\_requests}

Specifies how VTAM paces the flow of data from the boundary node containing the PLU to the boundary node that performs pacing for a channel-attached SNA device.

\textbf{Note:} You can also use the FIXED option on this operand. FIXED is an NCP option that allows you to override adaptive session pacing and use fixed pacing instead.

To use the FIXED option, you must code \texttt{(N,,FIXED)}. The second comma represents a minimum value that is used by some pre-ENA nodes.

For more information about pacing, see \textit{z/OS Communications Server: SNA Network Implementation Guide}.

\textbf{VTAMTOPO}
Specifies whether the VTAM topology agent reports the major nodes or subordinate minor nodes to the CMIP manager application programs for SNA local topology.

Specifying VTM.VTAMTOPO allows VTAMTOPO to be handled appropriately by the NCP Definition Facility for versions of NDF that do not recognize VTAMTOPO natively.

**Note:** See Appendix C, “VTAMTOPO filtering option reporting,” on page 1373 for more information about VTAMTOPO filtering option reporting.

**VTAMTOPO=REPORT**

- **On VBUILD:** Specifies that the VTAM topology agent reports subordinate PUs. This does not override any value specified on the GROUP or PU.

- **On GROUP:** Specifies that the VTAM topology agent reports subordinate PUs. This does not override any value specified on the PU.

- **On PU:** Specifies that the VTAM topology agent reports this PU.

**VTAMTOPO=NOREPORT**

- **On VBUILD:** Specifies that the VTAM topology agent does not report subordinate PUs. This major node is not reported by itself. This does not override any value specified on the GROUP or PU.

- **On GROUP:** Specifies that the VTAM topology agent does not report any subordinate PUs. This does not override any value specified on the PU.

- **On PU:** Specifies that the VTAM topology agent does not report this PU.

**VTAMTOPO=INCLUDE**

- **On VBUILD:** Specifies that the VTAM topology agent reports all subordinate PUs. VTAMTOPO=INCLUDE overrides any other value specified on the GROUP or PU.

**VTAMTOPO=IGNORE**

- **On VBUILD:** Specifies that the VTAM topology agent does not report any subordinate PUs. VTAMTOPO=IGNORE overrides any other value specified on the GROUP or PU.
XNETALS determines whether the network ID of an attaching node or VTAM's network ID is used for an adjacent nonnative node. Used to control how a nonnative node can attach to VTAM.

Note: The XNETALS option has changed in the way VTAM treats PUs that send in nonnative NETIDs in their XID. VTAM used to let the PU connect and treated it as if it sent in the same NETID that VTAM started with but that is no longer true. If the NETID that the PU sent in its XID is not the same as VTAM's and XNETALS=NO or is defaulted, IST680I will be issued.

XNETALS=NO
Indicates that VTAM's network ID will be used to represent the connecting node's network ID.

During connection:
- If the NETID operand is specified on the PU statement, then it must specify VTAM's network ID.
- If the NETID operand is not coded on the PU statement, then VTAM's network ID is assumed and the adjacent node's network ID is ignored.

During LU-LU session establishment:
- If the network ID for the originating LU is specified (during connection time) and is not identical to VTAM's network ID, the session will not be established.
- If the originating LU is not using a network qualified name, VTAM's network ID is assumed and the session will be established.

Note: If the PU is a dependent LU server or requester, nonnative network connection is not allowed if XNETALS=NO is specified and the NETID operand is not coded.

XNETALS=YES
indicates that the network ID of the adjacent PU will be used for connection and session establishment.

During connection:
- If the NETID operand on the PU statement is coded, it will be used as the PU's network ID. In this case, the network ID specified on the PU statement must match the network ID specified by the attaching node during connection processing.
- If the NETID operand is not specified on the PU statement, then the network ID specified by the attaching node during connection processing will be assumed.

During LU-LU session establishment:
- If the network ID for the originating LU is specified and does not match the PU's network ID, the session will not be established.
If the originating LU name is not a network qualified name, the session will be established. VTAM will qualify the LU's name with the attaching node's network ID.

**Note:** Providing that XNETALS=YES is coded, you can prohibit nonnative network attachment for LU-LU sessions for a particular PU by coding XNETALS=NO on the GROUP, LINE or PU statements.

---

### Transport resource list major node

To define a transport resource list major node, code the following definition statements:

- One VBUILD definition statement to begin the transport resource list major node.
- One transport resource list element (TRLE) definition statement for each multipath channel (MPC) connection.

The transport resource list element is not a resource, but describes the connectivity characteristics of the multipath channel line that is used for the connection.

An MPC connection can be used for both SNA & TCP/IP application traffic. See [z/OS Communications Server: IP Configuration Guide](https://www.ibm.com) Any user of the connection is referred to as an upper layer protocol (ULP).

For a connection between VTAM and an adjacent APPN node, the TRLE operand on the PU definition statement in the local SNA major node that defines the adjacent APPN node identifies the TRLE definition statement VTAM uses to route data over the connection.

For a connection between VTAM and a port on an Open Systems Adapter through which native access to an ATM network is achieved, the PORT definition statement in the XCA major node that defines the port identifies the TRLE definition statement VTAM uses to route data over the connection. The determination of which TRLE definition statement is used is made by matching the port name specified on the PORTNAME operand on the PORT definition statement to the port name specified on the PORTNAME operand on a TRLE definition statement in the TRL major node.

For all MPC connections defined by way of a TRLE definition, the subchannel addresses for the MPC connection must be defined. For all connections except OSA-Express, each subchannel address is dedicated to one direction of flow, either read or write, and the addresses are defined on the READ and WRITE operands.

With MPC dynamics support, if at least one WRITE and one READ subchannel path is allocated successfully, an MPC connection is activated. Additional paths (defined but not ONLINE) in an MPC group can later be dynamically added to the active group using the MVS command VARY device ONLINE. For example, if there is a need for an increase in capacity to allow for extra traffic over a channel, additional paths can be added to the active group without disruption. Similarly, paths can be deleted from the active group when no longer needed using the MVS command VARY device OFFLINE. Deletion of the last active READ or WRITE device can be controlled (allowed or disallowed) by MPC dynamics. See "WRITE" on page 674 for more information.
For OSA-Express, there are two types of subchannels. There are subchannels dedicated to control flows, and there are subchannels dedicated to data. Control flows include signalling flows to set up and terminate connections, and data identified by the ULP to be transmitted over control channels. Data is any data the ULP identifies as not control. Data subchannels can be used for sending and receiving data through the OSA-Express device, or for receiving trace data from the OSA-Express device. The control subchannels are defined on the READ and WRITE operands, and the data subchannels are specified on the DATAPATH operand. As with Open Systems Adapter ATM access, the PORTNAME operand on the TRLE definition statement is used by TCP/IP for TCP/IP to achieve connectivity through the OSA-Express device.

**Note:** For OSA-Express, Channel Unit Address and Subchannel are the same as the device number as specified in the hardware configuration definitions.

For detailed instructions on setting up an OSA-Express feature, see zEnterprise System and System z10 OSA-Express Customer’s Guide and Reference. See the z/OS Communications Server: SNA Network Implementation Guide for more information about MPC connections.

### Table 35. Transport resource list major node statements and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=TRL</td>
<td>674</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>TRLE</td>
<td>DATAPATH</td>
<td>662</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LNCTL</td>
<td>664</td>
<td>R</td>
<td>MPC</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXBFRU</td>
<td>665</td>
<td></td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td>4</td>
<td>MPCLEVEL=HPDT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MAXREADS</td>
<td>665</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 35. Transport resource list major node statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sift</td>
<td>effect Default</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:
R Required operand.

Transport resource list major node coding notes
This information contains coding notes for the transport resource list major node.

VBUILD definition statement
Code a VBUILD definition statement for each TRL major node.

TRLE definition statement
Code one TRLE definition statement for each MPC line that is used for a connection.

Transport resource list major node full syntax
The full syntax for the transport resource list major node follows. The syntax for each operand is repeated in “Transport resource list major node operand descriptions” on page 662.

VBUILD
VBUILD TYPE = TRL

TRLE

name TRLE

DATAPATH = channel unit address

LASTRW = DISALLOW

LNCTL = MPC

MAXBFRU = 5

MAXREADS = 4

MPCUSAGE = SHR

PORTNAME = (port_name, link_num)

PORTNUM = 0

MPCLEVEL = HPDT

NOHPDT

QDIO

PACKING = OFF

max_pdu_size

PACKING = ON
Notes:
1. MAXBFRU defaults to 1 when HPR=NO, XID=NO, CONNTYPE=LEN, or the VTAM node is not HPR capable.
2. MAXBFRU defaults to 5 when HPR=YES, XID=YES, CONNTYPE=APPN, and the VTAM node is HPR capable.

Transport resource list major node operand descriptions
The following information describes the operands you can code for the transport resource list major node.

DATAPATH
Statements: TRLE

range X’000’-X’FFFF’

Specifies the subchannel addresses used to read and write data through an OSA-Express connection. You must code at least one DATAPATH subchannel. You must specify only subchannel addresses that have been defined as OSA devices in the hardware I/O configuration (using HCD, IOCP, or another suitable means of hardware definition). Do not include subchannel addresses that are defined as OSAD devices. Do not code a subchannel address for the DATAPATH operand that is also coded for either the READ operand or the WRITE operand in the same host. See zEnterprise System and System z10 OSA-Express Customer’s Guide and Reference.

Subchannel addresses can be defined as a single address, as a range of addresses, or both. A range is indicated by a hyphen(-). VTAM generates addresses for the range. For example, DATAPATH=(132,128-130,13A) defines the addresses 132, 128, 129, 12A, 12B, 12C, 12D, 12E, 12F, 130, and 13A. A total number of 255 datapath addresses can be defined. For more information about hardware limits on the number of addresses that can be defined for this operand, see zEnterprise System and System z10 OSA-Express Customer’s Guide and Reference.

Each TCP/IP stack in the same logical partition instance that starts an OSA-Express interface in QDIO mode is assigned one of the DATAPATH channels by VTAM. If TCP/IP is using multiple virtual LANs (VLANs), each interface in the same logical partition instance that is started is assigned one of the DATAPATH channels by VTAM. Any TCP/IP in the same logical partition instance that starts an OSA-Express network traffic analyzer (OSAENTA) trace is also assigned one of the DATAPATH channels by VTAM. One OSA-Express network traffic analyzer can be active at any one time for each OSA port. You must code a sufficient number of DATAPATH subchannel addresses to accommodate the number of concurrent instances that will use an OSA-Express port. For example, if you want three instances of TCP/IP in the same logical partition and none of the three TCP/IP instances are using multiple VLANs or OSAENTAs, code the DATAPATH operand in the TRLE as follows: DATAPATH=(592,593,594), or DATAPATH=(592-594).
Specifies if the last active read or write device is allowed to be deactivated by the
MVS VARY OFFLINE command.

**LASTRW=DISALLOW**

Indicates if LASTRW=DISALLOW the MVS VARY OFFLINE command will not
deactivate the last read or write device within the MPC group. Instead of
deactivating the last device, VTAM will issue a warning message, IST1706I,
informing the operator that an attempt was made to deactivate the last R/W
device, and the command is not allowed. The device will remain in a pending
offline state. Instead of using VARY OFFLINE to deactivate the MPC group,
use the appropriate VTAM (VARY INACT) or TCP/IP (STOP DEVICE)
command, or VARY OFFLINE FORCE.

**LASTRW=ALLOW**

Specifies if LASTRW=ALLOW, the MVS VARY OFFLINE command will
deactivate any or all devices within the MPC group, including the last active
read or write device. When the last active read or write device is deactivated,
the MPC group will become inactive (for example, loss of connectivity to the
MPC device).

**Note:** Coding of the LASTRW operand is ignored for OSA. VTAM will ignore
VARY OFFLINE (without FORCE) for OSA in all cases.

**LNCTL**
Indicates that the link is a multipath channel link. This operand is required.

**MAXBFRU**

Notes:
1. MAXBFRU defaults to 1 when HPR=NO, XID=NO, CONNTYPE=LEN, or the VTAM node is not HPR capable.
2. MAXBFRU defaults to 5 when HPR=YES, XID=YES, CONNTYPE=APPN, and the VTAM node is HPR capable.

Specifies the number of 4K buffer pages VTAM uses to receive data when activating a multipath channel. The same MAXBFRU value is used for each of the READ subchannels. The resulting buffer size is \( \text{number_of_pages} \times 4K \). The total buffer space used is 4K multiplied by \( \text{number_of_pages} \) multiplied by the number of READ subchannels.

VTAM automatically substitutes a value of 16 for any coded value higher than 16 without issuing a warning message.

Some devices might have a hardware restriction on the maximum number of buffers. If there is a hardware buffer size for multipath channel, VTAM compares the MAXBFRU value and the hardware buffer size and uses the smaller of the two to avoid excessive fixed storage.

MAXBFRU is not used to determine the buffer size of the WRITE subchannels coded on this host. The WRITE subchannel buffer size is determined dynamically by VTAM, depending on the setting at the READ end of the channel.

You can reduce this value after examining VTAM multipath channel tuning statistics so that excessive storage is not allocated for each subchannel. To obtain tuning statistics output, code the TNSTAT start option.

For connections to an OSA-Express feature, MAXBFRU is fixed at 2 and should not be coded.

**MAXREADS**
MAXREADS = number_of_available_buffers

**statements:** TRLE
**range:** 4–8
**dependencies:** MPCLEVEL=HPDT

Specifies how many read buffers are available to read data over a high performance data transfer (HPDT) MPC connection.

This operand is valid only when MPCLEVEL=HPDT is specified.

**Note:** The number of read buffers on the TRLE definition statement defining a port on an Open Systems Adapter through which native access to an ATM network is achieved should always be defined to eight (MAXREADS=8.)

For connections to an OSA-Express feature, MAXREADS is fixed at 2 and should not be coded.

**MPCLEVEL**

**statements:** TRLE
**dependencies:** HPR start option, MAXREADS, PACKING, STORAGE

Specifies the level of MPC capability to be used for the connection. See z/OS Communications Server: SNA Network Implementation Guide for information about the different levels of MPC capability provided by VTAM.

**MPCLEVEL=HPDT**

Indicates that HPDT MPC is to be used for the connection. This operand is required for native ATM connections.

**Note:** For connections to an adjacent APPN node or an Open Systems Adapter port: When MPCLEVEL=HPDT, you must code HPR=RTP in the start list for this VTAM. This restriction does not apply if this MPC connection is only for IP traffic.

**MPCLEVEL=NOHPDT**

Indicates that non-HPDT MPC is to be used for the connection. If MPCLEVEL=NOHPDT is specified and a value is coded for the MAXREADS, PACKING, or STORAGE operand, the values specified for MAXREADS, PACKING, or STORAGE are ignored.

**MPCLEVEL=QDIO**

Indicates that the Queued Direct I/O interface is used for an OSA-Express feature. Control data is transmitted across the one READ and one WRITE device. Normal data is transmitted across a DATAPATH device, where each ULP is assigned its own device number for normal data when the device or
interface is started, and a device number for trace data when the OSA-Express network traffic analyzer (OSAENTA) trace is started. When the MPCLEVEL value is QDIO, HPDT MPC is also used for this connection across the control channels. When you code MPCLEVEL=NOHPDT, the PACKING and STORAGE operands are ignored. The MAXREADS value is forced to 2 and should not be coded.

**MPCUSAGE**

```
MPCUSAGE = SHR
MPCUSAGE = EXC
```

**statements:**

```
TRLE
```

**dependencies:**

```
MPCLEVEL=HPDT
```

Specifies whether multiple usage (sharing) or single usage (exclusive) of an MPC (with HPDT) group is allowed. This keyword is ignored for non-HPDT capable MPC groups. Users of the MPC group are the various protocol stacks (for example, APPN or TCP/IP).

**MPCUSAGE=SHR**

Indicates this group can be used by multiple protocol stacks at the same time.

**MPCUSAGE=EXC**

Indicates this group can be used only by a single protocol stack at a time. The first to activate the group becomes the exclusive user of that MPC group. All subsequent activation attempts (while the group is in use) will be rejected. Defining exclusive usage will provide improved performance.
On VBUILD:

name is optional. If a syntax or definition error is detected with this statement during activation of the major node, name appears in the message issued to identify the error.

The optional name you give to this definition statement can be the same as the name VTAM gives to a major node and its definitions when they are filed in the VTAM definition library. The name VTAM gives to the major node is used when messages are issued regarding the node's status.

On TRLE:

Assigns a required name for the TRLE. This name cannot match the VTAMLST member name containing this Transport resource list major node.

For an IBM Open Systems Adapter (OSA) port customized for HPDT ATM native mode, the name must match the OSA name specified at OSA customization through OSA/SF. If used by TCP/IP, the OSA name must also be specified as the device name in the TCP/IP profile DEVICE statement.

For an IBM Open Systems Adapter-Express (OSA-Express) port operating in QDIO mode the name specified here will be downloaded into the OSA-Express as OSA name at the time the TCP/IP device for this OSA connection is activated.

PACKING
Specifies whether to disable or enable HPDT packing. Use PACKING=max_pdu_size, varying max_pdu_size to tune HPDT packing.

PACKING=OFF

Specifies the packing size limit to zero which, in effect, disables the function. PACKING=OFF is the default.

PACKING=ON

Enables HPDT packing by setting a packing size limit of 2048.

PACKING=max_pdu_size

Enables HPDT packing by setting a packing size limit of max_pdu_size. The valid range for max_pdu_size is 1024 to 8192 inclusive.

Protocol Data Unit (PDU) is the MPC term for a logical piece of data to be read or written. A PDU consists of protocol headers, likely followed by application data. Protocol headers include the IP header, the HPR NHDR/THDR, the SNA transmission header (TH), and the SNA request header (RH).

The HPDT device driver compares the size of each outbound PDU to the packing size limit. If the PDU size is less than or equal, and other criteria are met, the PDU is physically moved to and transmitted from a packing buffer. If the PDU size is larger, the PDU will be transmitted in the traditional manner using Indirect-Data-Address Word (IDAW) relocation.

HPDT packing is a compromise between an increase in storage and CPU consumption in order to increase throughput by improving channel utilization. In the host-to-router or host-to-channel extender configuration, where bottlenecks may be the channel bandwidth or adjacent link station capacity, it is likely the benefit of the increase in throughput would exceed the cost of additional host storage and CPU. In other configurations, the cost of the packing buffers or CPU resource will exceed the throughput benefits.

It is recommended HPDT packing remain disabled for the following conditions:

1. On systems that are storage or CPU constrained.
2. When the majority of the PDUs are ineligible (for example, TCP/IP traffic), which will cause the packing buffers to be allocated but under used.
3. For MPC groups containing multiple write devices. The storage cost could be excessive and therefore likely outweigh the throughput benefit.

If none of the above conditions is a concern, and HPDT packing appears beneficial, the following steps are recommendations:

1. Adjust the write data segment size such that it's equal to one of the CSM data space pool sizes (4K, 16K, 32K, or 60K). For more information, see the information about channel connections between APPN nodes in the Communications Server: SNA Network Implementation Guide.
2. Review the SNA RU size (logmode entries) for native SNA traffic and the TCP/IP MTU size for Enterprise Extender traffic.

If max_pdu_size is specified within the range, and MPCLEVEL is not HPDT or the connection is not point-to-point, then the value is accepted but ignored.

PORTNAME

```
PORTNAME -- (port_name, link_num)
```

statements: TRLE

dependency: OSA port operating in either ATM native mode or in QDIO mode.

Specify a link_num value only for an OSA-Express port that is operating in QDIO mode and customized for LAN emulation. The link_num value is mutually exclusive with the PORTNUM operand of the TRLE definition statement.

port_name

Specifies the Open Systems Adapter (OSA) port name. The name must be unique in one logical partition and it must be unique for an OSA CHPID that is shared by multiple logical partitions. Some OSA-Express3 or later features provide support for multiple ports per CHPID. The port name assigned to each port on a specific CHPID must also be unique. In general, each defined port name should be unique within your network. The maximum length for the port_name value is 8 characters.

For an OSA port customized for HPDT ATM native mode, the port_name value must match the port name specified when OSA is customized through OSA/SF. If this name is used by TCP/IP, it must also be defined as the port name in the TCP/IP profile DEVICE statement. If this name is used for SNA HPR over native ATM connections, it must also be defined as the port name in the PORT statement in a VTAM XCA major node.

For an Open Systems Adapter-Express (OSA-Express) port operating in QDIO mode, the name specified for the port_name value is downloaded into the OSA-Express when the first TCP/IP device for this OSA-Express port is activated. For OSA-Express QDIO connections, this name must be defined as the device name in the TCP/IP profile DEVICE statement or as the port name in the TCP/IP profile INTERFACE statement for any TCP/IP stack that will use this OSA-Express feature. This name must also be defined as the port name on a VARY TCPIP, OSAENTA command or the port name on an OSAENTA profile statement if a TCP/IP stack will start an OSA-Express network traffic analyzer trace on this OSA feature.

link_num

Specifies the ATM LAN emulation client (LEC) port number 0 or 1 for an OSA-Express port operating in QDIO mode customized for LAN emulation. The default is 0. For each OSA ATM port in QDIO mode, two LAN Emulation Clients (LEC) can be customized through OSA/SF. Each LEC represents one logical OSA port, and therefore each must be represented by a different TRLE with unique PORTNAME and unique LEC port number. If two LECs are customized, both LECs must operate in QDIO mode.
PORTNUM

statements:
TRLE

dependency:
Specify a `port_num` value only for an OSA-Express3 or later port operating in QDIO mode. The PORTNUM operand is mutually exclusive with the `link_num` value specified on the TRLE PORTNAME definition statement.

range:
0-15

Some OSA-Express3 or later features provide support for multiple ports per network interface card (NIC). To access these ports, you must supply a physical port number on the QDIO TRLE definition to distinguish between the physical ports on an OSA-Express3 or later NIC. For additional information, see zEnterprise System and System z10 OSA-Express Customer’s Guide and Reference.

`port_num`
Specifies the physical port number of an OSA-Express3 or later NIC that is to be associated with this QDIO MPC group. The PORTNAME and PORTNUM values together uniquely define the OSA-Express3 or later physical port. The value of the PORTNUM operand is zero-based; the value `port_num` 0 represents the first physical port, `port_num` 1 represents the second port, up to `port_num` 15, which represents the sixteenth physical port. The default is 0.

Rule: Although the PORTNUM operand supports up to the value 15 (that is, 16 physical ports), do not code a value higher than what the OSA-Express3 or later NIC supports. If the `port_num` value is higher than the physical capabilities of the OSA-Express3 or later NIC, failures will occur when QDIO is activated.

READ
Specifies the subchannel address used to read data from the adjacent APPN node or Open Systems Adapter port. You must code at least one READ subchannel.

For an OSA-Express feature, it specifies the subchannel address used to read MPC or ULP specific control data. You must code exactly one READ subchannel.

Subchannel addresses can be defined as a single address, as a range of addresses, or both. A range is indicated by a hyphen(-). VTAM generates addresses for the range. For example, READ=(132,128-130,13A) defines the addresses 132, 128, 129, 12A, 12B, 12C, 12D, 12E, 12F, 130, and 13A.

For connections to an adjacent APPN host node:

each address on the READ operand, code the corresponding address on the WRITE operand in the adjacent host to provide a complete path.

For connections to an Open Systems Adapter port or an OSA-Express port:
you can specify only one subchannel address and that address must be an even number that is one less than the address specified on the corresponding WRITE operand. For example, if you specify 500 on the READ operand, you must specify 501 on the WRITE operand.

If this Open Systems Adapter port is for native ATM access, the last two digits of the subchannel address you specify on the READ operand must match the Even Unit Address as it is defined during Open Systems Adapter configuration on the ATM Native Support Settings panel, which is part of the OSA Configuration pull down option accessed through the OSA/SF interface.

If this Open Systems Adapter port is for an OSA-Express feature, there is no matching configuration required.

For an adjacent APPN node or an Open Systems Adapter port, do not code the same subchannel address for the READ operand and the WRITE operand in the same host.

Note: All READ and WRITE channels that you configure must have similar physical characteristics. If some channels take different physical paths with longer distances, this might result in data arriving out of order on the partner host. For connections to an adjacent APPN node, and MPCLEVEL=HPDT is specified, HPR is required to resequence the inbound packets that they arrive out of order. HPR performs best when packets arrive with consistent times and data arrives in order. If the physical paths of the underlying HPDT MPC connection are vastly different, this might lead to excessive retransmissions from HPR. The excessive retransmissions might lead to performance degradation.

REPLYTO
REPLYTO = timeout_value

Specifies how long VTAM waits for completion of a multipath channel XID I/O operation after starting a channel program. If this timeout expires, a message is written to inform the operator that a timeout has occurred. After the XID completes, REPLYTO has no meaning.

Specify timeout_value in seconds or seconds and tenths of a second. A value of 3.0 seconds should be adequate, unless the attached adjacent APPN node or Open Systems Adapter port is disabled for long intervals.

For connections to an OSA-Express feature, the REPLYTO parameter only applies to the control channels.

STORAGE

STORAGE = DS
STORAGE = ECSA

Specifies whether CSM ECSA or CSM data space is used for the multipath channel's inbound I/O storage.

This operand is valid only when MPCLEVEL=HPDT is specified.

For connections to an OSA-Express feature, the STORAGE parameter does not apply.

Users: application attempts to receive the data using OPTCD=XBUFLST (HPDT receive), VTAM moves the data once if the application's preference for incoming data does not match this STORAGE value. The LU 6.2 application can indicate its preference for ECSA, data space, or both by using the XBUFLST-Receive vector (ISTAPC82), which is located in the VTAM-APPCCMD vector list (ISTAPCVL). See z/OS Communications Server: SNA Programming for more information.

STORAGE=DS
Indicates that the data segment of the inbound buffer area is to reside in CSM data space pages. A 64-bit backed CSM data space is used.

STORAGE=ECSA
Indicates that the data segment of the inbound buffer area is to reside in CSM ECSA pages. This option is recommended only if you have many HPDT applications requesting ECSA storage for incoming data traffic.
TYPE

TYPE = TRL

statements: VBUILD

Specifies that the VBUILD statement defines a TRL major node to VTAM.

WRITE

WRITE = channel_unit_address

(statements: TRLE

range: X'000'–X'FFFF'

Specifies the subchannel address used to write data to the adjacent APPN node or Open Systems Adapter port. Code at least one WRITE subchannel address.

For an OSA-Express feature, it specifies the subchannel address used to write MPC or ULP specific control data. Code exactly one WRITE subchannel address.

Subchannel addresses can be defined as a single address, as a range of addresses, or both. A range is indicated by a hyphen(-). VTAM generates addresses for the range. For example, WRITE=(132,128-130,13A) defines the addresses 132, 128, 129, 12A, 12B, 12C, 12D, 12E, 12F, 130, and 13A.

For connections to an adjacent APPN host node:

For each address on the WRITE operand, code the corresponding address on the READ operand in the adjacent host to provide a complete path.

The WRITE subchannels are used in order, so certain subchannels can be given a higher priority by coding them first. For example, you might want to code subchannels with a higher bandwidth first.

For connections to an Open Systems Adapter or an OSA-Express feature:

You can specify only one subchannel address and that address must be an odd number that is one greater than the address specified on the corresponding READ operand. For example, if you specify 500 on the READ operand, you must specify 501 on the WRITE operand.

For an adjacent APPN node or an Open Systems Adapter port, do not code the same subchannel address for the READ operand and the WRITE operand in the same host.

For an adjacent APPN node or an Open Systems Adapter port, the WRITE subchannels are used in order, so certain subchannels can be given a higher priority by coding them first. For example, you might want to code subchannels with a higher bandwidth first.
Note: All READ and WRITE channels that you configure must have similar physical characteristics. If some channels take different physical paths with longer distances, this might result in data arriving out of order on the partner host. For connections to an adjacent APPN node, and MPCLEVEL=HPDT is specified, HPR is required to resequence the inbound packets that they arrive out of order. HPR performs best when packets arrive with consistent times and data arrives in order. If the physical paths of the underlying HPDT MPC connection are vastly different, this might lead to excessive retransmissions from HPR. The excessive retransmissions might lead to performance degradation.
Chapter 3. Routing and dynamic reconfiguration

This topic describes the definition statements that are used to define the following functions:

- VTAM routes
- Dynamic path update
- Dynamic reconfiguration and dynamic change
- Coding Notes
- Adjacent SSCP tables
- Adjacent cluster routing definitions
- APPN transmission group profiles
- APPN-to-subarea CoS mapping
- Subarea-to-APPN CoS mapping
- Border node CoS mapping
- Network node server lists
- Generic resources preference tables

VTAM routes

PATH definition statements are representations of the routes VTAM takes to communicate with other subarea nodes. One or more PATH definition statements are filed in the VTAM definition library. The name assigned to a path definition set is used to activate that set of paths. More than one path definition set can be filed and activated. Note that the VBUILD definition statement is not used.

Communication with other subarea nodes is not possible unless all the required path tables are active. Therefore, you should activate any required path tables before you activate any NCP or channel-attachment major nodes.

Note that, unlike NCP, VTAM requires you to code a reverse virtual route in the VTAM that receives an activate virtual route command because VTAM cannot dynamically allocate a virtual route when it receives the command.

VTAM routes coding notes

This information contains coding notes for VTAM routes.

PATH definition statement

Code a PATH definition statement to define routes.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>PATH</td>
<td>DESTSA</td>
<td>679</td>
<td>R</td>
<td></td>
<td></td>
<td>MXSUBNUM</td>
</tr>
<tr>
<td></td>
<td>DELETER</td>
<td>678</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>see description</td>
</tr>
<tr>
<td></td>
<td>ER0–ER15</td>
<td>679</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VR0–VR7</td>
<td>680</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VRPWS00–72</td>
<td>681</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 2000, 2015
Table 36. Definition statement for routes (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R  Required operand.

VTAM routes full syntax

The full syntax for VTAM routes follows. The syntax for each operand is repeated in "VTAM routes operand descriptions."

**PATH**

```
  name
  PATH DESTSA = subarea_number
  (subarea_number)
```

**DELETER**

```
  explicit_route_number
  (explicit_route_number)
```

**ER**

```
  (adjsub, tgn)
  VR n = er_number
```

**VRPWS**

```
  nn = (minimum_size, maximum_size)
```

VTAM routes operand descriptions

The following information describes the operands you can code for VTAM routes.

**DELETER**
DELETER = explicit_route_number (explicit_route_number)

statement: PATH
dependencies: see note

explicit_route_number is the explicit route number for the route to be deleted. The value of explicit_route_number can be ER0–ER15. Up to 16 explicit route numbers can be specified in DELETER.

VTAM deletes the specified explicit route from its routing table if the explicit route is not currently operative. If the requested explicit route is currently operative, the request is rejected and a warning message is displayed.

Note: Use extreme caution when coding this operand. Deleting one explicit route might make several routes unusable because multiple origin subareas can use the same routing table entry. Use the DELETER operand only when there is no other method of correcting errors or resolving problems. If the explicit route must be redefined, you should use the add or replace function.

DESTSA

DESTSA = subarea_number (subarea_number)

statement: PATH
dependencies: MXSUBNUM

Specifies the destination subarea number for the routes being defined to subarea nodes in your network. You can code more than one destination subarea as long as the route definition statement holds for each one.

The address of the destination subarea must not exceed the subarea addressing supported by the SSCP (as defined by the MXSUBNUM start option).

ER0-ER15
Names the adjacent subarea and, optionally, the transmission group (TG) number for the associated explicit routes originating in the host and leading to the destination subarea.

At least one ER must be specified on the PATH definition statement. You can change the ER value when the ER is deactivated or undefined.

\textbf{adjsub}

The subarea number (in decimal) of the adjacent subarea to which traffic should be sent from the host subarea for the explicit route being defined.

\textbf{tgn}

An optional transmission group number for the explicit route being defined.

If you do not specify a value, the default is 1.

\textbf{name}

Provides the optional name of the PATH represented by this definition statement.

\textit{name} is not checked by VTAM for validity, but its use is strongly recommended. Its purpose is to point out the particular PATH definition statement if a definition error or warning message occurs during activation.

\textbf{VR0-VR7}
Associates the virtual route with the explicit route.

\( er\_number \) is the explicit route number to which the virtual route is mapped. You can change VR to ER mapping after initial activation of a path definition when the VR is deactivated or undefined.

At least one VR must be specified on the PATH definition statement.

**VRPWS00-VRPWS72**

Specifies the pacing window size for a certain virtual route and transmission priority. The virtual route must be defined within this path definition. You can change the pacing window size after initial activation of a path definition when the VR is deactivated or undefined.

Virtual route pacing window sizes are taken from the PATH definition statements in the node that activates the virtual route. For instance, you could define a virtual route pacing window size for a route, but if that route is activated by another node, such as an NCP, the pacing window size is taken from the NCP definition, whether a specified value or a default.

The first number in the operand identifies the virtual route number; the second number identifies the transmission priority associated with the virtual route. For example, VRPWS02=(128,255) defines a minimum virtual route window size of 128 and a maximum window size of 255 for VR0, transmission priority 2.

The minimum window size must be greater than zero, and less than or equal to the maximum window size. The maximum window size must be greater than or equal to the minimum window size, and less than or equal to 255. If either value is not valid, VRPWS is ignored.

If you do not code this operand, VTAM sets the minimum window size equal to the ER length and the maximum window size equal to three times the ER length. However, if the virtual route ends in a subarea that is adjacent to VTAM, the maximum window size is set to the larger of the following two values:

- 15
- 255 minus 16\( n \), where \( n \) is the number of explicit routes (defined or operative) that originate in the host and pass through, but do not end in, the adjacent subarea.
This increases the maximum window size for a route to a channel-attached NCP that has only a few explicit routes passing through it.

For additional information about virtual route window sizes, see z/OS Communications Server: SNA Network Implementation Guide.

---

**Dynamic path update**

Dynamic path update enables the VTAM operator to add or replace NCP path definitions without regenerating and reloading the NCP if the explicit route (ER) is not currently operative. In addition, a delete function for inoperative ERs is provided for both VTAM and NCP path definitions. Dynamic path update can also be used to change path definition operands, such as VR pacing window size, for inoperative routes.

VTAM distinguishes path definition sets from dynamic path update members based on the first noncomment statement in the member. If the first noncomment statement is a PATH definition statement, it is considered a path definition set. If the first noncomment statement is a VPATH or NCPPATH definition statement, it is considered a dynamic path update member.

See z/OS Communications Server: SNA Network Implementation Guide for examples of dynamic path update and for a discussion of restrictions that apply to path definition sets and dynamic path update members.

---

**Dynamic path update (NCP)**

To define a dynamic path specification set for an NCP, code NCPPATH and PATH definition statements.

---

### Table 37. Definition statements for dynamic update NCP paths

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>ncpname</td>
<td>NCPPATH</td>
<td>NETID</td>
<td>686</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PATH</td>
<td>DESTSA</td>
<td>684</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DELETER</td>
<td>685</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ER0–ER15</td>
<td>684</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VR0–VR7</td>
<td>687</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VRPWS00–72</td>
<td>687</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

- **R** Required operand.

---

**Dynamic path update (NCP) coding notes**

This information contains coding notes for dynamic path update (NCP).

**NCPPATH definition statement**

Code the NCPPATH definition statement to identify the NCP and the network to which the path update specification set applies in a dynamic path update member. This definition statement is required, and must precede the path update specifications for each network supported by the NCP.
**PATH definition statement**
Code a PATH definition statement to define routes.

**Dynamic path update (NCP) full syntax**
The full syntax for dynamic update NCP paths follows. The syntax for each operand is repeated in the "Dynamic path update (NCP) operand descriptions."

**NCPPATH**

```plaintext
ncpname NCPPATH NETID = network_id
```

**PATH**

```plaintext
name PATH BESTSA = (subarea_number)
```

```plaintext
DELETER = (explicit_route_number)
```

```plaintext
ER n = (adjsub, tgn, lo, med, hi, tot)
```

```plaintext
VR n = vr_number VRPMS en = (minimum_size, maximum_size)
```

**Dynamic path update (NCP) operand descriptions**
The following information describes the operands you can code for dynamic update NCP paths.

**DELETER**
DELETER is the explicit route number for the route to be deleted. The value of `explicit_route_number` can be ER0–ER15. Up to 16 explicit route numbers can be specified in DELETER.

The NCP deletes the specified explicit route from its routing table if the explicit route is not currently operative. If the requested explicit route is currently operative, the request is rejected and a warning message is displayed.

Note:
1. Use extreme caution when coding this operand. Deleting one explicit route might make several routes unusable because multiple origin subareas can use the same routing table entry.
2. Use the DELETER operand only when there is no other method of correcting errors or resolving problems. If the explicit route must be redefined, you should use the replace function of dynamic path update.

DESTSA specifies the destination subarea number for the routes being defined to subarea nodes in your network. You can code more than one destination subarea as long as the route definition statement holds for each one.

ER0-ER15
Names the adjacent subarea and optionally specifies:

- The transmission group (TG) number for the associated explicit routes originating in the host and leading to the destination subarea.
- The NCP’s transmission group flow-control thresholds. The maximum value for any threshold is 4194304.

NCP maximum values for transmission group flow-control thresholds differ depending on NCP release. The maximum for NCP V6R2 and earlier is 524288. The maximum for NCP V6R3 and later is 4194304. Do not use a value greater than 524288 when doing NCP dynamic path updates unless the NCP release is V6R3 or later.

At least one ER must be specified on the PATH definition statement. You can change the ER value when the ER is deactivated or undefined.

- **adjsub**
  Specifies the subarea number (in decimal) of the adjacent subarea to which traffic should be sent from the host subarea for the explicit route being defined.

- **tgn**
  Specifies an optional transmission group number for the explicit route being defined.

- **lo**
  Specifies the NCP’s transmission group flow-control threshold for low priorities.

- **med**
  Specifies the NCP’s transmission group flow-control threshold for medium priorities.

- **hi**
  Specifies the NCP’s transmission group flow-control threshold for high priorities.

- **tot**
  Specifies the NCP’s transmission group flow-control threshold for the total traffic in characters.
Provides the optional name of the PATH represented by this definition statement.

$name$ is not checked by VTAM for validity, but its use is recommended. Its purpose is to point out the particular PATH statement if a definition error or warning message occurs during activation.

**ncpname**

Specifies a 1–7 character name of the NCP to which the path update specification set applies. $ncpname$ is required and must be identical to the PU name of the NCP.

If the specified NCP is not currently active or is being activated when the dynamic path update member is activated, VTAM issues a warning message and discards the path update specification set. The PATH definition statements that follow, however, are checked for validity, and warning messages are issued if there are errors.

**NETID**
**NETID—network_id**

**statement:** NCPPATH
**dependencies:** none

Specifies a 1–8 character network identifier to which the path update specification set applies. This operand is required. If you do not code it, VTAM ignores the entire path update specification set and issues a warning message.

**Note:** The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

If you do not define the specified NETID in the NCP, the NCP rejects the update request and sends an indicator to the SSCP that the NETID is not valid. This causes VTAM to issue a warning message.

**VR0-VR7**

**statement:** PATH
**dependencies:** none
**range:** 0–15

Associates the virtual route with the explicit route.

*er_number* specifies the explicit route number to which the virtual route is mapped. You can change VR to ER mapping after initial activation of a path definition when the VR is deactivated or undefined.

At least one VR must be specified on the PATH definition statement.

**VRPWS00-VRPWS72**
statement: \[ \text{VRPWS}_{nn} = \{ \text{minimum\_size}, \text{maximum\_size} \} \]

dependencies: none

Specifies the pacing window size for a certain virtual route and transmission priority. The virtual route must be defined within this path definition. You can change the pacing window size after initial activation of a path definition when the VR is deactivated or undefined.

The first number in the operand identifies the virtual route number; the second number identifies the transmission priority associated with the virtual route. For example, VRPWS02 = (128,255) defines a minimum virtual route window size of 128 and a maximum window size of 255 for VR0, transmission priority 2.

The minimum window size must be greater than zero and less than or equal to the maximum window size. The maximum window size must be greater than or equal to the minimum window size and less than or equal to 255. If either value is not valid, VRPWS is ignored.

VTAM sends the minimum and the maximum values specified with this operand to the NCP. If this operand is not coded, no window sizes are sent, and NCP uses already defined values, if they exist. If the window sizes are not defined for a VR when the VR is activated, NCP dynamically calculates the value at the VR activation time.

The value \((0,0)\) has special meaning. If this value is coded, the currently specified values in the NCP, if any, are nullified and values dynamically calculated at VR activation time are used.

For additional information on virtual route window sizes, see z/OS Communications Server: SNA Network Implementation Guide.

### Dynamic path update (VTAM)

To define a dynamic path specification set for VTAM, code VPATH and PATH definition statements.

#### Table 38. Definition statements for dynamic update VTAM paths

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>sscpname</td>
<td>VPATH</td>
<td>NETID</td>
<td>692</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PATH</td>
<td>DESTSA</td>
<td>690</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DELETER</td>
<td>690</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ER0–ER15</td>
<td>691</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>VR0–VR7</td>
<td>693</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>VRPWS00–72</td>
<td>693</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R Required operand.
Dynamic path update (VTAM) coding notes

The operands on this PATH definition statement are the same as those on "VTAM routes" on page 677, with one exception. If the SSCPNAME on the VPATH definition statement matches VTAM's, the values of the destination subarea and adjacent subareas used in the PATH definition statement must not exceed the subarea addressing supported by the SSCP (as defined by the MXSUBNUM start option).

Code the VPATH definition statement to identify the VTAM subarea and the network to which the path update specification set applies in a dynamic path update member. This definition statement is required and must precede the path update specifications for each subarea. The VPATH definition statement is not allowed in the path definition set.

Dynamic path update (VTAM) full syntax

The full syntax for dynamic update VTAM paths follows. The syntax for each operand is repeated in "Dynamic path update (VTAM) operand descriptions" on page 690.

VPATH

```
sscpname—VPATH—NETID=network_id
```

PATH
Dynamic path update (VTAM) operand descriptions

The following information describes the operands you can code for dynamic update VTAM paths.

**DELETER**

```plaintext
statement:
dependencies: see notes
```

Specifies the explicit route number for the route to be deleted. The value of `explicit_route_number` can be ER0–ER15. Up to 16 explicit route numbers can be specified in DELETER.

The NCP deletes the specified explicit route from its routing table if the explicit route is not currently operative. If the requested explicit route is currently operative, the request is rejected and a warning message is displayed.

**Note:**

1. Use extreme caution when coding this operand. Deleting one explicit route might make several routes unusable because multiple origin subareas can use the same routing table entry.
2. Use the DELETER operand only when there is no other method of correcting errors or resolving problems. If the explicit route must be redefined, you should use the replace function of dynamic path update.

**DESTSA**
Specifies the destination subarea number for the routes being defined to subarea nodes in your network. You can code more than one destination subarea as long as the route definition statement holds for each one.

**ER0-ER15**

Names the adjacent subarea and optionally specifies the transmission group (TG) number for the associated explicit routes originating in the host and leading to the destination subarea.

At least one ER must be specified on the PATH definition statement. You can change the ER value when the ER is deactivated or undefined.

- **adjsub**
  Specifies the subarea number (in decimal) of the adjacent subarea to which traffic should be sent from the host subarea for the explicit route being defined.

- **tgn**
  Specifies an optional transmission group number for the explicit route being defined.
statement:                         PATH
dependencies:                   none

Provides the optional name of the PATH represented by this definition statement.

$name$ is not checked by VTAM for validity, but its use is recommended. Its purpose is to point out the particular PATH statement if a definition error or warning message occurs during activation.

**NETID**

statement:                         VPATH
dependencies:                   none

Specifies a 1–8 character network identifier to which the path update specification set applies. This operand is required.

**Note:** The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

The specified NETID must be identical with the NETID specified in the VTAM START command; otherwise, VTAM ignores the path update specification set.

**sscpname**
**sscpname**

**statement:** PATH
**dependencies:** SSCPNAME start option

Specifies a 1–8 character name of the VTAM subarea to which the path update specification set applies. `sscpname` is required. `sscpname` must be identical with the SSCPNAME specified in the start option currently in use; otherwise, VTAM ignores the path update specification set.

**VR0-VR7**

**VRn=er_number**

**statement:** PATH
**dependencies:** none
**range:** 0–15

Associates the virtual route with the explicit route.

`er_number` specifies the explicit route number to which the virtual route is mapped. You can change VR to ER mapping after initial activation of a path definition when the VR is deactivated or undefined.

At least one VR must be specified on the PATH definition statement.

**VRPWS00-VRPWS72**
VRPWS statement:

```
VRPWS nn = (minimum_size, maximum_size)
```

Specifies the pacing window size for a certain virtual route and transmission priority. The virtual route must be defined within this path definition. You can change the pacing window size after initial activation of a path definition when the VR is deactivated or undefined.

The first number in the operand identifies the virtual route number; the second number identifies the transmission priority associated with the virtual route. For example, `VRPWS02 = (128,255)` defines a minimum virtual route window size of 128 and a maximum window size of 255 for VR0, transmission priority 2.

The minimum window size must be greater than zero and less than or equal to the maximum window size. The maximum window size must be greater than or equal to the minimum window size and less than or equal to 255. If either value is not valid, VRPWS is ignored.

VTAM sends the minimum and the maximum values specified with this operand to the NCP. If this operand is not coded, no window sizes are sent, and NCP uses already defined values, if they exist. If the window sizes are not defined for a VR when the VR is activated, NCP dynamically calculates the value at the VR activation time.

The value (0,0) has special meaning. If this value is coded, the currently specified values in the NCP, if any, are nullified and values dynamically calculated at VR activation time are used.


**Dynamic reconfiguration and dynamic change**

Dynamic reconfiguration (DR) allows you to change your network configuration without deactivating the affected major node. This means, for example, that you can change resources on an NCP without deactivating the NCP.

There are several methods of dynamic reconfiguration; this document describes how to add, delete, and move resources using a DR definition statements. To implement the changes, use the VARY DRDS command to activate the DR file. For more information on the status requirements for resources before they are dynamically reconfigured and how to implement dynamic reconfiguration, see [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/en/com.ibm.zos.v2r1.aix.5725-002/zos/aix/snmrea000.htm).

The following information shows the formats of the definition statements that are coded in a DR file for use with the VARY DRDS command. The statements must follow the rules described in "Format of definition statements" on page 8. File the completed statements in the VTAM definition library.
Add operation

For an ADD operation, code the following definition statements:

VBUILD
ADD
PU
LU

VBUILD definition statement
See “VBUILD definition statement” on page 697 for details on the VBUILD definition statement.

ADD definition statement
Code an ADD definition statement to indicate the adding of a resource.

PU and LU definition statements
Code one PU and LU statement for each resource to be added.

- Code the operands in the PU and LU definition statements in the same format as they appear in the appropriate major node.
- Code all of the operands that you want from the GROUP, LINE, or PU definition statements to which you are adding the physical unit or logical unit. VTAM ignores NCP-only operands and might issue a warning message.
- Sifting takes place within the hierarchy of minor nodes being added dynamically. Values for operands coded in the original hierarchy above the hierarchy being added do not sift down to the added resources. Therefore, if you want the values coded in the original hierarchy, you must code them in the group of DR definition statements for the resources you are adding.
- The most efficient way to add a physical unit and its associated logical units is to specify them on the same ADD definition statement and not use separate ADD definition statements for the physical unit and each logical unit.

Table 39. Definition statements for dynamic reconfiguration with VARY DRDS (ADD operation)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=DR</td>
<td>702</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>ADD</td>
<td>TO</td>
<td>702</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DRTYPE</td>
<td>700</td>
<td></td>
<td></td>
<td></td>
<td>NCP</td>
<td>Adding LUs to PUs only</td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>LU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R Required operand.

Delete operation

For a delete operation, code the following definition statements:

VBUILD
DELETE
PU
LU

**VBUILD definition statement**
See “VBUILD definition statement” on page 697 for details on the VBUILD definition statement.

**DELETE definition statement**
Code a DELETE definition statement to indicate the deletion of a resource.

**PU and LU definition statements**
Code the PU and LU statement for the resource to be deleted. Code only the PU or LU names. You do not have to code individual PU or LU operands.

*Note:* When you delete a physical unit, you also delete its associated logical units. Do not code the associated LU definition statements.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=DR</td>
<td>702</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>DELETE</td>
<td>FROM</td>
<td>700</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>PU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>LU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Legend:*

| R   | Required operand. |

**Move operation**
For a move operation, code the following definition statements:

```
VBUILD
MOVE
PU
```

The MOVE operation is only for physical units (and their associated logical units). To move an individual logical unit to another physical unit, use a DELETE and then a subsequent ADD. (For efficient storage management, code all DELETEs before their related ADDs.)

A DR MOVE is valid only for NCP PUs.

*Note:*

1. For an SSCP to successfully take over for another SSCP that has issued a DR MOVE, the takeover SSCP must also support DR MOVE.
2. The takeover SSCP must be configured the same as the original host.
**VBUILD definition statement**
See "VBUILD definition statement" for details on the VBUILD definition statement.

**MOVE definition statement**
Code a MOVE statement to indicate the moving of a physical unit from one line to another or to change the SDLC address of a physical unit.

To change the SDLC address for a physical unit on a certain line, use the MOVE definition statement and specify the new address for the ADDR operand, keeping the FROM line and the TO line the same.

**PU definition statement**
Code a PU statement to identify the physical unit you want to move.

Code only the names of the physical units you want to move. Do not code the names of associated logical units. VTAM automatically includes the associated logical units in the move. For each physical unit, you can optionally specify a new station address and whether you want the physical unit and its logical units automatically activated after the move.

**Note:** Values coded for operands on the original GROUP or LINE definition statements do not sift down to operands on the new PU and LU for the resources being moved.

**Table 41. Definition statements for dynamic reconfiguration with VARY DRDS (MOVE operation)**

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name VBUILD</td>
<td>TYPE=DR</td>
<td></td>
<td>702</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name MOVE</td>
<td>FROM</td>
<td></td>
<td>700</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TO</td>
<td></td>
<td>702</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name PU</td>
<td>ACTIVATE</td>
<td></td>
<td>699</td>
<td>NO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ADDR</td>
<td></td>
<td>699</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

| R | Required operand. |

**Coding notes**

This information contains coding notes for the VBUILD definition statement.

**VBUILD definition statement**
Code a VBUILD definition statement as the first statement in a group of dynamic reconfiguration definition statements.

You can file the dynamic reconfiguration definition statements for the different types of operations as one member. If multiple operations are defined, code only one VBUILD definition statement.
Dynamic reconfiguration with VARY DRDS full syntax

The full syntax for dynamic reconfiguration with VARY DRDS follows. The syntax for each operand is repeated in the "Dynamic reconfiguration with VARY DRDS operand descriptions" on page 699.

VBUILD

ADD

DELETE

MOVE
Dynamic reconfiguration with VARY DRDS operand descriptions

The following information describes the operands you can code for dynamic reconfiguration with VARY DRDS.

ACTIVATE operand description

Statements: PU
Dependencies: none

Specifies whether VTAM automatically activates the physical unit and its associated logical units (that have been defined with ISTATUS=ACTIVE) after the physical unit has been moved to a new line.

If you use ACTIVATE=NO, you can activate the physical unit after the move by issuing VARY ACT,ID=pu name.

ADDR operand description
addresses: 2-digit hexadecimal SDLC address
dependencies: none
range: X'01'–X'FE'

Specifies a new 2-digit hexadecimal SDLC address for a physical unit being moved. The new SDLC link station address replaces the physical unit's current link station address. This operand is optional.

To move multiple physical units and change their addresses concurrently, specify a unique address for each PU under the LINE definition statement. The valid address range is X'01'–X'FE'.

**DRTYPE operand description**

`DRTYPE = NCP`, `DRTYPE = LOCAL`

statements: ADD
dependencies: Adding LUs to PUs only

Specifies the major node you are dynamically reconfiguring. Valid values are:

- NCP An NCP major node
- LOCAL A local SNA major node

**FROM operand description**
FROM resource_name

statements: DELETE, MOVE
dependencies: none

resource_name can be any of the following items:

line name
Identifies a line from which you want to delete or move a physical unit.
The physical units attached to this line must be types 1, 2, or 2.1. The line
must be defined as non-switched, and can be active or inactive. This
operand is required if you are deleting or moving a physical unit. The line
from which you are moving the physical unit must be within the same
NCP as the line to which you are moving it.

Note: If the line is under a GROUP that specifies VIRTUAL=YES, dynamic
reconfiguration is not allowed for the line.

PU name
Identifies a physical unit from which you want to delete any associated
logical units. The physical unit must be type 1, type 2, or type 2.1. This
operand is required to identify the physical unit.

Note: If the physical unit is under a GROUP that specifies VIRTUAL=YES,
dynamic reconfiguration is not allowed for the physical unit.
Specifies an optional name for an individual DR statement for error-reporting purposes. If you code the name for a DR definition statement and the statement fails, VTAM includes the name in the appropriate operator message. The use of `name` is strongly recommended.

**TO operand description**

```
name
```

- **statements:** `VBUILD, ADD, DELETE, MOVE`
- **dependencies:** `none`

`resource_name` can be one of the following items:

- **line name**
  Identifies a line to which you want to add or move a physical unit. The physical units attached to this line must be type 1, 2, or 2.1. This line must be defined as non-switched, and can be active or inactive. This operand is required if you are adding or moving a physical unit. The line to which you are moving the physical unit must be within the same NCP as the line from which you are moving it.

  **Note:** If the line is under a GROUP that specifies VIRTUAL=YES, dynamic reconfiguration is not allowed for the line.

- **PU name**
  Identifies a physical unit to which you want to add any associated logical units. The physical unit must be type 1, type 2, or type 2.1. This operand is required to identify the physical unit.

  **Note:** If the physical unit is under a GROUP that specifies VIRTUAL=YES, dynamic reconfiguration is not allowed for the physical unit.

**TYPE operand description**

1. When coding the operands for the PU and LU definition statements, do not code NCP-only operands; otherwise, a warning message will be issued.

2. If you do not code the MAXDATA operand on the PU definition statement in a dynamic reconfiguration definition, a value of 261 bytes is used for a PU type 1 and a value of 265 is used for a PU type 2.
Meaning that this VBUILD definition statement defines a dynamic reconfiguration file to VTAM. This operand is required.

**Adjacent SSCP table**

The adjacent SSCP table contains lists of SSCPs that can be in session with the VTAM SSCP and can be used to reach destination SSCPs in the same or other networks. One or more adjacent SSCP tables can be activated in each VTAM host. File the definitions in VTAMLST files (or as members in the SYS1.VTAMLST data set).

With the dynamic adjacent SSCP table function, you are not required to code adjacent SSCP tables to establish cross-domain and cross-network sessions. With this function, VTAM dynamically routes session establishment requests to all active adjacent SSCPs until the correct SSCP is found. However, if you do code adjacent SSCP tables, VTAM uses them.

**Table 42. Definition statements for ADJSSCP table**

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=ADJSSCP</td>
<td>713</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>NETWORK</td>
<td></td>
<td>710</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>NETID</td>
<td></td>
<td></td>
<td>STARTOPT</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SORDER</td>
<td>710</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SSCPORD</td>
<td>711</td>
<td></td>
<td>STARTOPT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>listname</td>
<td>ADJLIST</td>
<td></td>
<td>709</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sscpname</td>
<td>CDRM</td>
<td>SORDER</td>
<td>710</td>
<td>N</td>
<td></td>
<td>STARTOPT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SSCPORD</td>
<td>711</td>
<td>N</td>
<td></td>
<td>STARTOPT</td>
<td></td>
</tr>
<tr>
<td>cdrmname</td>
<td>ADJCDRM</td>
<td></td>
<td>707</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

- **R** Required operand.
- **N** Code this operand on the higher-level NETWORK definition statement to take advantage of the sift effect, or code it on this definition statement.

**Adjacent SSCP table coding notes**

This information contains coding notes for an adjacent SSCP table.

**VBUILD definition statement**

Code a VBUILD definition statement as the first statement in an adjacent SSCP table.
**NETWORK definition statement**

Code a NETWORK definition statement to name a destination network. If you do not code a NETWORK definition statement, this adjacent SSCP table defines a default SSCP list for either a single-network or a multiple-network environment. In a multiple-network environment, you can use the NETWORK definition statement to provide a specific default SSCP list for a given destination network.

If ADJCDRM definition statements follow a NETWORK definition statement, without intervening CDRM definition statements, they also comprise a list of default SSCPs. A default SSCP list for a single network consists of a VBUILD definition statement with TYPE=ADJSSCP followed by a NETWORK definition statement and a set of ADJCDRM definition statements.

The following rules apply to the placement of the NETWORK definition statement within the adjacent SSCP table definition:

- The NETWORK definition statement can be followed by a NETWORK, CDRM, or ADJCDRM definition statement.
- Consecutive NETWORK definition statements result in the same adjacent SSCP table for each of the networks.
- If a statement that is not valid is encountered within the adjacent SSCP table definition, that statement and any subsequent statements are rejected until another NETWORK definition statement or the end of the definition is encountered.

**Note:** An ADJSSCP table defined with a NETWORK definition statement of the host network is used only when the destination resource is known to reside in the host network. A default table is coded without a NETWORK definition statement.

**CDRM definition statement**

Code one or more CDRM definition statements to restrict the scope of subsequent ADJCDRM statements to resources owned by a specific CDRM or list of CDRMs. One or more CDRM definition statements can follow a NETWORK or ADJCDRM definition statement.

In an SNA interconnected network, the CDRM definition statement names a destination SSCP in the network identified on the preceding NETWORK definition statement.

The following rules apply to the placement of the CDRM statement within the adjacent SSCP table definition:

- A CDRM definition statement can be followed by another CDRM definition statement.
- One or more CDRM definition statements must be followed by an ADJCDRM definition statement.

The CDRM definition statements within the adjacent SSCP table have no required operands, although any operands that are valid on the CDRM definition statement in a CDRM major node are accepted. These operands are ignored when the CDRM definition statement is part of an adjacent SSCP table.

**Note:** You can define adjacent SSCP tables to reach a new interconnected network by adding only a few statements to the existing CDRM definition of that network.
ADJLIST definition statement

Code an ADJLIST definition statement to define the name of a list of adjacent SSCPs. The list of adjacent SSCPs is created from the ADJCDRM definition statements which follow one or more ADJLIST definition statements.

An ADJLIST statement must be immediately followed by either:
- One or more ADJCDRM definition statements
- One or more ADJLIST definition statements

Coding any other definition signifies the end of the ADJLIST definitions. Once the first ADJCDRM definition statement has been encountered immediately after a ADJLIST definition statement, the next occurrence of an ADJLIST definition statement indicates the start of a new ADJLIST. The following example shows two lists that have been coded. LIST1 contains GW1 and GW2. LIST2 contains GW1, GW2, and GW3.

```
VBUILD TYPE=ADJSSCP
LIST1 ADJLIST
GW1 ADJCDRM
GW2 ADJCDRM
LIST2 ADJLIST
GW1 ADJCDRM
GW2 ADJCDRM
GW3 ADJCDRM
```

Multiple lists can be set up to contain identical ADJCDRMs without duplication of effort. Code all ADJLIST definition statements immediately followed by all the ADJCDRMs for which there would be an entry in each list. For example, using the following definition statements GW1, GW2, and GW3 reside in both LIST1 and LIST2.

```
LIST1 ADJLIST
LIST2 ADJLIST
GW1 ADJCDRM
GW2 ADJCDRM
GW3 ADJCDRM
```

**Note:** If an ADJLIST definition statement follows a NETWORK, CDRM, or ADJCDRM definition statement, those definitions are considered ended. This might result in a problem if the ADJLIST definition statement is placed incorrectly and the previous definition has not been completed when encountered.

ADJCDRM definition statement

Code consecutive ADJCDRM definition statements to create a list of default SSCPs.

Consecutive ADJCDRM definition statements after the CDRM definition statements give a list of adjacent SSCP names for the preceding destination SSCPs identified on the CDRM definition statements.

Adjacent SSCP table full syntax

The full syntax for the adjacent SSCP table follows. The syntax for each operand is repeated in "Adjacent SSCP table operand descriptions" on page 707.

VBUILD
**NETWORK**

```
/>[name] VBUILD TYPE ADJSSCP
```

**Notes:**
1. For CDRM, SORDER, and SSCPORD, default to the value coded on the preceding NETWORK statement.

**ADJLIST**
Notes:
1 For CDRM, SORDER, and SSCPORD, default to the value coded on the preceding NETWORK statement.

Adjacent SSCP table operand descriptions
The following information describes the operands you can code for the adjacent SSCP table.

cdrmname
The name of an adjacent SSCP as known to VTAM. It must be the same as the name on the CDRM definition statement that defines the SSCP to VTAM.

Each ADJCDRM definition statement should have a unique cdrmname. Otherwise, an error message is issued. Furthermore, in an SNA interconnected network, the CDRM names must be unique among SSCPs that participate in cross-network sessions.

The following rules apply to the placement of the ADJCDRM definition statement within the adjacent SSCP table definition:

- The ADJCDRM definition statement can be followed by an ADJCDRM, ADJLIST, CDRM, or NETWORK definition statement (or nothing, if it is the end of the table).
- If an ADJCDRM definition statement immediately follows a NETWORK definition statement, it is the default adjacent SSCP table for the specified network.
- If an ADJCDRM definition statement immediately follows the VBUILD definition statement or a NETWORK definition statement without NETID specified, it is the default adjacent SSCP table for all networks.

You can specify ISTAPNCP, instead of an SSCP name, for cdrmname. ISTAPNCP is a generic representation for an APPN CDRM. It can provide you with additional control over the order in which the network is searched.

If you code the SORDER=ADJSSCP start option and code ISTAPNCP in any or all of the user-defined adjacent SSCP tables, VTAM does not search the APPN network until it encounters ISTAPNCP in the adjacent SSCP table.

That is, if you code ISTAPNCP as the fourth CDRM in the user-defined adjacent SSCP table, VTAM does not search the APPN network until after it has searched unsuccessfully in the first three CDRMs in the table. If you code ISTAPNCP first in the adjacent SSCP table, VTAM searches the APPN network immediately.

Note:

1. If SSCPORD=PRIORITY and SORDER=ADJSSCP, and you do not code ISTAPNCP in the user-defined adjacent SSCP table, VTAM does not add ISTAPNCP to the adjacent SSCP routing table, and VTAM does not search the APPN network unless a previous request for this DLU has searched the APPN network. (That is, there is an ISTAPNCP entry in the ISTADSST table queued to the DLU RDTE.)

2. If you code the SORDER=APPN, SORDER=SUBAREA, or SORDER=APPNFRST start options (or let it default to APPN), VTAM ignores the ISTAPNCP entry in the user-defined adjacent SSCP table, because VTAM places ISTAPNCP in the appropriate place in the adjacent SSCP routing table.

3. If you code the SSCPORD=PRIORITY start option (or let it default to PRIORITY), and either SORDER=APPNFRST, SORDER=APPN, or SORDER=SUBAREA, VTAM searches the adjacent SSCP table in the following order. ISTAPNCP is added in one of the following places:
Before step 3a, if SORDER=APPNFRST
After step 3b, if SORDER=APPN
After step 3d, if SORDER=SUBAREA
a. The CDRM that owns the resource (if known)
b. The SSCPs for which the most recent session-initiation attempt succeeded
c. The SSCPs for which no session-initiation attempt has been made
d. The SSCPs for which the last session-initiation attempt failed.

4. If you code the SSCPORD=DEFINED start option, VTAM searches the adjacent
SSCP table in the same order that SSCPs are listed, with the exception that the
CDRM that owns the resource (if known) is searched first.

listname

statements: ADJLIST
dependencies: none

Defines the name of a list of adjacent SSCPs. This name is used by the ADJLIST
operand on the CDRSC definition statement to specify which adjacent SSCPs
should be used for all session setup requests for that resource.

The following names are reserved adjacent list names and may not be used for
listname:
   ALL
   DYNAMIC
   DEFAULT
   RESOURCE

name
Provides the optional name of the adjacent SSCP table represented by this definition statement or identifies this NETWORK definition statement. To avoid confusion, use the same value for both `name` and the NETID operand.

**NETID**

Specifies the optional 1–8 character identifier of the destination network. Omitting NETID defines a default SSCP list for all networks.

**Note:** The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

**SORDER**
Notes:

1. On the CDRM definition statement, SORDER will default to the value coded on the preceding NETWORK statement.

SORDER=APPNFRST

Specifies that VTAM should always try searching the APPN network first, even if the "owning" CDRM was defined or learned.

SORDER=APPN

Specifies that VTAM should try searching the APPN network first, unless the owning CDRM was defined or learned.

SORDER=ADJSSCP

Specifies that VTAM should search the APPN network only if ISTAPNCP has been explicitly coded within the ADJSSCP table that was chosen for this search/session.

SORDER=SUBAREA

Specifies that VTAM should always try searching the APPN network last (after trying all subarea ADJSSCPs).

SORDER=STARTOPT

Specifies that VTAM should use the current value of the SORDER start option for this ADJSSCP table each time it is chosen for searching. This value can be used on CDRM statements to override the sift down value from the preceding NETWORK statement. When coded on a NETWORK statement, this value works the same way as when SORDER= is not coded on the NETWORK statement.

SSCPORD
Notes:
1. On the CDRM definition statement, SSCPORD will default to the value coded on the preceding NETWORK statement.

statements: NETWORK, CDRM
dependencies: none

Specifies whether VTAM searches an adjacent SSCP table in priority order, in the order in which the table is defined, or in the order specified on the SSCPORD start option.

Tip: When coded on a NETWORK statement, the SSCPORD value sifts down to all CDRM statements under that NETWORK statement (unless SSCPORD is overridden on one or more of the CDRM statements).

**SSCPORD=DEFINED**
Specifies that VTAM searches the adjacent SSCP table in the same order that SSCPs are listed, if past session initiation attempts were successful, with the exception that the owner, if identified, is tried first.

**SSCPORD=PRIORITY**
Specifies that VTAM gives preference to SSCPs it uses. The search order is as follows:
1. The CDRM that owns the resource (if identified).
2. The SSCPs for which the most recent session-initiation attempt succeeded.
3. The SSCPs for which no session-initiation attempt has been made.
4. The SSCPs for which the last session-initiation attempt failed.

**SSCPORD=STARTOPT**
Specifies that VTAM should use the current value of the SSCPORD start option for this ADJSSCP table each time it is chosen for searching. This value can be used on CDRM statements to override the sift down value from the preceding NETWORK statement. When coded on a NETWORK statement, this value works the same way as when SSCPORD= is not coded on the NETWORK statement.

**sscpname**
**sscpname**

**statements:** CDRM
**dependencies:** must match SSCPNAME start option

Provides the name of the SSCP. It must match the name assigned to that SSCP in its own network by the SSCPNAME start option.

The sscpname field on each CDRM definition statement names a destination SSCP in the network identified by the preceding NETWORK definition statement, or, if there is no NETWORK definition statement, in the network where the table is defined.

**TYPE**

**statements:** VBUILD
**dependencies:** none

Specifies that this VBUILD definition statement defines an adjacent SSCP table.

---

**Adjacent cluster routing definitions**

The adjacent cluster (ADJCLUST) routing list allows you to define which adjacent APPN subnetworks VTAM should search. For each destination network, you can define a table that specifies the control point (CP) name of the adjacent nodes to which the search is sent.

Each routing list contains one or more network identifiers, followed by the CPNAMEs of the nodes that should be searched when VTAM receives a request to search for a resource with that network identifier. Coding a NETWORK statement without a NETID value creates a default routing list. The operands coded on the default list are used for all NETIDs that do not have an explicitly coded NETID list.

Figure 1 on page 714 contains the following three routing lists:

- A default routing list to be used when one of the following conditions is met.
  - A non-network qualified request is received.
  - A network qualified request is received and the NETID specified is not defined in any NETWORK statement.
- A list (NETAC) to be used when NETA or NETC is the network identifier specified on a request.
- A list (NETZ) to be used when NETZ is the network identifier and nodes are to be omitted.
### Table 43. Definition statements for ADJCLUST table

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=ADJCLUST</td>
<td>719</td>
<td>R</td>
<td></td>
<td>BN=YES</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>NETWORK</td>
<td>BNDYN</td>
<td>716</td>
<td></td>
<td>BNDYN start option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>NETID</td>
<td>718</td>
<td></td>
<td>BNORD start option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>NEXTCP</td>
<td>CPNAME</td>
<td>717</td>
<td></td>
<td>This node's cpname</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SNVC</td>
<td>719</td>
<td>N</td>
<td>SNVC start option</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>OMITTCP</td>
<td>718</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Legend:

- **R**: Required operand.
- **N**: Code this operand on the higher-level NETWORK definition statement to take advantage of the sift effect, or code it on this definition statement.

### Adjacent cluster routing definitions coding notes

This information contains coding notes for adjacent cluster routing definitions.
**VBUILD definition statement**
Code a VBUILD definition statement as the first statement in an adjacent cluster definition list.

**NETWORK definition statement**
Code a NETWORK definition statement for each NETID or group of NETIDs that you want to define routing.

**NEXTCP definition statement**
Code a NEXTCP definition statement for each node that is to be searched.

**Adjacent cluster routing definitions full syntax**
The full syntax for the adjacent cluster routing definitions follows. The syntax for each operand is repeated in "Adjacent cluster routing definitions operand descriptions" on page 716.

**VBUILD**

```
name  VBUILD TYPE = ADJCLUST
```

**NETWORK**

```
name  NETWORK BNDYN = BNDYN_start_option_value
       BNDYN = FULL  LIMITED  NONE
       BNORD = BNORD_start_option_value
       BNORD = PRIORITY  DEFINED
       NETID = network_id
```

**NEXTCP**
Adjacent cluster routing definitions operand descriptions

The following information describes the operands you can code for the adjacent cluster routing definition statements.

**BNDYN**

- **BNDYN=NONE**
  No nodes are added except those explicitly identified by NEXTCP definition statements.

- **BNDYN=LIMITED**
  All active border nodes in the native subnetwork as well as active adjacent nodes in nonnative subnetworks attached to this node are automatically added to the routing list if either of the following conditions is met:
  - BNs and nonnative NNs that VTAM learns about whose network ID matches the network ID of the destination LU.
  - BN or nonnative NNs that have previously sent a Locate that carried the network ID of the destination LU.

- **BNDYN=FULL**
  All active border nodes in the native subnetwork as well as active adjacent nodes in nonnative subnetworks attached to this node are automatically added to the routing list.

**BNORD**
Specifies the order in which VTAM performs cross-subnet searches.

**BNORD=DEFINED**

Specifies that searches are performed in the order that you define border nodes and non-native network nodes.

**Tip:** The value BNORD=DEFINED applies only to defined routing entries. Dynamic entries are reordered in priority order and placed after all defined entries.

**BNORD=PRIORITY**

Specifies that search preference is given to nodes for which the most recent search was successful and to nodes whose NETID matches the DLU's NETID value.

**CPNAME**

Specifies the CP name of the node to which the search request is to be forwarded.

The CP name may be network qualified. If you specify a `control_point_name`, but not a `netid`, VTAM uses this node's NETID. If you do not code CPNAME, VTAM uses this node's network qualified CP name.

The CP names of the following nodes can be listed when defining ADJCLUST definitions:
- Extended border nodes in this node's topology subnetwork, which own subnetwork boundaries with other APPN subnetworks.
- Nodes in adjacent nonnative subnetwork with which this node has a subnetwork boundary.
- This node's CPNAME which indicates that this node's native subnetwork should be searched.
Assigns a name to the definition statement.

To avoid confusion, use the same value for name and the following information:
- CPNAME when the NEXTCP definition statement is coded
- NETID when the NETID definition statement is coded

**NETID**

Specifies the optional 1–8 character identifier of the destination network.

If you do not code NETID at all, this NETWORK definition statement represents an adjacent cluster routing definition to be used for unknown or unlisted NETIDs (those not explicitly coded on other NETWORK statements).

Do not code the same NETID multiple times within a routing list. If a duplicate value is specified, only the last one listed is used.

**Note:** The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

**OMITCP**
OMITCP = NO

The node specified on the NEXTCP statement will not be omitted from the subnet routing list.

This is the default.

OMITCP = YES

The node specified on the NEXTCP statement will be omitted from the subnet routing list for the network ID (NETID) specified on the NETWORK statement. Omitting the node prevents the node from being added as a defined or dynamic entry to the subnet routing list.

SNVC

Defines a maximum number of subnetworks this border node will search when looking for a resource. If no value is coded for SNVC, the value specified on the SNVC start option is used.

Note:
1. If SNVC=1 is coded, the search scope is restricted to the originating subnetwork. A value of SNVC=2 is required to allow a request to traverse one subnetwork boundary.
2. The maximum subnetwork count can be modified downwards if a node along the path has a lower SNVC count. However, the received SNVC is never modified upwards.
Indicates that this VBUILD definition statement defines a set of adjacent cluster definitions to VTAM.

### APPN transmission group profile

Define a transmission group (TG) profile to specify a set of characteristics for a transmission group. The TG profile is not a resource, but describes the TG characteristics of a link.

An adjacent link station (PU) calls a TG profile. When an adjacent link station is activated, VTAM attempts to locate the TG profile. If the TG profile cannot be found, default values are used for the TG characteristics.

A newer TG profile with the same name as a previously defined TG profile updates the older TG profile. The update does not affect existing connections, but subsequent calls to the updated TG profile receive the new information.

### TGP definition statement

Code one TGP definition statement and optional operands for each TG profile.

### IBM-supplied transmission group profiles

The IBM-supplied transmission group profiles are shipped with the name IBMTGPS in SYS1.ASAMLIB. Copy IBMTGPS into SYS1.VTAMLST at VTAM installation.

**Note:** The IBMTGPS member is automatically activated.

See "APPN transmission group (TG) profile definitions (IBMTGPS)" on page 1356 for samples of the profiles, and see z/OS Communications Server: SNA Network Implementation Guide for more details.

#### Table 44. Definition statement for transmission group profile (TGP)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>TGP</td>
<td>CAPACITY</td>
<td>721</td>
<td></td>
<td>8K</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE</td>
<td>726</td>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTTIME</td>
<td>727</td>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY</td>
<td>728</td>
<td></td>
<td>TERRESTR</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SECURITY</td>
<td>728</td>
<td></td>
<td>UNSECURE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>UPARM1</td>
<td>729</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>UPARM2</td>
<td>730</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>UPARM3</td>
<td>730</td>
<td></td>
<td>128</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

- R Required operand.
APPN transmission group profile full syntax

The full syntax for the transmission group profile follows. The syntax for each operand is repeated in "APPN transmission group profile operand descriptions."

TGP

```
TGP
  name
  CAPACITY = 8K
  COSTBYTE = 0
  COSTTIME = 0
  PDELAY = TERRESTR
  SECURITY = UNSECURE
  UPARM1 = 128
  UPARM2 = 128
  UPARM3 = 128
```

Notes:
1. The value of COSTBYTE for intercluster links (ICL) defaults to 1.
2. NEGLIGIB is the default if the TRLE operand is coded.

APPN transmission group profile operand descriptions

The following information describes the operands you can code for the APPN transmission group profile.

CAPACITY
CAPACITY = number_of_bits

range: 1 - 1000 Kb
1 - 1000 Mb
1 - 100 Gb

Specifies the effective capacity of the link that comprises the TG. Specify the value in Kb per second (for example, 100K), Mb per second (for example, 100M), or Gb per second (for example, 10G). This number approximates the bits per second that the link can transmit (the transmission rate of the link, times the maximum load factor expressed as a percentage).

Because the value for CAPACITY is represented as a single byte, the precision of the specified number of bits might be lost. For example, numbers that are close (such as 100K and 101K) can be interpreted by VTAM, and displayed as the same value.

See Table 45 for a list of CAPACITY values you can specify and their corresponding values when displayed. The values appearing in the VALUE CODED column are the values you can specify for CAPACITY. The values in the D NET,TGP column are the hexadecimal values that VTAM displays for the specified value when you issue D NET,TGP. The values in the D NET,TOPO column are the values that VTAM displays for the specified value when you issue D NET,TOPO. So, if you specify 80K for CAPACITY: D NET,TGP displays X‘48’, and D NET,TOPO displays 77K.

Table 45. CAPACITY values when displayed

<table>
<thead>
<tr>
<th>Value coded</th>
<th>D NET, TGP</th>
<th>D NET, TOPO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1K</td>
<td>14</td>
<td>1K</td>
</tr>
<tr>
<td>2K</td>
<td>1C</td>
<td>2K</td>
</tr>
<tr>
<td>3K</td>
<td>22</td>
<td>3K</td>
</tr>
<tr>
<td>4K</td>
<td>25</td>
<td>4K</td>
</tr>
<tr>
<td>5K</td>
<td>28</td>
<td>5K</td>
</tr>
<tr>
<td>6K</td>
<td>2A</td>
<td>6K</td>
</tr>
<tr>
<td>7K</td>
<td>2B</td>
<td>7K</td>
</tr>
<tr>
<td>8K</td>
<td>2D</td>
<td>8K</td>
</tr>
<tr>
<td>9K</td>
<td>2F</td>
<td>9K</td>
</tr>
<tr>
<td>10K</td>
<td>30</td>
<td>10K</td>
</tr>
<tr>
<td>11K</td>
<td>31</td>
<td>11K</td>
</tr>
<tr>
<td>12–13K</td>
<td>32</td>
<td>12K</td>
</tr>
<tr>
<td>14K</td>
<td>33</td>
<td>14K</td>
</tr>
<tr>
<td>15K</td>
<td>34</td>
<td>15K</td>
</tr>
<tr>
<td>16K</td>
<td>35</td>
<td>16K</td>
</tr>
<tr>
<td>17K</td>
<td>36</td>
<td>17K</td>
</tr>
<tr>
<td>18–19K</td>
<td>37</td>
<td>18K</td>
</tr>
</tbody>
</table>
Table 45. CAPACITY values when displayed  (continued)

<table>
<thead>
<tr>
<th>Value coded</th>
<th>D NET, TGP</th>
<th>D NET, TOPO</th>
</tr>
</thead>
<tbody>
<tr>
<td>20–21K</td>
<td>38</td>
<td>20K</td>
</tr>
<tr>
<td>22–23K</td>
<td>39</td>
<td>22K</td>
</tr>
<tr>
<td>24–26K</td>
<td>3A</td>
<td>24K</td>
</tr>
<tr>
<td>27–28K</td>
<td>3B</td>
<td>27K</td>
</tr>
<tr>
<td>29–31K</td>
<td>3C</td>
<td>29K</td>
</tr>
<tr>
<td>32–33K</td>
<td>3D</td>
<td>32K</td>
</tr>
<tr>
<td>34–35K</td>
<td>3E</td>
<td>34K</td>
</tr>
<tr>
<td>36–38K</td>
<td>3F</td>
<td>36K</td>
</tr>
<tr>
<td>39–43K</td>
<td>40</td>
<td>39K</td>
</tr>
<tr>
<td>44–47K</td>
<td>41</td>
<td>44K</td>
</tr>
<tr>
<td>48–52K</td>
<td>42</td>
<td>48K</td>
</tr>
<tr>
<td>53–57K</td>
<td>43</td>
<td>53K</td>
</tr>
<tr>
<td>58–62K</td>
<td>44</td>
<td>58K</td>
</tr>
<tr>
<td>63–67K</td>
<td>45</td>
<td>63K</td>
</tr>
<tr>
<td>68–71K</td>
<td>46</td>
<td>68K</td>
</tr>
<tr>
<td>72–76K</td>
<td>47</td>
<td>72K</td>
</tr>
<tr>
<td>77–86K</td>
<td>48</td>
<td>77K</td>
</tr>
<tr>
<td>87–95K</td>
<td>49</td>
<td>87K</td>
</tr>
<tr>
<td>96–105K</td>
<td>4A</td>
<td>96K</td>
</tr>
<tr>
<td>106–115K</td>
<td>4B</td>
<td>106K</td>
</tr>
<tr>
<td>116–124K</td>
<td>4C</td>
<td>116K</td>
</tr>
<tr>
<td>125–134K</td>
<td>4D</td>
<td>125K</td>
</tr>
<tr>
<td>135–143K</td>
<td>4E</td>
<td>135K</td>
</tr>
<tr>
<td>144–153K</td>
<td>4F</td>
<td>144K</td>
</tr>
<tr>
<td>154–172K</td>
<td>50</td>
<td>154K</td>
</tr>
<tr>
<td>173–191K</td>
<td>51</td>
<td>173K</td>
</tr>
<tr>
<td>192–211K</td>
<td>52</td>
<td>192K</td>
</tr>
<tr>
<td>212–230K</td>
<td>53</td>
<td>212K</td>
</tr>
<tr>
<td>231–249K</td>
<td>54</td>
<td>231K</td>
</tr>
<tr>
<td>250–268K</td>
<td>55</td>
<td>250K</td>
</tr>
<tr>
<td>269–287K</td>
<td>56</td>
<td>269K</td>
</tr>
<tr>
<td>288–307K</td>
<td>57</td>
<td>288K</td>
</tr>
<tr>
<td>308–345K</td>
<td>58</td>
<td>308K</td>
</tr>
<tr>
<td>346–383K</td>
<td>59</td>
<td>346K</td>
</tr>
<tr>
<td>384–422K</td>
<td>5A</td>
<td>384K</td>
</tr>
<tr>
<td>423–460K</td>
<td>5B</td>
<td>423K</td>
</tr>
<tr>
<td>461–499K</td>
<td>5C</td>
<td>461K</td>
</tr>
<tr>
<td>500–537K</td>
<td>5D</td>
<td>500K</td>
</tr>
<tr>
<td>538–575K</td>
<td>5E</td>
<td>538K</td>
</tr>
<tr>
<td>576–614K</td>
<td>5F</td>
<td>576K</td>
</tr>
</tbody>
</table>
Table 45. CAPACITY values when displayed (continued)

<table>
<thead>
<tr>
<th>Value coded</th>
<th>D NET, TGP</th>
<th>D NET, TOPO</th>
</tr>
</thead>
<tbody>
<tr>
<td>615–691K</td>
<td>60</td>
<td>615K</td>
</tr>
<tr>
<td>692–767K</td>
<td>61</td>
<td>692K</td>
</tr>
<tr>
<td>768–844K</td>
<td>62</td>
<td>768K</td>
</tr>
<tr>
<td>845–921K</td>
<td>63</td>
<td>845K</td>
</tr>
<tr>
<td>922–998K</td>
<td>64</td>
<td>922K</td>
</tr>
<tr>
<td>999–1M</td>
<td>65</td>
<td>999K</td>
</tr>
<tr>
<td>2M</td>
<td>6D</td>
<td>2M</td>
</tr>
<tr>
<td>3M</td>
<td>71</td>
<td>3M</td>
</tr>
<tr>
<td>4M</td>
<td>75</td>
<td>4M</td>
</tr>
<tr>
<td>5M</td>
<td>78</td>
<td>5M</td>
</tr>
<tr>
<td>6M</td>
<td>79</td>
<td>6M</td>
</tr>
<tr>
<td>7M</td>
<td>7B</td>
<td>7M</td>
</tr>
<tr>
<td>8M</td>
<td>7D</td>
<td>8M</td>
</tr>
<tr>
<td>9M</td>
<td>7E</td>
<td>9M</td>
</tr>
<tr>
<td>10–11M</td>
<td>80</td>
<td>10M</td>
</tr>
<tr>
<td>12M</td>
<td>81</td>
<td>12M</td>
</tr>
<tr>
<td>13M</td>
<td>82</td>
<td>13M</td>
</tr>
<tr>
<td>14M</td>
<td>83</td>
<td>14M</td>
</tr>
<tr>
<td>15M</td>
<td>84</td>
<td>15M</td>
</tr>
<tr>
<td>16–17M</td>
<td>85</td>
<td>16M</td>
</tr>
<tr>
<td>18M</td>
<td>86</td>
<td>18M</td>
</tr>
<tr>
<td>19M</td>
<td>87</td>
<td>19M</td>
</tr>
<tr>
<td>20–22M</td>
<td>88</td>
<td>20M</td>
</tr>
<tr>
<td>23–24M</td>
<td>89</td>
<td>23M</td>
</tr>
<tr>
<td>25–27M</td>
<td>8A</td>
<td>25M</td>
</tr>
<tr>
<td>28–29M</td>
<td>8B</td>
<td>28M</td>
</tr>
<tr>
<td>30–31M</td>
<td>8C</td>
<td>30M</td>
</tr>
<tr>
<td>32–34M</td>
<td>8D</td>
<td>32M</td>
</tr>
<tr>
<td>35–36M</td>
<td>8E</td>
<td>35M</td>
</tr>
<tr>
<td>37–39M</td>
<td>8F</td>
<td>37M</td>
</tr>
<tr>
<td>40–44M</td>
<td>90</td>
<td>40M</td>
</tr>
<tr>
<td>45–49M</td>
<td>91</td>
<td>45M</td>
</tr>
<tr>
<td>50–54M</td>
<td>92</td>
<td>50M</td>
</tr>
<tr>
<td>55–58M</td>
<td>93</td>
<td>55M</td>
</tr>
<tr>
<td>59–63M</td>
<td>94</td>
<td>59M</td>
</tr>
<tr>
<td>64–68M</td>
<td>95</td>
<td>64M</td>
</tr>
<tr>
<td>69–73M</td>
<td>96</td>
<td>69M</td>
</tr>
<tr>
<td>74–78M</td>
<td>97</td>
<td>74M</td>
</tr>
<tr>
<td>79–88M</td>
<td>98</td>
<td>79M</td>
</tr>
<tr>
<td>89–98M</td>
<td>99</td>
<td>89M</td>
</tr>
<tr>
<td>Value coded</td>
<td>D NET, TGP</td>
<td>D NET, TOPO</td>
</tr>
<tr>
<td>-------------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>99–108M</td>
<td>9A</td>
<td>99M</td>
</tr>
<tr>
<td>109–117M</td>
<td>9B</td>
<td>109M</td>
</tr>
<tr>
<td>118–127M</td>
<td>9C</td>
<td>118M</td>
</tr>
<tr>
<td>128–137M</td>
<td>9D</td>
<td>128M</td>
</tr>
<tr>
<td>138–147M</td>
<td>9E</td>
<td>138M</td>
</tr>
<tr>
<td>148–157M</td>
<td>9F</td>
<td>148M</td>
</tr>
<tr>
<td>158–176M</td>
<td>A0</td>
<td>158M</td>
</tr>
<tr>
<td>177–196M</td>
<td>A1</td>
<td>177M</td>
</tr>
<tr>
<td>197–216M</td>
<td>A2</td>
<td>197M</td>
</tr>
<tr>
<td>217–235M</td>
<td>A3</td>
<td>217M</td>
</tr>
<tr>
<td>236–255M</td>
<td>A4</td>
<td>236M</td>
</tr>
<tr>
<td>256–275M</td>
<td>A5</td>
<td>256M</td>
</tr>
<tr>
<td>276–294M</td>
<td>A6</td>
<td>276M</td>
</tr>
<tr>
<td>295–314M</td>
<td>A7</td>
<td>295M</td>
</tr>
<tr>
<td>315–353M</td>
<td>A8</td>
<td>315M</td>
</tr>
<tr>
<td>354–393M</td>
<td>A9</td>
<td>354M</td>
</tr>
<tr>
<td>394–432M</td>
<td>AA</td>
<td>393M</td>
</tr>
<tr>
<td>433–471M</td>
<td>AB</td>
<td>433M</td>
</tr>
<tr>
<td>472–511M</td>
<td>AC</td>
<td>472M</td>
</tr>
<tr>
<td>512–550M</td>
<td>AD</td>
<td>511M</td>
</tr>
<tr>
<td>551–589M</td>
<td>AE</td>
<td>551M</td>
</tr>
<tr>
<td>590–629M</td>
<td>AF</td>
<td>590M</td>
</tr>
<tr>
<td>630–707M</td>
<td>B0</td>
<td>630M</td>
</tr>
<tr>
<td>708–786M</td>
<td>B1</td>
<td>708M</td>
</tr>
<tr>
<td>787–865M</td>
<td>B2</td>
<td>787M</td>
</tr>
<tr>
<td>866–943M</td>
<td>B3</td>
<td>865M</td>
</tr>
<tr>
<td>944M–1G</td>
<td>B4</td>
<td>944M</td>
</tr>
<tr>
<td>2G</td>
<td>BC</td>
<td>2G</td>
</tr>
<tr>
<td>3G</td>
<td>C1</td>
<td>3G</td>
</tr>
<tr>
<td>4G</td>
<td>C4</td>
<td>4G</td>
</tr>
<tr>
<td>5G</td>
<td>C7</td>
<td>5G</td>
</tr>
<tr>
<td>6G</td>
<td>C9</td>
<td>6G</td>
</tr>
<tr>
<td>7G</td>
<td>CB</td>
<td>7G</td>
</tr>
<tr>
<td>8G</td>
<td>CC</td>
<td>8G</td>
</tr>
<tr>
<td>9G</td>
<td>CE</td>
<td>9G</td>
</tr>
<tr>
<td>10G</td>
<td>CF</td>
<td>10G</td>
</tr>
<tr>
<td>11G</td>
<td>D0</td>
<td>11G</td>
</tr>
<tr>
<td>12G</td>
<td>D1</td>
<td>12G</td>
</tr>
<tr>
<td>13G</td>
<td>D2</td>
<td>13G</td>
</tr>
<tr>
<td>14–15G</td>
<td>D3</td>
<td>14G</td>
</tr>
</tbody>
</table>
Table 45. CAPACITY values when displayed (continued)

<table>
<thead>
<tr>
<th>Value coded</th>
<th>D NET, TGP</th>
<th>D NET, TOPO</th>
</tr>
</thead>
<tbody>
<tr>
<td>16G</td>
<td>D4</td>
<td>16G</td>
</tr>
<tr>
<td>17G</td>
<td>D5</td>
<td>17G</td>
</tr>
<tr>
<td>18G</td>
<td>D6</td>
<td>18G</td>
</tr>
<tr>
<td>19-20G</td>
<td>D7</td>
<td>19G</td>
</tr>
<tr>
<td>21-22G</td>
<td>D8</td>
<td>21G</td>
</tr>
<tr>
<td>23-25G</td>
<td>D9</td>
<td>23G</td>
</tr>
<tr>
<td>26-27G</td>
<td>DA</td>
<td>26G</td>
</tr>
<tr>
<td>28-30G</td>
<td>DB</td>
<td>28G</td>
</tr>
<tr>
<td>31-32G</td>
<td>DC</td>
<td>31G</td>
</tr>
<tr>
<td>33-35G</td>
<td>DD</td>
<td>33G</td>
</tr>
<tr>
<td>36-37G</td>
<td>DE</td>
<td>36G</td>
</tr>
<tr>
<td>38-40G</td>
<td>DF</td>
<td>38G</td>
</tr>
<tr>
<td>41-45G</td>
<td>E0</td>
<td>41G</td>
</tr>
<tr>
<td>46-50G</td>
<td>E1</td>
<td>46G</td>
</tr>
<tr>
<td>51-55G</td>
<td>E2</td>
<td>51G</td>
</tr>
<tr>
<td>56-60G</td>
<td>E3</td>
<td>56G</td>
</tr>
<tr>
<td>61-65G</td>
<td>E4</td>
<td>61G</td>
</tr>
<tr>
<td>66-70G</td>
<td>E5</td>
<td>66G</td>
</tr>
<tr>
<td>71-75G</td>
<td>E6</td>
<td>71G</td>
</tr>
<tr>
<td>76-80G</td>
<td>E7</td>
<td>76G</td>
</tr>
<tr>
<td>81-90G</td>
<td>E8</td>
<td>81G</td>
</tr>
<tr>
<td>91-100G</td>
<td>E9</td>
<td>91G</td>
</tr>
</tbody>
</table>

COSTBYTE
COSTBYTE = 0
\[ \text{cost per byte} \]

Notes:
1. TG characteristics are not valid with DYNTYPE=VN.

\textit{range:} \hspace{1em} 0–255

Specifies a cost-per-byte-transmitted to be associated with the TG, where 0 is the least expensive cost per byte and 255 is the most expensive.

COSTTIME

\[ \text{cost per unit time} \]

range: \hspace{1em} 0–255

Specifies a cost-per-unit-of-time to be associated with the TG, where 0 is the least expensive cost per unit of time and 255 is the most expensive.
Specifies the required name of this TG profile.

**PDELAY**

<table>
<thead>
<tr>
<th>PDELAY</th>
<th>TERRESTR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NEGLIGIB</td>
</tr>
</tbody>
</table>

Notes:

1. NEGLIGIB is the default if the TRLE operand is coded.

Specifies the maximum propagation delay of the link for the transmission group. Propagation delay represents the time needed for a signal to travel from one end of the link to the other.

**PDELAY=LONG**

Indicates a satellite delay (greater than 245.76 milliseconds).

**PDELAY=NEGLIGIB**

Indicates a local area network delay (less than 0.48 milliseconds).

**PDELAY=PACKET**

Indicates a packet-switched network delay (between 49.152 and 245.76 milliseconds).

**PDELAY=TERRESTR**

Indicates telephone network delay (between 0.48 and 49.152 milliseconds).

**SECURITY**
Specifies the security level of the transmission group. The following options range from the least secure level to the most secure level.

**SECURITY=UNSECURE**
- Specifies no security level

**SECURITY=PUBLIC**
- Specifies a public switched network

**SECURITY=UNDERGRO**
- Specifies an underground cable, not guarded

**SECURITY=SECURE**
- Specifies a secure conduit, not guarded

**SECURITY=GUARDED**
- Specifies a guarded conduit, physical only

**SECURITY=ENCRYPT**
- Specifies link encryption

**SECURITY=SHIELDED**
- Specifies a guarded conduit, physical and radiation shielded

**UPARM1**
APPN-to-subarea Class of Service mapping table

The APPN-to-subarea (APPNTOSA) Class of Service (CoS) mapping table enables you to define how APPN CoS names are mapped to subarea CoS names for a VR-based TG portion of an HPR route. Subarea CoS names are needed for a VR-based TG because VR-based TGs use subarea virtual routes. The subarea CoS name used to find the appropriate VRs to set up an HPR route is determined as follows:

1. If an APPNTOSA CoS mapping table is active and a specific subarea CoS is found for a given APPN CoS, that subarea CoS is used. Because a blank LOGMODE name is valid in subarea, use blanks after COS= to map to it. A blank logmode name is NOT valid in APPN. (If the blank LOGMODE name is the default, code as "COS=,DEFAULT=YES").
2. If an APPNTOSA CoS mapping table is active, a specific subarea CoS is not found for a given APPN CoS, and a default subarea CoS is coded, the default subarea CoS is used.
3. If an APPNTOSA CoS mapping table is active, a specific subarea CoS is not found for a given APPN CoS, and a default subarea CoS is not coded, the ISTVTVCOS Class of Service is used if coded.
4. If no APPNTOSA CoS mapping table is active, the ISTVT COS Class of Service is used if coded.
5. If none of the above conditions are met, route setup fails.

To modify an active APPNTOSA CoS mapping table:
1. Edit the APPNTOSA VTAMLST member, making the modifications that you want.
2. Reactivate that APPNTOSA VTAMLST member.

To delete an active APPNTOSA CoS mapping table:
1. Create a version of the APPNTOSA table that has only one entry in it. This
   entry must be a dummy entry that does not specify DEFAULT=YES.
2. Activate this table. Because the new table has no usable (or default) entries in
   it, the CoS mapping function will fail and VTAM will revert back to the
   original LOGMODE table method.

**Figure 2** illustrates the definition of an APPNTOSA CoS mapping table.

```plaintext
TABLE1 VBUILD TYPE=APPNTOSA
CONNECT MAPSTO COS=COSAPPL1,DEFAULT=YES
BATCH MAPSTO COS=COSAPPL2
BATCHSC MAPSTO COS=COSAPPL3
INTER MAPSTO COS=COSAPPL4
SNASVCMG MAPSTO COS=COSAPPL5
INTERSC MAPSTO COS=ISTCOSDF
```

**Figure 2. APPN-to-subarea CoS mapping table example**

**Figure 3 on page 732** illustrates disabling the APPNTOSA COS mapping function.
Table 46. Definition statements for APPNTOSA table

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=APPNTOSA</td>
<td>734</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>appn_cos</td>
<td>MAPSTO</td>
<td>COS</td>
<td>733</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R Required operand.

APPN-to-subarea Class of Service mapping table coding notes

This information contains coding notes for an APPN-to-subarea Class of Service mapping table.

**VBUILD definition statement**

Code a VBUILD definition statement as the first statement in an APPN-to-subarea CoS mapping table.

**MAPSTO definition statement**

Code a MAPSTO definition statement to define an APPN-to-subarea CoS mapping.

APPN-to-subarea Class of Service mapping table full syntax

The full syntax for the APPN-to-subarea CoS mapping table follows. The syntax for each operand is repeated in "APPN-to-subarea Class of Service mapping table operand descriptions" on page 733.

VBUILD
APPN-to-subarea Class of Service mapping table operand descriptions

The following information describes the operands you can code for the APPN-to-subarea CoS mapping table.

**appn_cos**

Statements: MAPSTO

Dependencies: none

Specifies a required 1-8 character name identifying an APPN CoS name.

**COS**
Specifies a 0–8 character name identifying a subarea CoS name. (If the subarea CoS name is blanks, just code COS=).

**DEFAULT=YES**

Specifies that this entry is the default subarea CoS for this table.

**name**

Assigns a name to the APPN-to-subarea CoS mapping table.

**TYPE**
Means that this VBUILD definition statement defines an APPN-to-subarea CoS mapping table.

**Subarea-to-APPN Class of Service mapping table**

The subarea-to-APPN (SATOAPPN) Class of Service (CoS) mapping table enables you to define how subarea CoS names are mapped to APPN CoS names when a session goes from the subarea part of the network to the APPN part of the network.

VTAM picks an APPN CoS as follows:
1. APPNCOS is first picked up from the requested LOGMODE entry, if it exists. A blank Subarea LOGMODE name is valid.
2. The new table, if present, is searched for an APPN mapping of the subarea CoS.
3. If a subarea CoS to APPN CoS mapping is in the table, and it is different from the APPN CoS found in the logon mode table entry, or a DEFAULT entry is coded in the table, the SATOAPPN CoS table entry is used for this session.
4. If no mapping exists, the APPNCOS from the logon mode table is used, or the APPNCOS default is used if no APPNCOS was in the LOGMODE entry of the logon mode table.

To modify an active SATOAPPN CoS mapping table:
1. Edit the SATOAPPN VTAMLST member, making the modifications that you want.
2. Reactivate that SATOAPPN VTAMLST member.

To delete an active SATOAPPN CoS mapping table:
1. Create a version of the SATOAPPN table that has only one entry in it. This entry must be a dummy entry that does not specify DEFAULT=YES.
2. Activate this table. Because the new table has no usable (or default) entries in it, the CoS mapping function will fail and VTAM will revert back to the original LOGMODE table method.

Figure 4 on page 736 illustrates definition of a SATOAPPN CoS mapping table.
Table 47. Definition statements for SATOAPPN table

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=SATOAPPN</td>
<td>738</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sa_cos</td>
<td>MAPSTO</td>
<td>COS</td>
<td>737</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DEFAULT</td>
<td></td>
<td>738</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R Required operand.

Subarea-to-APPN Class of Service mapping table coding notes

This information contains coding notes for a subarea-to-APPN Class of Service mapping table.

VBUILD definition statement

Code a VBUILD definition statement as the first statement in a subarea-to-APPN CoS mapping table.

MAPSTO definition statement

Code a MAPSTO definition statement to define a subarea-to-APPN CoS mapping.

Subarea-to-APPN Class of Service mapping table full syntax

The full syntax for the subarea-to-APPN CoS mapping table follows. The syntax for each operand is repeated in “Subarea-to-APPN Class of Service mapping table operand descriptions” on page 737.

VBUILD

TABLE2 VBUILD TYPE=SATOAPPN
COSAPPL1 MAPSTO COS=#CONNECT,DEFAULT=YES
COSAPPL2 MAPSTO COS=#BATCH
COSAPPL3 MAPSTO COS=#BATCHSC
MAPSTO COS=#INTER
COSAPPL5 MAPSTO COS=SNASVCMG
COSAPPL6 MAPSTO COS=#INTERSC

Figure 4. Subarea-to-APPN CoS mapping table example
Subarea-to-APPN Class of Service mapping table operand descriptions

The following information describes the operands you can code for the subarea-to-APPN CoS mapping table.

**sa_cos**

Specifies a required 0–8 character name identifying a subarea CoS name. (A blank subarea CoS name is valid).
COS—appn_cos

Statements: MAPSTO
Dependencies: none

Specifies a 1–8 character name identifying an APPN CoS name.

DEFAULT=YES

Statements: MAPSTO
Dependencies: none

Specifies that this entry is the default APPN CoS for this table.

name

Statements: VBUILD
Dependencies: none

Assigns a name to the subarea-to-APPN CoS mapping table.

TYPE
Means that this VBUILD definition statement defines an subarea-to-APPN CoS mapping table.

**Border node CoS mapping definitions**

The border node Class of Service (CoS) mapping definitions (BNCOSMAP) enable you to define how the CoS name from an adjacent APPN network (that is, a nonnative CoS name) should be mapped to the local network CoS name (native CoS name). The border node CoS mapping definitions enables each subnetwork to maintain its own CoS names.

You can code the CoS mapping definitions to map a nonnative CoS name to a local CoS name for a specific NETID. You can also define code definitions to be used if you have not defined an explicit table for the NETID.

If a table is coded, the nonnative CoS name is located in the table and mapped to the corresponding native CoS name. If an entry for the nonnative CoS name is not found, the CoS name in the adjacent subnetwork is used provided that the received CoS is defined for this node by the APPN Class of Service definitions. If the received CoS name is not defined, then the CoS name coded on the APPNCOS start option will be used.

[Figure 5 on page 740](#) defines an explicit CoS mapping table for NETWORKA, NETWORKB, and NETWORKC. The same mapping table is used for both NETWORKB and NETWORKC. The example also defines a default CoS mapping table, DEFAULT, to be used if an explicit table cannot be found.
TABLE1  VBUILD  TYPE=BNCOSMAP
************************************************************************************
* Adjacent network ID
* ***************************************************************************
NETWORKA NETWORK  NETID=NETWORKA  adjacent network ID
COS1  MAPSTO  COS=COSA  map COS
COS2  MAPSTO  COS=COSB  map COS
COS3  MAPSTO  COS=COSC  map COS
************************************************************************************
* Adjacent network ID
* ***************************************************************************
NETWORKB  NETWORK  NETID=(NETWORKB,NETWORKC)  adjacent network ID
COS1  MAPSTO  COS=COSA  map COS
COS3  MAPSTO  COS=COSF  map COS
COS5  MAPSTO  COS=COSE  map COS
************************************************************************************
* Default network ID
* ***************************************************************************
DEFAULT  NETWORK  default network ID
COS1  MAPSTO  COS=COSA  map COS

Figure 5. CoS mapping table example

Note: If a request is to be sent to a nonnative network node and the nonnative node does not have the capability to perform mapping, VTAM will map the native CoS to a nonnative CoS. VTAM will search the native CoS names (specified by the COS operand) under the NETWORK definitions for the destination NETID or the default definitions (if the NETID is not defined on a NETWORK statement). Using the first definition of the native CoS name, the cos_name will be used on the outbound request.

Table 48. Definition statements for COSMAP table

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>VBUILD</td>
<td>TYPE=BNCOSMAP</td>
<td></td>
<td>743</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NETWORK</td>
<td>NETID</td>
<td></td>
<td>742</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cos_name</td>
<td>MAPSTO</td>
<td>COS</td>
<td>741</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:
R  Required operand.

Border node CoS mapping definitions coding notes

This information contains coding notes for border node Class of Service mapping definitions.

**VBUILD definition statement**

Code a VBUILD definition statement as the first statement in a border node CoS mapping table.

**NETWORK definition statement**

Code a NETWORK definition statement for each network for which you are defining a border node CoS mapping table.
MAPSTO definition statement

Code a MAPSTO definition statement to define the native and nonnative APPN CoS mappings.

Border node CoS mapping definitions full syntax

The full syntax for the border node CoS mapping definition follows. The syntax for each operand is repeated in “Border node CoS mapping definitions operand descriptions.”

VBUILD

```
  [name] VBUILD TYPE = BNCOSMAP
```

NETWORK

```
  [name] NETWORK NETID = [network_id]
```

MAPSTO

```
  [cos_name] MAPSTO COS = [native_cosname]
```

Border node CoS mapping definitions operand descriptions

The following information describes the operands you can code for CoS mapping tables.

COS
Specifies the 1–8 character name identifying a native CoS name.

**cos_name**

Specifies the 1–8 character name identifying the nonnative CoS name.

**name**

On **VBUILD**:
Assigns a name to border node CoS mapping definitions.

On **NETWORK**:
Assigns a name to the NETWORK definition statement. To avoid confusion, use the same value for **name** and the NETID operand.

**NETID**
Specifies the optional 1–8 character network identifier of the adjacent network. If you do not code NETID at all, this NETWORK definition statement represents a default border node CoS mapping to be used for unknown or unlisted NETIDs (those not explicitly coded on other NETWORK statements).

Note: The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in other countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network identifiers. Although VTAM allows you to use #, other products might enforce this restriction.

**TYPE**

Means that this VBUILD definition statement defines a set of border node CoS mapping definitions.

## Network node server list

You define a network node server list for an end node (EN) with a VBUILD definition statement followed by one or more NETSRVR definition statements.

If you choose not to define a network node server list, a default network node server list consisting of a single nameless NETSRVR definition statement is automatically activated when the end node is initialized.

If the NNSPREF start option is defined, the network node specified will have a higher priority than any other network node specified in the active NETSRVR list.

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=NETSRVR ORDER</td>
<td>748</td>
<td>R</td>
<td>FIRST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>NETSRVR</td>
<td>NETID</td>
<td>747</td>
<td></td>
<td>REQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SLUINIT</td>
<td>747</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ENBCAST</td>
<td>745</td>
<td></td>
<td>NO</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 49. Definition statement for network node server list (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
</table>

Legend:

R Required operand.

Network node server list coding notes

This information contains coding notes for a network node server list.

**VBUILD definition statement**

Code a VBUILD definition statement to begin the network node server list.

**NETSRVR definition statement**

Code a NETSRVR definition statement to define a network node (NN) as a network node server to the EN. There is an unlimited number of NETSRVR definition statements in any list (until VTAM runs out of storage). Duplicate entries in the list are ignored.

If you do not code a NETSRVR definition statement, the end node selects a network node in the same way as when no network node server list is specified.

Network node server list full syntax

The full syntax for the network node server list follows. The syntax for each operand is repeated in "Network node server list operand descriptions" on page 745.
**Network node server list operand descriptions**

The following information describes the operands you can code for the network node server list.

**name**

---

On **VBUILD**:  
Assigns an optional name to a network node server list.

On **NETSRVR**:  
Identifies the CP name of a NN that can be used as a network node server for this EN.

If you do not specify a name on a **NETSRVR** definition statement, any other known adjacent NN, not included in this network node server list, will be considered as a potential network node server for this EN. Only one nameless **NETSRVR** definition statement entry may appear in a network node server list, and it must be the last entry in the list. Any subsequent entries in the list are ignored. Note that any named **NETSRVR** entry previously considered and rejected as a network node server is not tried again during nameless entry processing.

**ENBCAST**
Specifies whether the network node server is allowed to search the end node for unknown resources.

**ENBCAST=NO**

Specifies that the network node server is not allowed to search the end node for unknown resources.

**ENBCAST=YES**

Specifies that the network node server is allowed to search the end node for unknown resources. This will result in the end node being searched on all broadcast searches, which could have an impact on performance.

This operand can be used to enable the generic resource function to operate with a network node server that is not connected to the same generic resource structure as the end node. Because of the possible performance impact, this is intended as a backup mode of operation only; therefore, **ENBCAST=YES** should be coded only on the NETSRVR definition statement of a backup network node server.

In addition, this backup mode of operation works only when all end nodes connected to the same generic resource structure are using the same backup node as their network node server. For this reason, it is necessary to configure all of these end nodes such that they will switch to the backup-mode network node server only when the primary network node server fails (rather than, for example, when a link failure occurs).

**Recommendations:**

1. Configure all end nodes attached to the same generic resource structure with the same network node server (NETSRVR) list. The first entry in the network node server list should define the primary network node server (that is, the network node that is attached to the same generic resource structure as the end nodes) and should specify (or default to) **ENBCAST=NO**. The second (and last) entry in the list should define the backup-mode network node server and should specify **ENBCAST=YES**. (In order to guarantee that all end nodes switch to the backup-mode network node server at the same time, all end nodes must define with the same primary and backup-mode network node server names.)

2. Code **ORDER-FIRST** in the network node server (NETSRVR) list, so that the end nodes will always attempt to use the primary network node server (the one attached to the same generic resource structure) first.

3. Code the **NNSPREF** start option on all end nodes with the name of the primary network node server (the one attached to the same generic resource structure). This will cause all end nodes to automatically terminate the backup-mode of operation by switching back to the primary network node server as soon as it becomes available again.

4. To reduce the possibility of a link failure causing only some end nodes to switch to the backup-mode network node server, it is recommended that redundant links (which support CP-CP sessions) be defined between each end node and its primary network node server. XCF links are a very good choice for this, because they provide the necessary
connectivity as long as there is connectivity to the coupling facility. (In fact, because connectivity to the coupling facility is required to support generic resources, defining XCF links may eliminate the need to define redundant links specifically for this purpose.)

**NETID**

```
/NETID=network_id/
```

Specifies a network ID for the named NN. If the NETID operand is not specified, the network ID of this EN is assumed.

**Note:** NETID is not valid when no value is specified for `name`.

**ORDER**

```
/ORDER=FIRST/
```

Determines how the network node server list is processed.

**ORDER=FIRST**

Specifies that the EN always attempts to find a network node server from the network node server list starting with the first entry. If the first NN in the list is unavailable, the EN attempts to get the second NN in the list to act as its network node server. If the second NN is also unavailable, the third NN in the list is checked, and so on until an available NN is found. Thus, a prioritized list is created where the most preferred NN server is the first entry in the list, the second most preferred is the second entry in the list, and so on. FIRST is the default order of selection.

**ORDER=NEXT**

Specifies that the EN attempts to find a network node server starting with the next entry after the NN selected the last time the list was used. If the next NN in the list is unavailable, the EN attempts to get the following NN in the list to act as its network node server, and so on until an available NN is found. When the bottom of the list is encountered, the first entry in the list is considered to be the next entry. Thus, NNs are selected in a round-robin manner and no preference is given to one node in the list over another node.

**SLUINIT**
Specifies whether the network node server is required to support SLU-initiated sessions as a prerequisite for establishment of CP-CP sessions between the indicated network node and this end node.

**SLUINIT=OPT**

Specifies that CP-CP sessions between this end node and the indicated network node do not depend on the level of network node server support provided by the network node for SLU-initiated sessions.

**SLUINIT=REQ**

Specifies that CP-CP sessions between this end node and the indicated network node are not permitted when, during session setup, the network node indicates that its network node server support does not include SLU-initiated session capability.

**TYPE**

Means that this VBUILD definition statement defines a network node server list.

---

### Generic resources preferences table

Use the generic resources preferences table (GRPREFS) to define how a generic resource resolution is performed. You can define an entry in the table for each generic name; you can also define a default entry. VTAM does not provide a default generic resources preferences table; however, all generic resource preferences have a default value.

Only one GRPREFS table can be active at a time. The most recently activated GRPREFS table entirely replaces any previously active GRPREFS table.

There is no function to delete a GRPREFS table, however to effectively delete any previous GRPREFS table and return to using the VTAM default generic resource preferences you can activate a GRPREFS table containing only one GRPREF definition statement without a name and without any operands.

Figure 6 on page 749 shows the definition of a GRPREFS table.
Table 50. Definition statements for GRPREFS table

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=GRPREFS</td>
<td></td>
<td>R</td>
<td></td>
<td></td>
<td>R</td>
</tr>
<tr>
<td>name</td>
<td>GRPREF</td>
<td>GREXIT=NO</td>
<td>750</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WLM=YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCAPPL=YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCLU=YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PASSOLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WLM=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCAPPL=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PASSOLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WLM=YES</td>
<td>751</td>
<td>NO</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCAPPL=YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCLU=YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PASSOLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WLM=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCAPPL=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PASSOLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WLM=YES</td>
<td>752</td>
<td>NO</td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCAPPL=YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCLU=YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PASSOLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WLM=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCAPPL=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PASSOLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WLM=YES</td>
<td>753</td>
<td>NO</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCAPPL=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PASSOLU=NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R Required operand.

Generic resources preferences table coding notes

This information contains coding notes for a generic resources preferences table.

**VBUILD definition statement**

Code a VBUILD definition statement as the first statement in a generic resource preference table.

**GRPREF definition statement**

Code a GRPREF definition statement to define generic resource preferences for a generic resource name or to define the default generic resource preferences.

**Generic resources preferences table full syntax**

The full syntax for the generic resources preferences table follows. The syntax for each operand is repeated in "Generic resources preferences table operand descriptions" on page 750.

**VBUILD**
Generic resources preferences table operand descriptions

The following information describes the operands you can code for the generic resources preferences table.

**GREXIT**

- **GREXIT=NO**
  - Indicates that the generic resource exit (ISTEXCGR) is not called for generic resource resolution.

- **GREXIT=YES**
  - Indicates that the generic resource exit (ISTEXCGR) is called for generic resource resolution.

**LOCAPPL**

- Indicates whether the generic resource exit (ISTEXCGR) is called for generic resource resolution.

**LOCAPPL=NO**
  - Indicates that the generic resource exit (ISTEXCGR) is not called for generic resource resolution.

**LOCAPPL=YES**
  - Indicates that the generic resource exit (ISTEXCGR) is called for generic resource resolution.
Indicates whether generic resource resolution for a session initiated from an application on this host should prefer generic resource instances on this host.

**LOCAPPL=YES**
Indicates that generic resource resolution for a session initiated from an application on this host prefers generic resource instances on this host. If no generic resource instances are available on this host, then all instances of the generic resource are eligible for resolution.

**LOCAPPL=NO**
Indicates that generic resource resolution for a session initiated from an application on this host does not prefer generic resource instances on this host. All instances of the generic resource are eligible for resolution.

**LOCLU**
Indicates whether generic resource resolution for a session initiated from a local LU that is part of a local SNA or local non-SNA major node on this host should prefer generic resource instances on this host.

**LOCLU=YES**
Indicates that generic resource resolution for a session initiated from a local LU that is part of a local SNA or local non-SNA major node on this host prefers generic resource instances on this host. If no generic resource instances are available on this host, then all instances of the generic resource are eligible for resolution.

**LOCLU=NO**
Indicates that generic resource resolution for a session initiated from a local LU that is part of a local SNA or local non-SNA major node on this host does not prefer generic resource instances on this host. All instances of the generic resource are eligible for resolution.
On the VBUILD definition statement, the name operand assigns a name to the generic resource preference table. On the GRPREF definition statement, the name operand assigns a name to the generic resource preferences. The GRPREF name must match the name of the generic resource; however, a GRPREF definition can be defined before the generic resource becomes active.

If a generic resource table is defined, then the first GRPREF definition that does not have a name is used to define the default generic resource preferences. It will be displayed with a name of "NAMELESS". If a generic resource preferences table is not defined or is defined without a nameless entry, then a default generic resource preference will be automatically activated. It will be displayed with a name of "DEFAULT". The default generic resource preferences are used for generic resource resolution of any generic resource name not found in the generic resource preference table.

**PASSOLU**

For third-party-initiated (CLSDST PASS) sessions, the PASSOLU operand indicates whether generic resource resolution should prefer generic resource instances located on the origin logical unit (OLU) host (the resource that has the session that is being passed).

**PASSOLU=YES**
For third-party-initiated (CLSDST PASS) sessions, indicates that generic resource resolution prefers generic resource instances located on the OLU host (the resource that has the session that is being passed). If no generic resource instances are available on the OLU host, then all instances of the generic resource are eligible for resolution.

**PASSOLU=NO**
For third-party-initiated (CLSDST PASS) sessions, indicates that generic resource resolution does not prefer generic resource instances located on the OLU host (the resource that has the session that is being passed). All instances of the generic resource are eligible for resolution.

**Tip:** Both the LOCAPPL and LOCLU generic resource preferences could affect the PASSOLU preference. If either of the LOCAPPL or PASSOLU preferences are set to YES, then a CLSDST PASS session from a local application to a generic resource will prefer a generic resource on the local host. If either of the
LOCLU or PASSOLU preferences are set to YES, then a CLSDST PASS session from a local LU to a generic resource will prefer a generic resource on the local host.

**TYPE**

```
TYPE = GRPREFS
```

Means that this VBUILD definition statement defines a generic resource preference table.

**WLM**

```
WLM = YES
```

Indicates whether the MVS workload manager is called for generic resource resolution.

**WLM=YES**

Indicates that the MVS workload manager is called for generic resource resolution.

**WLM=NO**

Indicates that the MVS workload manager is not called for generic resource resolution.
Chapter 4. VTAM start options

This topic describes the VTAM start options and how to code them.

See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) for information on the following topics:

- Start option sources
- Creating start option and configuration lists
- Overriding start options
- Entering start options during the start procedure

Coding VTAM start options

Code start option lists as one or more 80-byte records in the following format:

<table>
<thead>
<tr>
<th>position</th>
<th>position</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>72</td>
</tr>
</tbody>
</table>

... Indicates blanks. One or more blanks can precede the first item.

**item** Represents a start option in a start option list. Separate items with commas. Use no intervening blanks.

**c** Reserved for the continuation indicator. The continuation indicator is optional, but is ignored. VTAM ignores positions 72–80.

The following information shows a sample start list, ATCSTRyy. The first line is a scale to show you column positions. Every tenth column position is indicated by a number.

<table>
<thead>
<tr>
<th>...+....1....+....2....+....3....+....4....+....5....+....6....+....7....</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSCPID=01,</td>
</tr>
<tr>
<td>SSCPNAME=SSCP1A</td>
</tr>
<tr>
<td>NETID=NETA,</td>
</tr>
<tr>
<td>DYNASSCP=YES</td>
</tr>
<tr>
<td>SSCPOYN=YES,</td>
</tr>
<tr>
<td>HOSTSA=1</td>
</tr>
</tbody>
</table>

One start list cannot reference another, and only an operator can enter LIST=yy on the start command to use this start list.

Depending on the number of items, two or more records might be required for the start option list. Observe the following rules if more than one record is required:

- You can code data through position 71, even if you are in the middle of an item. The remainder of the item can be continued in the next record.
- If you do not code data through position 71, end the data in one record with an item, and begin the next record with a new item.
- To create a comment line, enter an asterisk (*) in position 1 of any record.
Using MVS System Symbols in VTAM Start Options

You can use the following MVS system symbols in the values you specify for VTAM start options:

&SYSCLONE
  Represents a 1- or 2-character shorthand notation for the name of the system

&SYSNAME
  Represents a 1–8-character system name

&SYSPLEX
  Represents a 1–8-character sysplex name

&installation-defined system symbol
  Represents a 1–8-character identifier. You can define this symbol to be any identifier you want.

The substitution text for these system symbols is defined in MVS and is substituted by MVS in place of the symbols during VTAM startup. See MVS Initialization and Tuning Reference for information about how to define the symbols in MVS.

MVS system symbols can be used to allow a single start option list to be used to start multiple VTAMs in a multisystem environment. For information about how to use MVS system symbols in VTAMLST, see z/OS Communications Server: SNA Network Implementation Guide.

Start options syntax diagrams
Options:

- **NETID**
- **SSCPID**
- **SSCPNAME**
- **AFFDELAY**
- **ALSREQ**
- **API64R**
- **APPNCOS**
- **ASIRFMSG**
- **ASYDE**
- **AUTHLEN**
- **AUTOTI**
- **BNORD**
- **BSCTMOUT**
- **CACHETI**
- **CDRSCTI**
- **CDSRVR**
- **CDSREFER**
- **CINDXSIZ**

1. **APPNCOS**
2. **AUTOTI**
3. **BN**
4. **BNORD**
5. **BSCTMOUT**
6. **CDSREFER**
INOPCODE = (ALL, ALL, DUMPDEFAULT)

INOPDUMP = ON

INOPDUMP = OFF

INOPDUMP = OFF

IOINT = 180

IOINT = number_of_seconds

IOMSGLIM = 100

IOMSGLIM = number_of_message_pairs

IPADDR = IP_address

IPINFO = SENDALL

IPINFO = SENDALL

IRNSTRGE = MAX

IRNSTRGE = size

ISTCOSDF = INDLU

ISTCOSDF = ALL

LIMINTCP = number_of_seconds

LIST = 00

LIST = start_option_list_id

LISTBKUP = start_option_list_id

LISTBKUP = DEFAULTS

LISTBKUP = PROMPT

MAXHNRES = 100

MAXHNRES = max_hostname_resolutions

MAXEETST = 500

MAXEETST = max_ee_connectivity_tests
Notes:
1. APPNCOS is meaningful only if the NODETYPE start option is also used.
2. BN is meaningful only if the NODETYPE=NN start option is also used.
3. BNDYN is meaningful only if the BN=YES start option is also used.
4. BNORD is meaningful only if the BN=YES start option is also used.
5. CDSERVR is meaningful only if the NODETYPE=NN start option is also used.
6. CDSREFER is meaningful only if the NODETYPE=NN and CDSERVR=NO start options are also used.
7. The CMPMIPS start option is meaningful only if the value for CMPVTAM is greater than 1.
8. CONNTYPE is meaningful only if the NODETYPE start option is also used.
9. CPCP is meaningful only if the NODETYPE start option is also used.
10. Specify the CSDUMP start option twice to set both message and sense code triggers.
11. DIRSIZE is meaningful only if the NODETYPE=NN start option is also used.
12. DIRTIME is meaningful only if the NODETYPE=NN start option is also used.
13. DLURSAW is meaningful only if the NODETYPE=NN start option is also used.
14. If the DSPLYMAX start option value is less than 100, that value is the default for DSPLYDEF.
15. DYNADJCP is meaningful only if the NODETYPE start option is also used.
16. Two character prefix.
17. EEHPRANR is meaningful only when the NODETYPE=NN start option is also used.
18. The EEVERIFY start option is meaningful only if VTAM provides RTP-level
HPR support. The NODETYPE start option must be coded and the RTP value must be specified on the HPR start option.

ENCRIPTN=CCA needs to be coded when Triple Des Encryption is desired.

The ENSEMBLE setting is used to either permit or deny connectivity to the intraensemble data network (IEDN) and the intranode management network (INMN) by allowing or denying activation of OSX and OSM interfaces.

HOSTNAME is meaningful only if the NODETYPE start option is also used.

HOSTSA specifies the subarea number of this VTAM. If HOSTSA is not coded, then a default subarea number of 1 is used.

HPR is meaningful only if NODETYPE is also used.

This start option was provided by APAR OW36113 for use in migration to VTAM V4R5. Do not use this option unless you use the default value of RESPMODE.

HPRCLKRT=ADAPTIVE is meaningful only for Enterprise Extender configurations that have a defined capacity of 1 Gb or higher access speeds.

This option is meaningful only if VTAM provides RTP-level HPR support.

This option is meaningful only if VTAM provides RTP-level HPR support.

This option is meaningful only if VTAM provides RTP-level HPR support.

This option is meaningful only if VTAM provides RTP-level HPR support.

HPRSESLSM=DISABLED is meaningful only on interchange nodes.

This option is meaningful only if VTAM provides RTP-level HPR support.

INITDB is meaningful only if the NODETYPE=NN start option is also used.

INOPCODE has no effect unless INOPDUMP is active for the resource when an inoperative condition is detected.

INOPTH status is propagated to resources that are defined within a transport resource list entry when the entry is activated and the TRLE InOpDump status has not been explicitly set.

The INOPCODE start option provides more granular control of the INOPDUMP function. Refer to the INOPCODE in this section and the DISPLAY INOPCODE command in z/OS Communications Server: SNA Operation for additional details.

IPADDR is meaningful only if the NODETYPE start option is also used.

The IQDCHPID option controls which IQD CHPID (and related subchannel devices) VTAM selects to dynamically build the iQDIO (IUTIQDIO) MPC group. The IUTIQDIO MPC group is used for TCP/IP dynamic XCF communications within this System z system. Although this option can be modified (and the modification will immediately be displayed) while the IUTIQDIO MPC group is currently active, any modifications will have the effects: 1) Modified from ANY (or CHPID) to NONE has no effect on current usage but blocks subsequent activations; 2) Modified from NONE to ANY (or CHPID) has no effect on current usage but allows subsequent activations; 3) Modified from CHPID_X to CHPID_Y has no effect on current usage. VTAM only uses the CHPID value when building the IUTIQDIO MPC group. To change CHPIDs for an active MPC group, the steps must be done: 1) All TCP/IP iQDIO (HiperSockets™) devices must be stopped; 2) Make any necessary HCD/IOCDS changes; 3) Verify that new subchannel devices are
varied online; 4) Verify that the MPC group has deactivated (with no usage, it times out after approximately two minutes); 5) Modify IQDCHPID=chpid (to new CHPID); 6) Restart the TCP/IP iQDIO device or devices. In order to use iQDIO communications, the processor must have the necessary hardware support. If the processor does not support iQDIO communications, then modifications to this start option will not be accepted and the IQDCHPID option will not be displayed (displayed as ***NA***).

This option only affects iQDIO devices that use a MFS of 64k. The smaller frame sizes will always use 126 SBALs.

LIST can be entered by a VTAM operator only. If LIST is coded in an ATCSTR\text{xx} file, it is considered to be an error and is ignored.

LISTBKUP can be coded in a start option file only. If you enter it on the START command or at an operator prompt, VTAM will ignore it.

MAXLOCAT is meaningful only if NODTYPE is specified.

MULTPATH is meaningful only if the NODTYPE start option is also specified.

This option only affects iQDIO devices that use a MFS of 64k. The smaller frame sizes will always use 126 SBALs.

LIST can be entered by a VTAM operator only. If LIST is coded in an ATCSTR\text{xx} file, it is considered to be an error and is ignored.

LISTBKUP can be coded in a start option file only. If you enter it on the START command or at an operator prompt, VTAM will ignore it.

MAXLOCAT is meaningful only if NODTYPE is specified.

MULTPATH is meaningful only if the NODTYPE start option is also specified.
Notes:

1. NNSPREF can be specified only if NODETYPE=EN is specified during VTAM START processing.

2. NODETYPE enables APPN function. The combination of HOSTSA, NODETYPE, and SACONNS determines the configuration (subarea node, interchange node, migration data host, network node, or end node).

3. NUMTREES is meaningful only if the NODETYPE=NN start option is also used.

4. PMTUD is meaningful only if the NODETYPE start option is also used.

5. A VTAM operator cannot enter the PROMPT or NOPROMPT start option; it can be coded only in ATCSTR00. The value coded in ATCSTR00 is ignored if start options are entered on the START command or if VTAM finds an error in a start list. Upon finding an error in a start list, VTAM prompts the operator so that the operator can specify the option correctly.

6. QDIOSTG defaults to MAX for 64-bit (z/Architecture®) machines and MIN for non 64-bit machines.

7. RESUSAGE is meaningful only if the NODETYPE=NN start option is also used.

8. ROUTERES is meaningful only if the NODETYPE=NN start option is also used.
SONLIM = (-60, -30)

SONLIM = (-percent_1, -percent_2)
   (-percent_1)
   (-percent_2)

SORDER = APPN
SORDER = ADJSSCP
   APPN
   APPNFRST
   SUBAREA

SRCHRED = OFF

SRCOUNT = 10
SRCOUNT = number_of_search_requests

SSCDYN = YES
SSCDYN = NO
   YES

SSCPORD = PRIORITY
SSCPORD = DEFINED
   PRIORITY

SSDTMOUT = 30
SSDTMOUT = number_of_seconds

SSEARCH = YES

STRGR = ISTGENERIC
   ISTGENERIC
   NONE
   structure_name

STRMNPS = ISTMNPS
   ISTMNPS
   NONE
   structure_name

SUPP = NOSUP
SUPP = NOSUP
   NOSUP
   INFO
   WARN
   NORM
   SER

SWNORDER = (CPNAME, FIRST)
   CPNAME
   STATID
   (.FIRST)
   (.ONLY)
   (CPNAME, FIRST)
   (STATID, FIRST)
   (STATID, ONLY)

TCPNAME = tcp_job-name

TDUDIAG = 1000
   resource_threshold
   ALWAYS
   NEVER

Chapter 4. VTAM start options 767
Notes:
1 SAVERSCV is meaningful only if NODETYPE is also used.
2 The SECLVLCP start option is meaningful only if the NODETYPE and VERIFYCP start options are also used.
3 SNVC is meaningful only if the BN=YES start option is also used.
4 SORDER is meaningful only in an interchange node or a migration data host.
5 SRCOUNT is meaningful only if the SRCHRED=ON start option is also used.
6 SRTIMER is meaningful only if the SRCHRED=ON start option is also used.
7 SSEARCH is meaningful only if the NODETYPE=NN start option is also used.
8 TCPNAME is meaningful only if the NODETYPE start option is also used.
9 TDUDIAG is meaningful only if the NODETYPE=NN start option is also used.
10 TOPOTIME is meaningful only if the NODETYPE start option is also used.
CSDUMP message trigger:

IO trace operands:

LINE trace operands:

MODULE operands:
SIT trace operands:

\[
\begin{align*}
\text{ID} &= \text{line_name} \\
\text{COUNT} &= \text{ALL}^6 \\
\text{number_of_bytes} \\
\text{TRACEPT} &= \text{trace_point_id}
\end{align*}
\]

STATE trace operands:

\[
\begin{align*}
\text{ID} &= \text{node_name} \\
\text{IDTYPE} &= \text{RESOURCE} \\
\text{OPTION} &= \text{all} \\
\text{option} \\
\text{ADJCP} \\
\text{APPL} \\
\text{CDRM} \\
\text{CDRSC} \\
\text{GROUP} \\
\text{LINE} \\
\text{LNKS} \\
\text{LU} \\
\text{NCP} \\
\text{PU}
\end{align*}
\]

VIT operands:

\[
\begin{align*}
\text{MODE} &= \text{INT}, \text{SIZE} = \text{INT} \\
\text{MODE} &= \text{EXT} \\
\text{BFRNUM} \\
\text{INT} \\
\text{SIZE} = \text{size}
\end{align*}
\]
Notes:

1. UNRCHTIM is meaningful only if the NODETYPE start option is also used.
2. UPDDELAY is meaningful only if the OSIMGMT=YES start option is also used.
3. The VERIFYCP start option is meaningful only if the NODETYPE start option is also used.
4. VFYREDTI is meaningful only if the NODETYPE=NN start option is also used.
5. VRTG is meaningful only if the NODETYPE and HOSTSA start options are also used.
6 VRTGCPCP is meaningful only if the NODETYPE and HOSTSA start options are also used.

7 XCFINIT=YES is the default if VTAM is started as an APPN node (that is, the NODETYPE start option has been specified). XCFINIT=YES is not allowed for pure subarea nodes. XCFINIT=DEFINE is the default if VTAM is started as a pure subarea node (the NODETYPE start option has not been specified).

8 The IOBUF pool (IO00, IO) is the only buffer pool where all seven values can be specified. For all other buffer pools, the xpanlim field is not supported. If you specify the xpanlim field for any buffer pool other than the IOBUF pool (IO00, IO), even if the field is null, you get an IST1072I message.

9 The default options apply only to MODE=INT.

10 PSS and SMS can be turned off.

11 When SUBTRACE=ARBP is specified, if a single OPTION value is coded, it must be HPR, ALL, or one of the group options (hprgrpopt) that include HPR as an individual option equivalent. The applicable group options are DLUROPTS, EEOPTS, HPDTOPTS, HPROPTS, QDIOOPTS, and XCFOPTS.

12 If multiple trace options are coded in parentheses, either HPR or one of the group options (hprgrpopt) that include HPR as an individual option equivalent must be coded inside the parentheses when SUBTRACE=ARBP is coded.

13 When you specify SUBTRACE=DIO and you code a single OPTION value, the OPTION value must be CIA, ALL, or one of the group options (ciaggrpopt) that include CIA as an individual option equivalent. The applicable group options are EEOPTS, HPDTOPTS, HPROPTS, QDIOOPTS, TCPOPTS, and XCFOPTS.

14 When SUBTRACE=DIO is coded and you code multiple trace options in parentheses, you must code either CIA or one of the group options (ciaggrpopt) that include CIA as an individual option equivalent inside the parentheses.

15 When SUBTRACE=TGVC or SUBTRACE=TREE is coded, if a single OPTION value is coded, it must be SSCP, ALL, or one of the group options (groupopt), all of which include SSCP as an individual option equivalent. The group options are APIOPTS, APPCOPTS, CPCPOPTS, CSMOPTS, DLUROPTS, EEOPTS, HPDTOPTS, HPROPTS, LCSOPTS, QDIOOPTS, STDOPTS, TCPOPTS, and XCFOPTS.

16 If multiple trace options are coded in parentheses, either SSCP or one of the group options (groupopt) must be coded inside the parentheses when SUBTRACE=TGVC or SUBTRACE=TREE is coded.

**Summary of start options**

The tables in this information show the start options organized by function. Each table includes a brief description of the start option.

The tables show the start options that define or affect:

- SSCP or CP characteristics  [Table 51 on page 775]
- VTAM initialization  [Table 52 on page 776]
- Storage  [Table 53 on page 776]
- Session control  [Table 54 on page 777]
- Connectivity  [Table 55 on page 779]
- Session security  [Table 56 on page 780]
• Messages (Table 57 on page 780)
• Recording and statistics (Table 58 on page 782)
• Traces and dumps (Table 59 on page 782)
• Buffer pools (Table 60 on page 783)
• Performance and tuning (Table 61 on page 783)
• APPN characteristics (Table 62 on page 786)
• Functions from zappable constants (Table 63 on page 789)
• HPR and Enterprise Extender (Table 64 on page 791)
• Network management (Table 65 on page 792)
• Coupling facility and SYSPLEX (Table 66 on page 793)

See “Descriptions of start options” on page 794 for more detailed descriptions of the start options.

SSCP or CP characteristics start options

Table 51. Start options that define SSCP or CP characteristics. These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=SSCPCP command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BN</td>
<td>Whether this node is to provide extended border node function</td>
</tr>
<tr>
<td>CDSERVR</td>
<td>Whether this node is a central directory server</td>
</tr>
<tr>
<td>DATEFORM</td>
<td>The format that VTAM uses for dates</td>
</tr>
<tr>
<td>ENHADDR</td>
<td>Whether high-order element addressing for PLUs can be used.</td>
</tr>
<tr>
<td>GWSSCP</td>
<td>Whether VTAM is a gateway SSCP</td>
</tr>
<tr>
<td>HOSTPU</td>
<td>The name of the host subarea PU</td>
</tr>
<tr>
<td>HOSTSA</td>
<td>The host subarea number</td>
</tr>
<tr>
<td>LIMINTCP</td>
<td>The interval to retain a free SNASVCMG session on limited resources for network management flows</td>
</tr>
<tr>
<td>MAINTLVL</td>
<td>The maintenance level of the host as defined by the user</td>
</tr>
<tr>
<td>MAXSSCPS</td>
<td>The maximum number of SSCPs that are searched on a specific path for the destination SSCP before alternate paths are tried</td>
</tr>
<tr>
<td>MAXSUBA</td>
<td>The highest subarea number that can be assigned to any node in this network</td>
</tr>
<tr>
<td>MXSUBNUM</td>
<td>The maximum subarea number supported by a network to which a VTAM host is interconnected</td>
</tr>
<tr>
<td>NETID</td>
<td>The name of the network containing the host</td>
</tr>
<tr>
<td>NNSPREF</td>
<td>Specifies the name of the preferred network node server for this end node</td>
</tr>
<tr>
<td>NODETYPE</td>
<td>The APPN node type (network node or end node)</td>
</tr>
<tr>
<td>NQNMODE</td>
<td>Whether VTAM defines cross-network resources by their non-network-qualified names in addition to their network-qualified names</td>
</tr>
<tr>
<td>SSCPID</td>
<td>The SSCP identifier used when a PU or external CDRM establishes contact with VTAM</td>
</tr>
<tr>
<td>SSCPNAME</td>
<td>The name of the VTAM SSCP or CP</td>
</tr>
<tr>
<td>STRGR</td>
<td>The name of the generic resources structure</td>
</tr>
</tbody>
</table>
Table 51. Start options that define SSCP or CP characteristics (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=SSCPCP command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STRMNPS</td>
<td>The name of the structure for multinode persistent sessions</td>
</tr>
<tr>
<td>TRANSLAT</td>
<td>Which names are to be translated by alias name translation</td>
</tr>
<tr>
<td>USSTAB</td>
<td>The name of the USS table for operator commands and messages</td>
</tr>
</tbody>
</table>

VTAM initialization start options

Table 52. Start options that affect VTAM initialization. These start options (excluding LISTBKUP, NOPROMPT, and PROMPT) can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=VTAMINIT command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLD</td>
<td>On starting VTAM, restore major nodes to their initial status as defined by the user</td>
</tr>
<tr>
<td>CONFIG</td>
<td>The list of major nodes to activate when VTAM is started</td>
</tr>
<tr>
<td>INITDB</td>
<td>Whether to load the APPN directory database, topology database, neither, or both when VTAM is started</td>
</tr>
<tr>
<td>LIST</td>
<td>The ID of the start option list to use for starting VTAM</td>
</tr>
<tr>
<td>LISTBKUP</td>
<td>The action that VTAM should take if an error is found in a start option list</td>
</tr>
<tr>
<td>NODELST</td>
<td>The name of the configuration restart file</td>
</tr>
<tr>
<td>NOPROMPT</td>
<td>Do not prompt the operator to enter start options</td>
</tr>
<tr>
<td>OSIMGMT</td>
<td>Indicates support of CMIP application programs on VTAM, including the VTAM topology agent</td>
</tr>
<tr>
<td>PROMPT</td>
<td>Prompt the operator to enter start options</td>
</tr>
<tr>
<td>SNAMGMT</td>
<td>Indicates support of SNA Network Management Interface, including the attachment of the ISTMGCEH subtask which acts as the socket server for the interface</td>
</tr>
<tr>
<td>WARM</td>
<td>On starting VTAM, restore major nodes to their status before VTAM's last termination, as recorded in the configuration restart file</td>
</tr>
</tbody>
</table>

Storage usage start options

Table 53. Start options that define storage usage. These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=STORAGE command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API64R</td>
<td>Determines whether 64-bit backed storage can be passed to applications</td>
</tr>
<tr>
<td>CSALIMIT</td>
<td>The maximum amount of common service area (CSA) that can be used by VTAM</td>
</tr>
<tr>
<td>CSA24</td>
<td>The maximum amount of 24-bit addressable CSA that can be used by VTAM</td>
</tr>
<tr>
<td>DLRTCB</td>
<td>The maximum number of task control blocks that VTAM allocates in private storage for various functions</td>
</tr>
</tbody>
</table>
Table 53. Start options that define storage usage (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=STORAGE command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRNSTRGE</td>
<td>The maximum amount of VTAM private storage that can be used for host intermediate routing node (IRN) transmissions</td>
</tr>
<tr>
<td>MAXHNRES</td>
<td>The maximum number of Enterprise Extender (EE) host name resolution requests that can run concurrently</td>
</tr>
<tr>
<td>VTAMEAS</td>
<td>The size of the table that contains information about sessions with an application program</td>
</tr>
</tbody>
</table>

### Session control start options

Table 54. Start options that affect session control. These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=SESSCONT command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFFDELAY</td>
<td>Specifies the number of seconds after a session termination that Generic Resource affinities for non-syncpoint, non-Limited Resource LU 6.2 sessions should be retained</td>
</tr>
<tr>
<td>APPNCOS</td>
<td>The APPN Class of Service that is used if the requested Class of Service cannot be found</td>
</tr>
<tr>
<td>ASYDE</td>
<td>Whether to terminate a session when an asynchronous device end RU is received from a local non-SNA device</td>
</tr>
<tr>
<td>AUTORTRY</td>
<td>Specifies which adjacent node activation will cause a retry of pending automatic logon requests</td>
</tr>
<tr>
<td>BNDYN</td>
<td>Indicates how nodes are added dynamically to adjacent cluster routing lists</td>
</tr>
<tr>
<td>BNORD</td>
<td>Indicates the order VTAM does cross-subnet searches</td>
</tr>
<tr>
<td>CDRDYN</td>
<td>Specifies whether the host is authorized to dynamically define CDRSC representations of cross-domain resources</td>
</tr>
<tr>
<td>CMPMIPS</td>
<td>Use to determine the amount of time for adaptive mode compression</td>
</tr>
<tr>
<td>CMPVTAM</td>
<td>The data compression level to be used for sessions with the application programs running on the host</td>
</tr>
<tr>
<td>CPCDRSC</td>
<td>Whether VTAM allows applications or LUs to initiate and establish sessions to a dynamic LEN CP ILU</td>
</tr>
<tr>
<td>DIALRTRY</td>
<td>Specifies whether VTAM should continue searching for a switched resource after a dial-out failure</td>
</tr>
<tr>
<td>DLRORDER</td>
<td>The way VTAM locates a DLUR PU</td>
</tr>
<tr>
<td>DUPDEFS</td>
<td>Whether APPN should continue searching for the target resource when the resource has been found but is not active</td>
</tr>
<tr>
<td>DYNADJCP</td>
<td>Whether dynamic entries are created in ISTADJCP for adjacent APPN nodes</td>
</tr>
<tr>
<td>DYNASSCP</td>
<td>Whether VTAM dynamically routes session establishment requests to all active adjacent SSCP tables if no appropriate adjacent SSCP table is defined</td>
</tr>
<tr>
<td>DYNDLGMD</td>
<td>Specifies the default logon mode name used for dynamic cross-domain resources when the resource is acting as the secondary logical unit (SLU) and a logon mode name is not otherwise provided</td>
</tr>
</tbody>
</table>
Table 54. Start options that affect session control (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=SESSCONT command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYNLU</td>
<td>Whether independent LUs can be dynamically defined</td>
</tr>
<tr>
<td>DYNMODTB</td>
<td>Specifies the name of a logon mode table used for dynamic cross-domain resources to correlate each logon mode name with a set of session parameters when the resource acts as the SLU</td>
</tr>
<tr>
<td>EXPFLTRM</td>
<td>The EXPFLTRM start option defines the number of expedited data flow control (DFC) and session control (SC) request units (RU), that VTAM allows a boundary-attached LU to send without receiving a response from the partner LU.</td>
</tr>
<tr>
<td>HOTIOTRM</td>
<td>The percentage of the I/O buffer pool that can be used by one session before the amount is considered to be excessive and the session is terminated</td>
</tr>
<tr>
<td>HPRSESLM</td>
<td>Specifies the maximum number of sessions that should be assigned to a single RTP pipe</td>
</tr>
<tr>
<td>ISTCOSDF</td>
<td>Indicates the resource types that can use the ISTCOSDF logmode entry</td>
</tr>
<tr>
<td>RESUSAGE</td>
<td>The number of times an APPN resource (node or TG) can be used for routing before it is considered overused</td>
</tr>
<tr>
<td>ROUTERES</td>
<td>The relative amount of resistance to having this APPN node perform intermediate session routing</td>
</tr>
<tr>
<td>SMEAUTH</td>
<td>Controls session management exit (SME) authorization data without having to modify or code an SME</td>
</tr>
<tr>
<td>SORDER</td>
<td>The order in which to search APPN and subarea networks when a network search request is received from a subarea network</td>
</tr>
<tr>
<td>SSCPdyn</td>
<td>Whether VTAM can dynamically add entries to an adjacent SSCP table</td>
</tr>
<tr>
<td>SSCPORD</td>
<td>The order in which VTAM searches an adjacent SSCP table</td>
</tr>
<tr>
<td>SSEARCH</td>
<td>Whether to search a subarea network when a network search request is received from an APPN network</td>
</tr>
<tr>
<td>SWNORDER</td>
<td>The way VTAM locates a switched PU</td>
</tr>
<tr>
<td>UNRCHTIM</td>
<td>Specifies the default number of seconds that a partner node for a session path through an Enterprise Extender connection network is considered unreachable after certain Enterprise Extender connection network failures. During this period, the path through the Enterprise Extender connection network to this partner node will not be considered for new sessions or HPR path switches. UNRCHTIM also specifies the limit for the number of unreachability records associated with an Enterprise Extender connection network before that connection network is considered unusable for session route selection for new sessions and HPR path switches.</td>
</tr>
</tbody>
</table>
**Connectivity start options**

*Table 55. Start options that affect connectivity.* These start options affect session establishment, termination, or search order, and can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=CONNECT command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALSREQ</td>
<td>Whether adjacent link station names are required to be in the default ALS list of independent LUs</td>
</tr>
<tr>
<td>AUTHLEN</td>
<td>Whether VTAM is enabled to pass transmission priority from one LEN node to another</td>
</tr>
<tr>
<td>CONNTYPE</td>
<td>Whether a LEN connection or an APPN connection is established with type 2.1 PUs</td>
</tr>
<tr>
<td>CPCP</td>
<td>The types of connections on which CP-CP sessions are allowed with adjacent nodes</td>
</tr>
<tr>
<td>DISCNTIM</td>
<td>Amount of time VTAM delays deactivation of SSCP-PU session when there are no outstanding LU-LU session requests</td>
</tr>
<tr>
<td>DYNHPPFX</td>
<td>Indicates if DYNHPPFX is not specified, then dynamic PUs created for HPR will be prefixed with the three characters, CNR. If DYNHPPFX=cc where cc is the two character prefix that you want, then dynamic PUs created for HPR will be prefixed with the two characters, cc. In these two cases, respectively, the dynamic PU names will have the format CNRxxxxx or cxxxxxx.</td>
</tr>
<tr>
<td>DYNPUPFX</td>
<td>Specifies a two character prefix that is used in creating the dynamic PUs. This prefix will be used when generating names for these dynamically built PUs. The prefix defined by the start option will be used if DYNPUPFX is not coded on the group statement for the line under which the PU is built. A unique value will then be generated and concatenated to the defined prefix to create an eight character PU name.</td>
</tr>
<tr>
<td>DYNVNPFX</td>
<td>Specifies the prefix to be used when generating the name for dynamic connection network PUs</td>
</tr>
<tr>
<td>ENSEMBLE</td>
<td>Specifies whether this z/OS Communications Server will permit connectivity to the intraensemble data network (IEDN) and to the intranode management network (INMN). Access to these networks is through OSA-Express3 or later features that are configured in OSX and OSM modes.</td>
</tr>
<tr>
<td>HPR</td>
<td>Specifies the level of High-Performance Routing (HPR) support provided by VTAM</td>
</tr>
<tr>
<td>HPRNCPBF</td>
<td>Specifies whether HPR can be used although it may cause session data to travel through an NCP twice</td>
</tr>
<tr>
<td>IQDCHPID</td>
<td>Specifies superfast connectivity replacing XCF links</td>
</tr>
<tr>
<td>MPCACT</td>
<td>Specifies whether activations of MPC subchannel groups are to be suspended if the minimum number of read and write subchannels is not available, either because they are offline or because no valid path exists to the connecting host. VTAM automatically resumes activation when the minimum number of subchannels becomes available. This option applies only to MPC groups consisting of CTC connections.</td>
</tr>
<tr>
<td>NNSPREF</td>
<td>Specifies the name of the preferred network node server for this end node</td>
</tr>
<tr>
<td>SACONNS</td>
<td>Specifies if subarea connections will be used</td>
</tr>
</tbody>
</table>
Table 55. Start options that affect connectivity (continued). These start options affect session establishment, termination, or search order, and can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=CONNECT command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLOWVAL</td>
<td>Specifies that a network node which has been placed in a SLOWDOWN mode can now allow a maximum SLOWDOWN time to be set</td>
</tr>
<tr>
<td>SNVC</td>
<td>Maximum number of networks searched for a resource by this border node</td>
</tr>
<tr>
<td>SSDTMOUT</td>
<td>The amount of time that a switched PU can go without a session before the connection is dropped</td>
</tr>
<tr>
<td>VRTG</td>
<td>Whether virtual-route-based transmission groups connections are requested when SSCP-SSCP session is established</td>
</tr>
<tr>
<td>VRTGCPCP</td>
<td>Whether CP-CP sessions are supported over the virtual-route-based transmission group</td>
</tr>
<tr>
<td>XCFGRPID</td>
<td>Specifies the group ID suffix that VTAM will use when joining groups in the sysplex. VTAM will join groups ISTXCFvv and ISTCFSvv, where vv is the value provided with XCFGRPID</td>
</tr>
<tr>
<td>XCFINIT</td>
<td>Specifies whether VTAM will join a sysplex group at VTAM initialization and establish the capability for XCF connectivity to other VTAM nodes in the sysplex</td>
</tr>
<tr>
<td>XNETALS</td>
<td>Whether VTAM can establish sessions with LUs in adjacent networks</td>
</tr>
</tbody>
</table>

Session security start options

Table 56. Start options that affect session security. These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=SECURITY command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENCRPREF</td>
<td>The prefix to be used for common cryptographic architecture (CCA) key-encrypting key (KEK) labels</td>
</tr>
<tr>
<td>ENCRYPTN</td>
<td>VTAM’s level of cryptography support</td>
</tr>
<tr>
<td>IPINFO</td>
<td>Provides control over the displaying and passing IP characteristics received in this host.</td>
</tr>
<tr>
<td>SECLVLCP</td>
<td>Whether VTAM uses basic or enhanced protocol for session-level security verification during activation of LU 6.2 sessions involving APPN control points</td>
</tr>
<tr>
<td>VERIFYCP</td>
<td>Whether VTAM performs LU-LU session-level verification during activation of LU 6.2 sessions involving APPN control points</td>
</tr>
</tbody>
</table>

Messages start options

Table 57. Start options that affect messages. These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=MESSAGES command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASIRFMSG</td>
<td>The SSCP’s issue messages IST890I and IST896I when an autologon session initiation request fails</td>
</tr>
</tbody>
</table>
Table 57. **Start options that affect messages (continued).** These start options can be displayed as a group with the `DISPLAY VTAMOPTS,FUNCTION=MESSAGES` command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNMTAB</td>
<td>Specifies the name of the communication network management (CNM) routing table used to route unsolicited request units</td>
</tr>
<tr>
<td>CNNRTMSG</td>
<td>Specifies whether VTAM is to suppress messages when an optimal CNN route exists and is not chosen during session activation</td>
</tr>
<tr>
<td>DSIRFMSG</td>
<td>The SCCPs issue messages IST663I, IST664I, IST889I, and their message groups when searches that do not result in a session fail to locate the target resource</td>
</tr>
<tr>
<td>DSPLYDEF</td>
<td>Limits the number of messages displayed when certain DISPLAY commands are issued without the MAX operand</td>
</tr>
<tr>
<td>DSPLYMAX</td>
<td>Limits the value that can be specified for the DSPLYDEF start option and for the MAX operand on certain DISPLAY commands</td>
</tr>
<tr>
<td>DSPLYWLD</td>
<td>Whether wildcards are permitted in certain DISPLAY commands</td>
</tr>
<tr>
<td>ESIRFMSG</td>
<td>The SCCPs issue messages IST891I, IST892I, and IST893I when a session initiation request fails and extended sense data exists</td>
</tr>
<tr>
<td>FLDTAB</td>
<td>Specifies whether VTAM uses a message-flooding prevention table to suppress duplicate messages</td>
</tr>
<tr>
<td>FSIRFMSG</td>
<td>The SCCPs issue messages IST894I, IST895I, IST1704I, and IST1705I when a session initiation request fails because VTAM is unable to locate the destination LU by using an adjacent SSCP table</td>
</tr>
<tr>
<td>HPRITMSG</td>
<td>Specifies whether VTAM should issue the enhanced version or base version of HPR activation and deactivation messages</td>
</tr>
<tr>
<td>HPRPSMSG</td>
<td>Specifies whether VTAM should limit the number of HPR path switch messages that are issued</td>
</tr>
<tr>
<td>IOINT</td>
<td>The interval after which the VTAM operator is notified of outstanding I/O requests</td>
</tr>
<tr>
<td>IOMSGLIM</td>
<td>The maximum number of IST530I and IST532I message pairs that are written for individual subareas by the IOPD facility</td>
</tr>
<tr>
<td>LSIRFMSG</td>
<td>The network nodes issue messages IST663I, IST664I, IST889I, and subordinate messages when APPN locate searches fail to locate the target resource.</td>
</tr>
<tr>
<td>MSGLEVEL</td>
<td>Whether VTAM is to issue the level of messages containing network-qualified names</td>
</tr>
<tr>
<td>MSGMOD</td>
<td>Specifies whether VTAM is to insert in messages the name of the module that issued the message</td>
</tr>
<tr>
<td>PLUALMSG</td>
<td>Specifies whether to suppress duplicate session setup failure messages associated with LOGAPPL failures to the same PLU</td>
</tr>
<tr>
<td>PPOLOG</td>
<td>Specifies whether to send operator commands and their resulting messages to the primary program operator for logging</td>
</tr>
<tr>
<td>RSIRFMSG</td>
<td>The SCCPs issue messages IST1460I, IST1461I, IST2102I, IST2103I, and IST2104I when a session initiation request fails and RSCV data exists</td>
</tr>
</tbody>
</table>
### Start options that affect messages (continued)

These start options can be displayed as a group with the `DISPLAY VTAMOPTS,FUNCTION=MESSAGES` command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIRFMSG</td>
<td>The SSCP's issue IST663I, IST664I, IST889I, and subordinate messages when a session initiation request fails.</td>
</tr>
<tr>
<td>SLOWVAL</td>
<td>The interval after which the VTAM operator is notified that a resource is in SLOWDOWN mode or is taken out of SLOWDOWN mode by VTAM.</td>
</tr>
<tr>
<td>SLUALMSG</td>
<td>Specifies whether to suppress messages for session setup failures that occur because the SLU is not enabled.</td>
</tr>
<tr>
<td>SUPP</td>
<td>The message suppression level for the operator console.</td>
</tr>
<tr>
<td>VARYWLD</td>
<td>Specifies whether wildcards are permitted in certain VARY commands.</td>
</tr>
</tbody>
</table>

### Recording and statistics start options

These start options can be displayed as a group with the `DISPLAY VTAMOPTS,FUNCTION=RECSTATS` command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSCMDRS</td>
<td>Which BSC miscellaneous data records are to be recorded in LOGREC.</td>
</tr>
<tr>
<td>NMVTLOG</td>
<td>Whether to record NMVT alerts in LOGREC.</td>
</tr>
<tr>
<td>PPOLOG</td>
<td>Whether to send operator commands and their resulting messages to the primary program operator for logging.</td>
</tr>
<tr>
<td>SDLCMDRS</td>
<td>Which SDLC miscellaneous data records are to be recorded in LOGREC.</td>
</tr>
<tr>
<td>TNSTAT</td>
<td>Whether global tuning statistics are being recorded.</td>
</tr>
</tbody>
</table>

### Traces and dumps start options

These start options (excluding `CSDUMP`, `INOPCODE`, and `TRACE`) can be displayed as a group with the `DISPLAY VTAMOPTS,FUNCTION=TRACDUMP` command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSDUMP</td>
<td>Specifies a sense code trigger, message trigger, or both, that will cause a dump to be taken when the sense code or the message is issued.</td>
</tr>
<tr>
<td>INOPCODE</td>
<td>Whether to dump VTAM for specific INOP conditions when INOPDUMP is enabled for the resource.</td>
</tr>
<tr>
<td>INOPDUMP</td>
<td>Whether a resource is eligible to initiate a VTAM dump under certain INOP conditions.</td>
</tr>
<tr>
<td>PSSTRACE</td>
<td>Whether to record IRB and SRB entries in the VTAM internal trace table when the PSS trace option is in effect.</td>
</tr>
<tr>
<td>SNAPREQ</td>
<td>The number of requests for VTAM buffers between snapshot dumps.</td>
</tr>
<tr>
<td>TRACE</td>
<td>Which traces to start at VTAM initialization.</td>
</tr>
</tbody>
</table>
Table 59. Start options that affect traces and dumps (continued). These start options (excluding CSDUMP, INOPCODE, and TRACE) can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=TRACDUMP command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NACPROBE</td>
<td>Specifies whether VTAM takes an FFST™ probe dump when VTAM does not complete the CLOSE ACB processing for an application timely. The title of the dump is “NACCT – CLOSE TIMER EXPIRED FOR applname”. Alternatively, you can use this option to specify the time interval at which VTAM takes an FFST probe dump when VTAM does not complete the CLOSE ACB processing for another application timely.</td>
</tr>
</tbody>
</table>

Buffer pools start options

Table 60. Start options that define buffer pools. These start options can be displayed as a group with the DISPLAY BFRUSE command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APBUF</td>
<td>The buffer pool that provides fixed common storage with 24-bit addressable buffers</td>
</tr>
<tr>
<td>BSBUF</td>
<td>The buffer pool that provides boundary session control block buffers</td>
</tr>
<tr>
<td>CRA4BUF</td>
<td>The buffer pool that provides copied request at 4080 addressable buffer size</td>
</tr>
<tr>
<td>CRA8BUF</td>
<td>The buffer pool that provides copied request at 8176 addressable buffer size</td>
</tr>
<tr>
<td>CRPLBUF</td>
<td>The buffer pool that provides copied request parameter list buffers</td>
</tr>
<tr>
<td>IOBUF</td>
<td>The buffer pool that provides input/output buffers</td>
</tr>
<tr>
<td>LFBUF</td>
<td>The buffer pool that provides large fixed buffers</td>
</tr>
<tr>
<td>LPBUF</td>
<td>The buffer pool that provides large pageable buffers</td>
</tr>
<tr>
<td>SFBUF</td>
<td>The buffer pool that provides small fixed buffers</td>
</tr>
<tr>
<td>SPBUF</td>
<td>The buffer pool that provides small pageable buffers</td>
</tr>
<tr>
<td>TIBUF</td>
<td>The buffer pool that contains control information to support HPDT services</td>
</tr>
<tr>
<td>T1BUF</td>
<td>The buffer pool that consolidates control information to support HPR data transmissions and acts as a small packing buffer for HiperSockets and QDIO</td>
</tr>
<tr>
<td>T2BUF</td>
<td>The buffer pool that consolidates control information to support HPR data transmissions and acts as a large packing buffer for HiperSockets and QDIO</td>
</tr>
<tr>
<td>XDBUF</td>
<td>The buffer pool that provides XID buffers</td>
</tr>
</tbody>
</table>

Performance and tuning start options

Table 61. Start options that affect performance and tuning. These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=PERFTUNE command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTOTI</td>
<td>Specifies how often pending automatic logon requests owned by this host are retried</td>
</tr>
</tbody>
</table>
Table 61. Start options that affect performance and tuning (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=PERFTUNE command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSCTMOUT</td>
<td>The interval in which two general poll failures are allowed for BSC 3270 terminals</td>
</tr>
<tr>
<td>CACHETI</td>
<td>The amount of time that an APPN search request is remembered to prevent processing the same search again</td>
</tr>
<tr>
<td>CDRSCTI</td>
<td>The amount of time that dynamic CDRSCs are saved after the last session has ended</td>
</tr>
<tr>
<td>CINDXSIZ</td>
<td>The maximum size of the ISTCIT and ISTCONVT index tables</td>
</tr>
<tr>
<td>DIRSIZE</td>
<td>The maximum number of dynamic resources that VTAM stores in the APPN directory database</td>
</tr>
<tr>
<td>DIRTIME</td>
<td>The amount of time that a dynamic resource can go unused before it is deleted from the APPN directory database</td>
</tr>
<tr>
<td>DLRTCB</td>
<td>Specifies the maximum number of task control blocks that VTAM allocates in private storage for various functions</td>
</tr>
<tr>
<td>HNTSIZE</td>
<td>The maximum size of the host node table used to find element addresses</td>
</tr>
<tr>
<td>HPRARB</td>
<td>The algorithm that is used for adaptive pacing (not recommended unless you use the default value of RESPMODE)</td>
</tr>
<tr>
<td>HPRCLKRRT</td>
<td>Specifies whether the HPR clock rate runs in standard or adaptive mode. These modes control how reactive VTAM is to the packet loss that occurs on RTP pipes which run over Enterprise Extender.</td>
</tr>
<tr>
<td>HPRITMSG</td>
<td>Specifies whether VTAM should issue the enhanced version or base version of HPR activation and deactivation messages</td>
</tr>
<tr>
<td>HPRPSDLY</td>
<td>Specifies the minimum amount of time that VTAM delays prior to initiating path switch logic for an RTP pipe that has an unresponsive partner</td>
</tr>
<tr>
<td>HPRPSMSG</td>
<td>Specifies whether VTAM should limit the number of HPR path switch messages that are issued</td>
</tr>
<tr>
<td>HPRPST</td>
<td>The maximum time that VTAM tries a path switch before ending a connection</td>
</tr>
<tr>
<td>HPRSESLM</td>
<td>Specifies the maximum number of sessions that should be assigned to a single RTP pipe</td>
</tr>
<tr>
<td>HSRTSIZE</td>
<td>The size of the SRT directory for the network containing the VTAM host node</td>
</tr>
<tr>
<td>IOINT</td>
<td>The interval after which the VTAM operator is notified of outstanding I/O requests</td>
</tr>
<tr>
<td>IOPURGE</td>
<td>The interval after which outstanding I/O requests are assumed to be lost and recovery steps are taken</td>
</tr>
<tr>
<td>MAXEETST</td>
<td>Specifies the maximum number of Enterprise Extender connectivity tests (D NET,EEDIAG,TEST=YES) that are allowed to run concurrently</td>
</tr>
<tr>
<td>MAXHNRES</td>
<td>Specifies the maximum number of host name resolver requests that are allowed to run concurrently for the D NET,EE or D NET,EEDIAG commands</td>
</tr>
</tbody>
</table>
Table 61. Start options that affect performance and tuning (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=PERFTUNE command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAXLOCAT</td>
<td>The maximum threshold value to be reached before VTAM suspends sending APPN search requests to an adjacent node.</td>
</tr>
<tr>
<td>MAXLURU</td>
<td>The maximum RU length for LU-LU sessions.</td>
</tr>
<tr>
<td>MIHTMOUT</td>
<td>The interval in which you can adjust the missing interrupt handling interval for channel-to-channel communication.</td>
</tr>
<tr>
<td>MULTPATH</td>
<td>Specifies whether VTAM supports TCP/IP multipath routing for EE connections.</td>
</tr>
<tr>
<td>MXSAWBUF</td>
<td>The maximum number of buffers to use for session awareness data.</td>
</tr>
<tr>
<td>MXSSCPRU</td>
<td>The maximum RU length for all SSCP sessions.</td>
</tr>
<tr>
<td>NCPBUFSZ</td>
<td>The RU size used for loading or dumping a remote NCP.</td>
</tr>
<tr>
<td>NSRTSIZE</td>
<td>Provides an override of OSRTSIZE for a maximum of nine network identifiers.</td>
</tr>
<tr>
<td>NUMTREES</td>
<td>The maximum number of APPN routing trees that can be saved.</td>
</tr>
<tr>
<td>OSIEVENT</td>
<td>Specifies whether VTAM topology notifications are generated for CMIP services events.</td>
</tr>
<tr>
<td>OSITOPO</td>
<td>Specifies whether the VTAM topology agent reports all types of cross domain resources (CDRSCs) to CMIP manager application programs.</td>
</tr>
<tr>
<td>OSRTSIZE</td>
<td>The size of the SRT directory for networks other than the VTAM host node's network.</td>
</tr>
<tr>
<td>PDTRCBUF</td>
<td>The number of buffers to build for a session monitor request.</td>
</tr>
<tr>
<td>PIUMAXDS</td>
<td>A value used to calculate the maximum buffer limit for PIU data space.</td>
</tr>
<tr>
<td>PMTUD</td>
<td>Specifies whether VTAM uses path MTU discovery for IPv4 and IPv6 Enterprise Extender connections.</td>
</tr>
<tr>
<td>PSRETRY</td>
<td>The interval between attempts to switch a rapid transport protocol (RTP) session to a path that can be equal to or better than the current path, and an option to immediately attempt an RTP path switch when a link on this host is activated or changes status.</td>
</tr>
<tr>
<td>PSWEIGHT</td>
<td>The weight comparison between an old route and a newly calculated route before a path switch is attempted.</td>
</tr>
<tr>
<td>SAWMAXXS</td>
<td>A value used to calculate the maximum buffer limit for SAW data space.</td>
</tr>
<tr>
<td>SAWMXQPK</td>
<td>A value used to calculate the number of SAW buffers to queue before packing the buffers.</td>
</tr>
<tr>
<td>SONLIM</td>
<td>The maximum percentage of I/O buffers that can be used for session outage notification RUs.</td>
</tr>
<tr>
<td>SRCHRED</td>
<td>Indicates whether this node enables eliminating and reducing searches for unreachable resources.</td>
</tr>
<tr>
<td>SRCOUNT</td>
<td>The number of search requests that fail before VTAM performs another resource discovery search.</td>
</tr>
</tbody>
</table>
Table 61. Start options that affect performance and tuning (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=PERFTUNE command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRTIMER</td>
<td>The amount of time VTAM does not conduct searches for an unavailable resource</td>
</tr>
<tr>
<td>UPDDELAY</td>
<td>Specifies the maximum time the VTAM topology agent waits between checking the list of resources in transient states (such as PENDING ACTIVE and PENDING INACTIVE) to determine whether the resources have been in transient states long enough to be sent to the topology manager</td>
</tr>
<tr>
<td>VFYRED</td>
<td>Specifies whether the node should attempt to perform resource verification reduction</td>
</tr>
<tr>
<td>VFYREDTI</td>
<td>Specifies the node-wide maximum time interval that a resource's location is not verified</td>
</tr>
<tr>
<td>VOSDEACT</td>
<td>Can be coded in the VTAM start list, or turned on or off by way of the MODIFY VTAMOPTS command, to cause VTAM to bring a virtual route (VR) down if the VR goes out of sequence</td>
</tr>
<tr>
<td>VTAMEAS</td>
<td>The size of the table that contains information about sessions with an application program</td>
</tr>
</tbody>
</table>

APPN start options

Table 62. Start options that define APPN characteristics. These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=APPNCHAR command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPNCOS</td>
<td>The APPN Class of Service that is used if the requested Class of Service cannot be found</td>
</tr>
<tr>
<td>BN</td>
<td>Whether this node is to provide extended border node function</td>
</tr>
<tr>
<td>BNDYN</td>
<td>Indicates how nodes are added dynamically to adjacent cluster routing lists</td>
</tr>
<tr>
<td>BNORD</td>
<td>Indicates the order VTAM performs cross-subnet searches</td>
</tr>
<tr>
<td>CDSERVR</td>
<td>Whether this node is a central directory server</td>
</tr>
<tr>
<td>CDSREFER</td>
<td>Specifies how many of the nearest central directory servers are used for Locate searches and central resource registration</td>
</tr>
<tr>
<td>CONNTYPE</td>
<td>Whether a LEN connection or an APPN connection is established with type 2.1 PUs</td>
</tr>
<tr>
<td>CPCP</td>
<td>The types of connections on which CP-CP sessions are allowed with adjacent nodes</td>
</tr>
<tr>
<td>DIRSIZE</td>
<td>The maximum number of dynamic resources that VTAM stores in the APPN directory database</td>
</tr>
<tr>
<td>DIRTME</td>
<td>The amount of time that a dynamic resource can go unused before it is deleted from the APPN directory database</td>
</tr>
</tbody>
</table>
Table 62. Start options that define APPN characteristics (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS, FUNCTION=APPNCHAR command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLURSAW</td>
<td>Specifies whether there will be an NLDM application started that needs to receive information of DLUR's pipe path switch. This process is to update the NLDM of RSCV updates when the DLUR's pipe is path switched. The endpoint of the HPR pipe that path switched, is not this VTAM. By default the DLURSAW capability will be enabled in the host. Whether path switch updates are actually received by NLDM also depends on the capability and configuration of the DLUR.</td>
</tr>
<tr>
<td>DUPDEFS</td>
<td>Whether APPN should continue searching for the target resource when the resource has been found but is not active</td>
</tr>
<tr>
<td>DYNADJCP</td>
<td>Whether dynamic entries are created in ISTADJCP for adjacent APPN nodes</td>
</tr>
<tr>
<td>EEHPRANR</td>
<td>Determines whether VTAM maintains diagnostic EE counters for inbound HPR packets that are routed by using the ANR protocol through this network node.</td>
</tr>
<tr>
<td>EEPORTCK</td>
<td>The EEPORTCK start option determines whether VTAM accepts inbound UDP packets that specify an ephemeral source port.</td>
</tr>
<tr>
<td>EEVERIFY</td>
<td>Specifies whether VTAM should automatically send an LDLC probe to the remote partner to determine whether all of the ports are accessible during the activation of the EE connection. This option might also specify the time interval at which VTAM should send the probe on active EE connections.</td>
</tr>
<tr>
<td>GVBKDLY</td>
<td>Determines whether VTAM internally paces the subcommands that are generated from a VARY INACT, TYPE=GIVEBACK command. This start option controls GIVEBACK commands that use wildcard values for the ID operand to deactivate dependent LU requesters (DLURs).</td>
</tr>
<tr>
<td>HOSTNAME</td>
<td>Specifies the default hostname for use in name-to-address resolution for acquiring a static VIPA for use with Enterprise Extender connections</td>
</tr>
<tr>
<td>HPR</td>
<td>Specifies the level of High-Performance Routing (HPR) support provided by VTAM</td>
</tr>
<tr>
<td>HPRARCB</td>
<td>The algorithm that is used for adaptive pacing (not recommended unless you use the default value of RESPMODE)</td>
</tr>
<tr>
<td>HPRCLKRT</td>
<td>Specifies whether the HPR clock rate runs in standard or adaptive mode. These modes control how reactive VTAM is to the packet loss that occurs on RTP pipes that run over Enterprise Extender.</td>
</tr>
<tr>
<td>HPRNCBPBF</td>
<td>Specifies whether HPR can be used although it may cause session data to travel through an NCP twice</td>
</tr>
<tr>
<td>HPRPSDL</td>
<td>Specifies the minimum amount of time that VTAM delays prior to initiating path switch logic for an RTP pipe that has an unresponsive partner</td>
</tr>
<tr>
<td>HPRPST</td>
<td>The maximum time that VTAM tries a path switch before ending a connection</td>
</tr>
<tr>
<td>HPRSESLM</td>
<td>Specifies the maximum number of sessions that should be assigned to a single RTP pipe</td>
</tr>
</tbody>
</table>
Table 62. Start options that define APPN characteristics (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=APPNCHAR command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPRSTALL</td>
<td>The amount of time that an RTP pipe can remain continuously stalled before it is stopped automatically</td>
</tr>
<tr>
<td>INITDB</td>
<td>Whether to load the APPN directory database, topology database, neither, or both when VTAM is started</td>
</tr>
<tr>
<td>IOPURGE</td>
<td>The interval after which outstanding I/O requests are assumed to be lost and recovery steps are taken</td>
</tr>
<tr>
<td>IPADDR</td>
<td>The default IP address of the local TCP/IP stack that is used with EE connections</td>
</tr>
<tr>
<td>MAXLOCAT</td>
<td>The maximum threshold value that can be reached before VTAM suspends sending APPN search requests to an adjacent node</td>
</tr>
<tr>
<td>MULTIPATH</td>
<td>Specifies whether VTAM supports TCP/IP multipath routing for EE connections</td>
</tr>
<tr>
<td>NETID</td>
<td>The name of the network containing the host</td>
</tr>
<tr>
<td>NNSPREF</td>
<td>Specifies the name of the preferred network node server for this end node</td>
</tr>
<tr>
<td>NODETYPE</td>
<td>The APPN node type (network node or end node)</td>
</tr>
<tr>
<td>NUMTREES</td>
<td>The maximum number of APPN routing trees that can be saved</td>
</tr>
<tr>
<td>PMTUD</td>
<td>Specifies whether VTAM uses path MTU discovery for IPv4 and IPv6 Enterprise Extender connections</td>
</tr>
<tr>
<td>PSRETRY</td>
<td>The interval between attempts to switch a rapid transport protocol (RTP) session to a path that can be equal to or better than the current path, and an option to immediately attempt an RTP path switch when a link on this host is activated or changes status</td>
</tr>
<tr>
<td>PSWEIGHT</td>
<td>The weight comparison between an old route and a newly calculated route before a path switch is attempted</td>
</tr>
<tr>
<td>RESUSAGE</td>
<td>The number of times an APPN resource (node or TG) can be used for routing before it is considered overused</td>
</tr>
<tr>
<td>ROUTERES</td>
<td>The relative amount of resistance to having this APPN node perform intermediate session routing</td>
</tr>
<tr>
<td>SACONNS</td>
<td>Specifies if subarea connections will be used</td>
</tr>
<tr>
<td>SAVERSCV</td>
<td>Indicates whether or not Route Selection Control Vectors (RSCVs) for future sessions are to be saved for display purposes following the completion of session activation</td>
</tr>
<tr>
<td>SECLVLCP</td>
<td>Whether VTAM uses basic or enhanced protocol for session-level security verification during activation of LU 6.2 sessions involving APPN control points</td>
</tr>
<tr>
<td>SNVC</td>
<td>Maximum number of networks that can be searched for a resource by this border node</td>
</tr>
<tr>
<td>SORDER</td>
<td>The order in which to search APPN and subarea networks when a network search request is received from a subarea network</td>
</tr>
<tr>
<td>SRCHRED</td>
<td>Indicates whether this node enables eliminating and reducing searches for unreachable resources</td>
</tr>
</tbody>
</table>
### Functions from zappable constants

Table 63. Start options that were zappable constants. These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=ZAPCON command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASIRFMSG</td>
<td>The SSCPs issue messages IST890I and IST896I when an autologon session initiation request fails</td>
</tr>
</tbody>
</table>
Table 63. Start options that were zappable constants (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=ZAPCON command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSCTMOUT</td>
<td>The interval which two general poll failures are allowed for BSC 3270 terminals</td>
</tr>
<tr>
<td>CINDXSIZ</td>
<td>The maximum size of the ISTCIT and ISTCONVT index tables</td>
</tr>
<tr>
<td>ESIRFMSG</td>
<td>The SSCPs issue messages IST891I, IST892I, and IST893I when a session initiation request fails and extended sense data exists</td>
</tr>
<tr>
<td>FSIRFMSG</td>
<td>The SSCPs issue messages IST894I, IST895I, IST1704I, and IST1705I when a session initiation request fails because VTAM is unable to locate the destination LU by using an adjacent SSCP table</td>
</tr>
<tr>
<td>HNTSIZE</td>
<td>The maximum size of the host node table used to find element addresses</td>
</tr>
<tr>
<td>HSRTSIZE</td>
<td>The size of the SRT directory for the network containing the VTAM host node</td>
</tr>
<tr>
<td>INOPDUMP</td>
<td>Whether a resource is eligible to initiate a VTAM dump under certain INOP conditions</td>
</tr>
<tr>
<td>IOMSGLIM</td>
<td>The maximum number of IST530I and IST532I message pairs that are written for individual subareas by the IOPD facility</td>
</tr>
<tr>
<td>IRNSTRGE</td>
<td>The maximum amount of VTAM private storage that can be used for host intermediate routing node (IRN) transmissions</td>
</tr>
<tr>
<td>MAXLURU</td>
<td>The maximum RU length for LU-LU sessions</td>
</tr>
<tr>
<td>MAXSSCPs</td>
<td>The maximum number of SSCPs that are searched on a specific path for the destination SSCP before alternate paths are tried</td>
</tr>
<tr>
<td>MIHTMOUT</td>
<td>The interval which you can adjust the missing interrupt handling interval for channel-to-channel communication</td>
</tr>
<tr>
<td>MSAWBUFS</td>
<td>The maximum number of buffers to use for session awareness data</td>
</tr>
<tr>
<td>MXSSCPRU</td>
<td>The maximum RU length for all SSCP sessions</td>
</tr>
<tr>
<td>MXSUBNUM</td>
<td>The maximum subarea number supported by a network to which a VTAM host is interconnected</td>
</tr>
<tr>
<td>NCPBUFSIZE</td>
<td>The RU size used for loading or dumping a remote NCP</td>
</tr>
<tr>
<td>OSRTSIZE</td>
<td>The size of the SRT directory for networks other than the VTAM host node’s network</td>
</tr>
<tr>
<td>PDTRCBUF</td>
<td>The number of buffers to build for a session monitor request</td>
</tr>
<tr>
<td>PIUMAXDS</td>
<td>A value used to calculate the maximum buffer limit for PIU data space</td>
</tr>
<tr>
<td>PLUALMSG</td>
<td>Whether VTAM is to suppress duplicate session setup failure messages associated with LOGAPPL failures to the same PLU</td>
</tr>
<tr>
<td>PSSTRACE</td>
<td>Whether to record IRB and SRB entries in the VTAM internal trace table when the PSS trace option is in effect</td>
</tr>
<tr>
<td>SAWMAXDS</td>
<td>A value used to calculate the maximum buffer limit for SAW data space</td>
</tr>
<tr>
<td>SAWMXQPK</td>
<td>A value used to calculate the number of SAW buffers to queue before packing the buffers</td>
</tr>
<tr>
<td>SDLCMDRS</td>
<td>Which SDLC miscellaneous data records are to be recorded in LOGREC</td>
</tr>
</tbody>
</table>
Table 63. Start options that were zappable constants (continued). These start options can be displayed as a group with the DISPLAY VTAMOPTS,FUNCTION=ZAPCON command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIRFMSG</td>
<td>The SSCPs in which to issue IST663I, IST664I, IST889I, and subordinate messages when a session initiation request fails.</td>
</tr>
<tr>
<td>SLUALMSG</td>
<td>Whether to suppress messages for session setup failures that occur when the SLU is not enabled.</td>
</tr>
<tr>
<td>SNAPREQ</td>
<td>The number of requests for VTAM buffers between snapshot dumps.</td>
</tr>
<tr>
<td>SSDTMOUT</td>
<td>The amount of time that a switched PU can go without a session before the connection is dropped.</td>
</tr>
<tr>
<td>TRANSLAT</td>
<td>Which names are to be translated by alias name translation.</td>
</tr>
<tr>
<td>VTAMEAS</td>
<td>The size of the table that contains information about sessions with an application program.</td>
</tr>
</tbody>
</table>

HPR and Enterprise Extender start options

Table 64. Start options that define HPR and Enterprise Extender (EE) characteristics. You can display these start options as a group with the DISPLAY VTAMOPTS,FUNCTION=HPREE command.

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYNHPPFX</td>
<td>A 2-character prefix that is used to create PU names for dynamic HPR PUs.</td>
</tr>
<tr>
<td>EEHPRANR</td>
<td>Determines whether VTAM maintains diagnostic EE counters for inbound HPR packets that are routed by using the ANR protocol through this network node.</td>
</tr>
<tr>
<td>EEPORTCK</td>
<td>Determines whether VTAM will accept inbound UDP packets that specify an ephemeral source port.</td>
</tr>
<tr>
<td>EEVERIFY</td>
<td>Specifies whether VTAM should automatically send an LDLC probe to the remote partner to determine whether all of the ports are accessible during the activation of the EE connection. This option might also specify the time interval at which VTAM should send the probe on active EE connections.</td>
</tr>
<tr>
<td>GVBDKDLY</td>
<td>Determines whether VTAM internally paces the subcommands that are generated from a VARY INACT,TYPE=GIVEBACK command. This start option controls GIVEBACK commands that use wildcard values for the ID operand to deactivate dependent LU requesters (DLURs).</td>
</tr>
<tr>
<td>HOSTNAME</td>
<td>The default TCP/IP domain name used in name-to-address resolution to acquire a static VIPA for EE connections.</td>
</tr>
<tr>
<td>HPR</td>
<td>The level of High-Performance Routing (HPR) support provided by VTAM.</td>
</tr>
<tr>
<td>HPRAR     B</td>
<td>The algorithm that is used for adaptive pacing (not recommended unless you use the default value of RESPMODE).</td>
</tr>
<tr>
<td>HPRCLKRT</td>
<td>The mode used for the HPR clock rate that controls how reactive VTAM is to packet loss on RTP pipes that run over EE connections.</td>
</tr>
<tr>
<td>HPRITMSG</td>
<td>Specifies whether VTAM should issue the enhanced version or base version of HPR activation and deactivation messages.</td>
</tr>
</tbody>
</table>
Table 64. Start options that define HPR and Enterprise Extender (EE) characteristics (continued). You can display these start options as a group with the DISPLAY VTAMOPTS,FUNCTION=HPREE command

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPRNCPBF</td>
<td>Specifies whether HPR can be used even though it might cause session data to travel through a network control program (NCP) twice</td>
</tr>
<tr>
<td>HPRPSDLY</td>
<td>The minimum amount of time that VTAM delays prior to initiating path-switch logic for an RTP pipe that has an unresponsive partner</td>
</tr>
<tr>
<td>HPRPSMSG</td>
<td>Specifies whether VTAM should limit the number of HPR path-switch messages that are issued</td>
</tr>
<tr>
<td>HPRPST</td>
<td>The maximum amount of time that VTAM tries a path switch before ending a connection</td>
</tr>
<tr>
<td>HPRSESLM</td>
<td>The maximum number of sessions that should be assigned to a single RTP pipe</td>
</tr>
<tr>
<td>HPRSTALL</td>
<td>The amount of time that an RTP pipe can remain continuously stalled before it is stopped automatically</td>
</tr>
<tr>
<td>IPADDR</td>
<td>The default IP address of the local TCP/IP stack that is used with EE connections</td>
</tr>
<tr>
<td>MAXEETST</td>
<td>The maximum number of EE connectivity tests (D NET,EEDIAG,TEST=YES) that can run concurrently</td>
</tr>
<tr>
<td>MAXHNRES</td>
<td>The maximum number of EE host name resolution requests that can run concurrently</td>
</tr>
<tr>
<td>MULTPATH</td>
<td>Specifies whether VTAM supports TCP/IP multipath routing for EE connections</td>
</tr>
<tr>
<td>PMTUD</td>
<td>Specifies whether VTAM uses path MTU discovery for IPv4 and IPv6 EE connections</td>
</tr>
<tr>
<td>PSRETRY</td>
<td>The interval between attempts to switch a rapid transport protocol (RTP) session to a path that can be equal to or better than the current path, and an option to immediately attempt an RTP path switch when a link on this host is activated or changes status</td>
</tr>
<tr>
<td>PSWEIGHT</td>
<td>The weight comparison between an old route and a newly calculated route, which determines whether a path switch is attempted</td>
</tr>
<tr>
<td>TCPNAME</td>
<td>The job name of the TCP/IP stack through which VTAM communicates with the IP network</td>
</tr>
<tr>
<td>UNRCHTIM</td>
<td>The amount of time that a session path through an EE connection network is considered unusable after certain connection failures</td>
</tr>
</tbody>
</table>

Network management start options

Table 65. Start options that define network management characteristics. You can display these start options as a group with the DISPLAY VTAMOPTS,FUNCTION=NETMGMT command

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNMTAB</td>
<td>The name of the communication network management (CNM) routing table used to route unsolicited request units</td>
</tr>
</tbody>
</table>
### Table 65. Start options that define network management characteristics (continued)

You can display these start options as a group with the DISPLAY VTAMOPTS,FUNCTION=NETMGMT command

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLURSAW</td>
<td>Specifies whether path-switch information about RTP pipes with DLUR endpoints should be sent to a Network Logic Data Manager (NLDM) application</td>
</tr>
<tr>
<td>IPINFO</td>
<td>Provides control over the display and transfer of IP characteristics received by this host</td>
</tr>
<tr>
<td>MXSAWBUF</td>
<td>The maximum number of buffers used for session awareness (SAW) data</td>
</tr>
<tr>
<td>NMVTLOG</td>
<td>Specifies whether network management vector transport (NMVT) alerts are recorded in LOGREC</td>
</tr>
<tr>
<td>OSIEVENT</td>
<td>Specifies whether VTAM topology notifications are generated for CMIP services events</td>
</tr>
<tr>
<td>OSIMGMT</td>
<td>Specifies whether Common Management Information Protocol (CMIP) application programs on VTAM are supported, including the VTAM topology agent</td>
</tr>
<tr>
<td>OSIOTOPO</td>
<td>Specifies whether the VTAM topology agent reports all types of cross-domain resources (CDRSCs) to CMIP manager application programs</td>
</tr>
<tr>
<td>PDTRCBUF</td>
<td>The number of buffers to build for a session monitor request</td>
</tr>
<tr>
<td>SAWMAXDS</td>
<td>A value used to calculate the maximum buffer limit for a SAW data space</td>
</tr>
<tr>
<td>SAWMXQPK</td>
<td>A value used to calculate the number of SAW buffers to queue before packing the buffers</td>
</tr>
<tr>
<td>SNAMGMT</td>
<td>Specifies whether the SNA network management interface (NMI) is supported, including the attachment of the ISTMGCEH subtask, which acts as the socket server for the interface.</td>
</tr>
<tr>
<td>UPDDELAY</td>
<td>The maximum length of time that the VTAM topology agent waits between checks for resources that are in transient states to determine whether updates for a resource should be sent to the topology manager</td>
</tr>
</tbody>
</table>

### Coupling facility and SYSPLEX start options

Table 66. Start options that define coupling facility and SYSPLEX characteristics. You can display these start options as a group with the DISPLAY VTAMOPTS,FUNCTION=SYSPLEX command

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFFDELAY</td>
<td>The amount of time that generic resource affinities for a non-syncpoint, non-Limited Resource LU 6.2 session should be retained after session termination</td>
</tr>
<tr>
<td>STRGR</td>
<td>The name of the generic resources structure</td>
</tr>
<tr>
<td>STRMNPS</td>
<td>The name of the structure for multinode persistent sessions</td>
</tr>
<tr>
<td>XCFGRPID</td>
<td>The group ID suffix that VTAM uses when joining groups in the sysplex</td>
</tr>
</tbody>
</table>
Table 66. Start options that define coupling facility and SYSPLEX characteristics (continued). You can display these start options as a group with the DISPLAY VTAMOPTS,FUN=SYSPLEX command

<table>
<thead>
<tr>
<th>Start option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XCFINIT</td>
<td>Specifies whether VTAM joins a sysplex group at VTAM initialization and establishes the capability for XCF connectivity to other VTAM nodes in the sysplex</td>
</tr>
</tbody>
</table>

Descriptions of start options

The remainder of this topic describes the start options in alphabetical order. Information given about each start option includes the function of the option, the default, if any, and any restrictions on the option's use.

Note: For some start options, there is a table following the syntax diagram which shows a dependency, or a value range, or both. If there is not a table following a syntax diagram, then there is not a dependency or a value range.

AFFDELAY start option

```
AFFDELAY = 600
AFFDELAY = number_of_seconds
```

Range: 0–65535

Specifies the number of seconds the affinity for a non-synchpoint, non-Limited Resource LU 6.2 session should be retained after the session terminates. The allowed range is 0-65535, where 0 means there is to be no delay in the deletion of the affinity after session termination. The default is 600, that is, 10 minutes.

If AFFDELAY is 0 when the session terminates the affinity will be deleted. If the AFFDELAY is nonzero, the affinity is retained and the current TOD clock value is recorded in the SPTE control block. Later, when the SPTE is referenced during a session initiation, the stored TOD clock is compared against the current TOD clock to see if the time limit specified by AFFDELAY has expired. If it has expired, the SPTE will deleted, thus ending the affinity, and the session initiation will proceed as though no affinity existed. If the AFFDELAY time has not expired, the affinity will be considered to be in force and the new session will be assigned to the same Generic Resource member.

You can change the value of AFFDELAY with the MODIFY VTAMOPTS command while VTAM is running. The change takes effect immediately. This means that if AFFDELAY is changed, the new value is used in any subsequent compare of the TOD clock values in the SPTE, and the new value is used to determine whether to retain the affinity at session termination time. But a change in the AFFDELAY value does not go out and delete already retained affinities before they are referenced as part of a subsequent session initiation.

ALSREQ start option
Specifies whether adjacent link station names (or an appropriate matching name) must appear in the default adjacent link station list of the independent LU at a session request.

If adjacent link station names are required, only session requests over a matching adjacent link station in the list are allowed. For a LEN adjacent link station, this means the names must match. For an APPN adjacent link station, it means that another APPN adjacent link station (real or ISTAPNPU) is found in the list. If adjacent link station names are not required, any adjacent link station can receive a session request for this LU. Furthermore, the link station name is added to the list if it is LEN, or ISTAPNPU is added to the list if it is APPN.

You can change the value of ALSREQ with the MODIFY VTAMOPTS command while VTAM is running.

The value you specify on the ALSREQ start option provides the default for the ALSREQ operand on the CDRSC definition statement if you do not specify it.

**ALSREQ=NO**
- Specifies that an adjacent link station name need not match an entry in the default link station list of the independent LU.

**ALSREQ=YES**
- Specifies that an adjacent link station name must match an entry in the default link station list of the independent LU.

**API64R start option**
Note:
1. API64R is only meaningful if the machine is operating in z/Architecture mode.
2. API64R=NO may cause a data move. Use API64R=YES unless API64R=NO is required.

Specifies whether 64-bit backed CSM data space storage can be passed to VTAM and TCP/IP application programs.

Storage that is 64-bit backed is storage that, when fixed, can be assigned to real storage frames on or above the 2 Gigabyte real storage bar. With only a few exceptions, all CSM data space storage acquired by VTAM and TCP/IP when operating in z/Architecture mode is 64-bit backed.

When API64R=YES, application programs that accept CSM data space buffers for a receive operation will likely be given 64-bit backed CSM data space storage. If an application attempts to issue the LRA instruction (Load Real Address) against this storage, and the storage is resident on or above the 2 Gigabyte real storage bar, a special operation exception program interrupt will occur. Note that LRA is used primarily in preparation for I/O. There are no known application programs that perform I/O directly into or out of CSM data space storage.

In the unlikely event that an application program experiences this special operation exception, API64R=NO can be used to provide temporary relief. API64R=NO forces all data in 64-bit backed CSM data space to be copied to 31-bit backed CSM data space before being passed to an application program as part of a receive operation. The application program support team should be notified of the program interruption and once the application program has been modified to handle 64-bit backed storage, API64R=YES should be reinstated.

**API64R=NO**
- Forces all storage being passed to VTAM and TCP/IP application programs to be 31-bit backed.

**APPNCOS start option**
Notes:

1. APPNCOS is meaningful only if the NODETYPE start option is also used.

Specifies the APPN Class of Service to be used if a requested Class of Service cannot be found in the topology and routing services Class of Service database. See "IBM-supplied APPN Classes of Service" on page 1007 for the names of the seven classes of service and "APPN Class of Service definitions" on page 990 for information on implementing a user-defined Class of Service.

If a requested Class of Service cannot be found, the value you specify in APPNCOS is substituted for it, and there is a possibility that the characteristics of the substitute Class of Service are not the ones you intended for the route.

For example, a secure Class of Service might have been intended, but the substitution provided in APPNCOS might offer a Class of Service that uses unsecured links.

You can change the value of APPNCOS with the MODIFY VTAMOPTS command while VTAM is running.

**ASIRFMSG start option**
Controls the display of messages IST890I and IST896I. These messages are issued when an autologon session initiation request fails.

**Requirement:** The ASIRFMSG start option is subordinate to the SIRFMSG start option and SIRFMSG coded on a CDRSC or APPL. Therefore, you must code SIRFMSG as ALLSSCP or OLUSSCP to receive the ASIRFMSG messages.

You can change the value of ASIRFMSG with the MODIFY VTAMOPTS command while VTAM is running.

**ASIRFMSG=ALLSSCP**
Specifies that messages are issued in all SSCP.

**ASIRFMSG=OLUSSCP**
Specifies that messages are issued only in the origin logical unit (OLU) SSCP.

**ASIRFMSG=NONE**
Specifies that no messages are issued in any SSCP.

**ASYDE start option**

Specifies whether to keep or terminate the session between the device and its application when VTAM receives an asynchronous device end from a local non-SNA device. For example, an asynchronous device end is caused when the device is powered off, or, for a printer, when the cover is lifted.

This option applies to all locally attached non-SNA devices except printers. See also “FEATUR2” on page 280 for a description of FEATUR2=PRINTR or NOPRINTR for a LOCAL device.

**ASYDE=KEEP**
Specifies that the session is to remain up. VTAM generates an exception request for the application, and the application determines whether to end the session.

**ASYDE=TERM**
Specifies that VTAM unconditionally terminates the session.

**AUTHLEN start option**
AUTHLEN = YES
Indicates that VTAM will forward the specified transmission priority from one LEN node to another.

AUTHLEN = NO
Indicates that the transmission priority specified by the entry LEN node will not be passed by VTAM; instead VTAM will set the transmission priority field to medium priority.

AUTORTRY start option

AUTORTRY = AUTOCAP
Specifies that automatic logon requests are retried only when an adjacent CDRM or an adjacent CP which supports automatic logon is activated.

AUTORTRY = CDRM
Specifies that automatic logon requests are retried only when an adjacent CDRM is activated.

AUTORTRY = ALL
Specifies that automatic logon requests are retried when any adjacent CP or CDRM is activated.

AUTORTRY = NONE
Specifies that automatic logon requests are not retried when any adjacent node is activated.

AUTOTI start option

Note: For an NN-EN configuration where the controlling PLU resides in the EN, the NNs might not receive the controlling PLU’s registration at the time of the session attempt. To ensure autologon redrives, code the AUTOTI start option.

You can change the value of AUTORTRY with the MODIFY VTAMOPTS command while VTAM is running.
AUTOTI = 0

\[ \text{AUTOTI} = \text{time period} \]

range: 0–7 days (or equivalent seconds, minutes, or hours)

Specifies how often pending automatic logon requests owned by this host are retried. AUTOTI is specified in seconds (S), minutes (M), hours (H), or days (D).

If you specify a number equal to or greater than 1 but less than 60 with an S, or without any suffix, AUTOTI defaults to 60 seconds. A value of 0 specifies that an automatic logon request is not retried during the period that an application (acting as the primary logical unit) is unavailable. When AUTORTRY≠NONE, VTAM reissues the automatic logon request when the application becomes available.

You can change the value of AUTOTI with the MODIFY VTAMOPTS command while VTAM is running.

**BN start option**

\[ \text{BN} = \text{NO} \]

\[ \text{BN} = \text{YES} \]

Notes:
1. BN is meaningful only if the NODETYPE=NN start option is also used.

Specifies whether this node is to provide extended border node function.

**BN=NO**

Specifies that this node does not provide extended border node function.

**BN=YES**

Specifies that this node provides extended border node function.

Note: When end to end path switching is performed, a BN, which is not an endpoint or the NNS for the endpoint, may be asked to compute an HPR route for a piece of the end to end route. To support this function, the BN needs to provide RTP-level HPR support. Ensure the BN and HPR start options are coded accordingly.

**BNDYN start option**
Notes:

1. BNDYN is meaningful only if the BN=YES start option is also used.

Defines how VTAM will add nodes dynamically to the adjacent cluster routing list when searching for a resource whose NETID matches the NETID specified on the NETID statement.

You can change the value of BNDYN with the MODIFY VTAMOPTS command while VTAM is running.

**BNDYN=NONE**

Specifies that no entries will be added to the routing table except those explicitly identified by the NEXTCP definition statements, in the adjacent cluster table.

**Guideline:** In order to search its own network, when BNDYN=NONE is coded and VTAM is the NNS (OLU) or CP (OLU), two adjacent cluster tables are needed (one for alias requests and the other with VTAM's NETID). The adjacent cluster tables are needed in the case that VTAM's directory information does not reflect the actual location of the DLU.

**BNDYN=LIMITED**

Specifies that all active border nodes in the native subnetwork, as well as active border nodes and peripheral network nodes in nonnative subnetworks, attached to this node are automatically added to the routing list if either of the following conditions is met:

- BNs and nonnative network nodes that VTAM learns about whose network ID matches the network ID of the DLU.
- BNs or nonnative network nodes that have previously sent a Locate that carried the network ID of the DLU.

**BNDYN=FULL**

Specifies that all active border nodes in the native subnetwork as well as active border nodes and peripheral network nodes in nonnative subnetworks attached to this node are automatically added to the routing list.

**BNORD start option**
Notes:
1 BNORD is meaningful only if the BN=YES start option is also used.

Specifies the order in which VTAM will perform cross-subnet searches.

You can change the value of BNORD with the MODIFY VTAMOPTS command while VTAM is running.

**BNORD=DEFINED**
- Specifies that searches are done in the order that you define border nodes and nonnative network nodes.

  **Note:** BNORD=DEFINED only applies to defined routing entries. Dynamic entries are reordered in priority order and placed after all defined entries.

**BNORD=PRIORITY**
- Specifies that search preference is given to nodes for which the most recent search was successful and to nodes whose NETID matches the DLU’s NETID.

**BSCMDRS start option**
Specifies whether VTAM records subarea BSC inoperative and statistical miscellaneous data records (MDRs) in LOGREC.

**Note:** If you do not record MDRs, and a problem occurs in a BSC device, problem diagnosis can be difficult because there are no MDRs in LOGREC.

You can change the value of BSCMDRS with the MODIFY VTAMOPTS command while VTAM is running.

**BSCMDRS=STATS**
- Specifies that VTAM will record the BSC statistical MDRs in LOGREC.

**BSCMDRS=NOSTATS**
- Specifies that VTAM will not record BSC statistical MDRs in LOGREC.

**BSCMDRS=INOPS**
- Specifies that VTAM will record the BSC inoperative MDRs in LOGREC.

**BSCMDRS=NOINOPS**
- Specifies that VTAM will not record BSC inoperative MDRs in LOGREC.

**BSCTMOUT start option**
BSCTMOUT = 286
BSCTMOUT = units_of_time

range: 0–2147483647

Specifies an interval during which VTAM tracks the number of general poll failures that occur for a BSC 3270 terminal. If two failures occur within this interval, VTAM treats this as an unrecoverable failure of the terminal and deactivates it.

There are 1.048576 seconds in each units_of_time. If 0 is specified, the terminal is not deactivated as a result of general poll failures.

Buffer Pool start option

Buffer pool values:

Notes:
1. The IOBUF pool (IO00, IO) is the only buffer pool where all seven values can be specified. For all other buffer pools, the xpanlim field is not supported. If you specify the xpanlim field for any buffer pool other than the IOBUF pool (IO00, IO), even if the field is null, you get an IST1072I message.

Allows you to modify the default buffer pool attributes for the various pools (both basic allocation and dynamic allocation values). Buffer pools are used by VTAM for holding data or building control blocks. See "Buffer pool default values" on page 808 for information about the IBM-supplied buffer pool values.

For more information on buffer pools, types of buffer pools, and guidelines for basic buffer allocation and dynamic buffer allocation, see z/OS Communications Server: SNA Network Implementation Guide.

poolname
The name of the fixed-length buffer pool to which these options apply.

poolname can be the full buffer pool name, the buffer pool ID, or the abbreviation (for example, either SFBUF, SF00, or SF, respectively, for specifying a small buffer pool in fixed storage).
See Table 67 for a list of the buffer pool names and a description of each buffer pool's function. A complete description of each buffer pool is located in the z/OS Communications Server: SNA Network Implementation Guide.

### Table 67. VTAM buffer pools

<table>
<thead>
<tr>
<th>Poolname</th>
<th>Pool ID</th>
<th>Abbreviation</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>APBUF</td>
<td>AP00</td>
<td>AP</td>
<td>Used for device related CIBs</td>
</tr>
<tr>
<td>BSBUF</td>
<td>BS00</td>
<td>BS</td>
<td>Provides storage for the boundary type 2.1, type 2, and type 1 peripheral node session control blocks</td>
</tr>
<tr>
<td>CRA4BUF</td>
<td>CRA4</td>
<td>CRA4</td>
<td>Used for error recovery</td>
</tr>
<tr>
<td>CRA8BUF</td>
<td>CRA8</td>
<td>CRA8</td>
<td>Used for error recovery</td>
</tr>
<tr>
<td>CRPLBUF</td>
<td>CRPL</td>
<td>CR</td>
<td>Contains the RPL-copy pool</td>
</tr>
<tr>
<td>IOBUF</td>
<td>IO00</td>
<td>IO</td>
<td>Used for input/output data</td>
</tr>
<tr>
<td>LFBUF</td>
<td>LF00</td>
<td>LF</td>
<td>Provides buffers needed for each active application program and TSO user logged on</td>
</tr>
<tr>
<td>LPBUF</td>
<td>LP00</td>
<td>LP</td>
<td>Used for error recovery</td>
</tr>
<tr>
<td>SFBUF</td>
<td>SF00</td>
<td>SF</td>
<td>Contains application program information and LU control blocks</td>
</tr>
<tr>
<td>SPBUF</td>
<td>SP00</td>
<td>SP</td>
<td>Used for large message (LMPEO) requests</td>
</tr>
<tr>
<td>TIBUF</td>
<td>TI00</td>
<td>TI</td>
<td>Contains control information to support HPDT services for HPR or IP</td>
</tr>
<tr>
<td>T1BUF</td>
<td>T100</td>
<td>T1</td>
<td>Similar to the TIBUF but larger. Used as a packing buffer by HiperSockets accelerator and QDIO. Also used to contain the HPR headers and the media, IP, and UDP headers for an Enterprise Extender connection.</td>
</tr>
<tr>
<td>T2BUF</td>
<td>T200</td>
<td>T2</td>
<td>Similar to the T1BUF but even larger. T2BUFs are used exclusively for HiperSockets accelerator and QDIO when the T1BUF is perceived to be of insufficient size to pack all the headers and data. Also used by HPR for all retransmissions over HiperSockets and QDIO.</td>
</tr>
<tr>
<td>XDBUF</td>
<td>XD00</td>
<td>XD</td>
<td>Provides storage for the I/O buffer during XID contact processing to peripheral nodes</td>
</tr>
</tbody>
</table>

**baseno**

- **range:** 1 - 32767
- **dependencies:** bufsize

Specifies the base number of buffers in the pool. Do not code commas in the value specified.

**Note:**

1. The value for baseno should contain the maximum number of buffers that will fit on each page. If baseno is smaller than the maximum number of buffers per page, VTAM adjusts the value to the maximum value.
2. Storage for buffer pools is allocated by page. After you know the number of buffers per page (for the given bufsize), you can determine the maximum baseno for your environment.

3. Because storage is allocated by page increments, the amount of buffer storage reserved per page might be less than the size of a page. This storage difference must be considered when determining the value for xpanlim.

**bufsize**

<table>
<thead>
<tr>
<th>range:</th>
<th>256-3992 bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependencies:</td>
<td>baseno, IOBUF pool only</td>
</tr>
</tbody>
</table>

Specifies the number of bytes assigned to each buffer in the pool. If you specify a value less than 256, it is changed to 256. If an odd value is specified, it is rounded down to the closest even number.

**Note:**

1. The bufsize value specified must allow individual buffers to begin on a doubleword boundary and not expand over a page boundary. See “Specifying IOBUF buffer values” on page 810 for specifying IO buffer pool bufsize values.

2. The value for bufsize does not include header information added to each buffer by VTAM. See “Specifying IOBUF buffer values” on page 810 for determining the actual size of each buffer.

3. The only buffer pool for which you can change the bufsize is IOBUF. You may code the default values for the other buffer pools. If you specify a value other than the default value, a message will be issued and initialization will continue with VTAM using the default values.

4. Some local SNA controllers, such as 3174, require an even number and will not operate if bufsize is an odd value. To accommodate this requirement, a specification of an odd bufsize value will be rounded down (-1) to the next even value. Refer to the appropriate device description manual for other resource types.

**slowpt**

<table>
<thead>
<tr>
<th>dependencies:</th>
<th>baseno</th>
</tr>
</thead>
</table>

Provides a set of buffers reserved for priority requests (for example, read from a channel-attached device). When the number of available buffers in this buffer pool is equal to or less than this number, buffers are allocated only for priority requests, and normal requests are queued or rejected. This is called slowdown mode.

**F**

Specifies that the storage for this buffer pool is to be page-fixed. F has meaning only for buffer pools that are normally in pageable storage, but it can be coded for buffer pools that are located in fixed storage by default.

**Note:**

1. To use the xpanno and xpanpt operands, but not the F operand, code either double commas or 0 between the slowpt operand and the xpanno operand. For example, coding either:
LPBUF=(baseno,bufsize,slowpt,,xpanno,xpanpt)
or
LPBUF=(baseno,bufsize,slowpt,0,xpanno,xpanpt)
Causes the buffer pool's default attribute to be used.

2. See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) to determine which buffer pools can be pageable.

**xpanno**

dependencies: `bufsize`  
range: 0-32767

Specifies the number of buffers that VTAM acquires when expanding the buffer pool. This value is rounded to the number of buffers that fills the nearest whole page of storage, however the 32767 maximum values cannot be exceeded. For such a case, `xpanno` is rounded down to the largest number less than or equal to 32767 that will completely fill a whole number of pages.

**Note:**
1. If you code 0, dynamic buffering is not performed and the buffer pool never expands. If you do not code `xpanno`, the IBM-supplied defaults are used.
2. The use of an `xpanno` value that is too small for a pool affects run time performance rather than storage utilization. This is because VTAM will be frequently expanding and/or contracting the pool. See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) for dynamic expansion guidelines.

**xpanpt**

range: `(slowpt + 1)-(baseno - minval) - 1`

Specifies the number of buffers that set the expansion point for this buffer pool. When the number of available buffers in the buffer pool falls to a value that is equal to or less than `xpanpt`, VTAM expands the buffer pool by the number of buffers defined by `xpanno`.

**Note:**
1. Specify a value for `xpanpt` so that the buffer pool is not expanded during VTAM initialization.
2. The value of `xpanpt` must be greater than the value of `slowpt`, but less than the difference between `baseno` and `minval`. `minval` is the minimum number of buffers required to open the VTAM ACB and start VTAM. The `minval` values are listed with the IBM-supplied defaults (see “Buffer pool default values” on page 808).

See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) for information and examples of dynamic buffer allocation.

**xpanlim**

dependencies: `IOBUF` pool only, `xpanno`
range: 0-2097152 Kb

Specifies the total size, in Kb, for this buffer pool including the base allocation and expansions.
VTAM transforms the value of xpanlim into the number of buffers represented by that limit.

Note:
1. If you code 0, or if you do not specify an xpanlim, buffer expansion is not limited, provided the value for xpanno is nonzero.
2. If xpanlim is less than the number of bytes in the basic buffer pool allocation, no expansion is performed.
3. The xpanlim value can be modified after VTAM startup by using the MODIFY BFRUSE command. See z/OS Communications Server: SNA Operation for information about this command.

When a successful expansion of the IO buffer pool brings the total size of the pool to 80 percent (or more) of the expansion limit specified, VTAM scans the IO buffer pool for sessions that are using more than 10 percent of the current size of the buffer pool. Any session using more than 10 percent of the pool receives an IST930I message. This message contains the session partner names as well as the percentage of the pool being used by the session. This happens for each expansion in which the size of the pool is 80 percent or greater of the specified expansion limit.

After VTAM has determined that a session is using more than 10 percent of the IO buffer pool, it checks whether the percentage is greater than or equal to the value specified for HOTIOTRM. If so, VTAM ends the session and any other session between the same resource pair to prevent CSA from being exhausted and possibly causing an outage.

Buffer pool default values
Table 68 on page 809 shows the IBM-supplied values for the buffer pools.

Note:
1. The bufsize value specified is not the actual length of a buffer; VTAM adds additional storage to each buffer for buffer management. The amount of storage added is the same for all buffers except for the IOBUF buffer pool.
   - To estimate storage needs for all buffer pools except IOBUF:
     a. Add 16 bytes to the buffer size.
     b. Round bufsize up to a doubleword (multiple of 8).
   - For IOBUF (no encryption specified):
     a. Add 91 bytes to the bufsize value.
     b. Round bufsize up to a doubleword (multiple of 8).
   - For IOBUF, if YES, 24, or 31 is specified on the ENCRYPTN start option:
     a. Add 98 bytes to the bufsize value.
     b. Round bufsize up to a doubleword (multiple of 8).
2. minval is the minimum number of buffers required to open the VTAM ACB and start VTAM. The minval value is a constant. You cannot change this value.
3. The minimum value for IOBUF bufsize that will be used by VTAM is 256. If a value less than 256 is specified, then it will be over-ridden to 256 during VTAM initialization.

D NET,BFRUSE,BUFFER=IO
IST9201 1000  BUFFSIZE 331

or
4. The default base values for the T1BUF and T2BUF pools are low because these buffers are large. The T1BUF and T2BUF buffers are used only with Enterprise Extender connections that use OSA QDIO or HiperSockets as the underlying data link control (DLC). If these buffers are not used in these cases, the buffer pool allocations waste storage if they are defined to be too large. If EE is being used, you typically set the pool allocations for the T1BUF and T2BUF buffers to be greater than their default values. Use IBM Health Checker for z/OS to check whether the T1BUF or T2BUF base values are greater than the default values when Enterprise Extender is not being used. You can also use IBM Health Checker for z/OS to check whether the base values are at their default values when Enterprise Extender is being used. For more details about IBM Health Checker for z/OS, see [z/OS Communications Server: IP Diagnosis Guide](https://www.ibm.com/support/knowledgecenter/en/SSEXJ1_7.4.0/com.ibm.zos.v2r11.snm.doc/vsam_diagnosis/zos_communication_server_ip_diagnosis_guide.html).

### Table 68: IBM-supplied VTAM buffer pool values

<table>
<thead>
<tr>
<th>Buffer pool name</th>
<th>baseno</th>
<th>bufsize</th>
<th>slowpt</th>
<th>xpanno</th>
<th>xpanpt</th>
<th>xpanlim</th>
<th>minval</th>
</tr>
</thead>
<tbody>
<tr>
<td>APBUF</td>
<td>56</td>
<td>56</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BSBUF</td>
<td>28</td>
<td>260</td>
<td>0</td>
<td>1</td>
<td>14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRA4BUF</td>
<td>50</td>
<td>4080</td>
<td>0</td>
<td>10</td>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRA8BUF</td>
<td>12</td>
<td>8176</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRPLBUF</td>
<td>75</td>
<td>144</td>
<td>0</td>
<td>60</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IOBUF</td>
<td>55</td>
<td>256</td>
<td>5</td>
<td>1</td>
<td>6</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>LFBUF</td>
<td>30</td>
<td>120</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LPBUF</td>
<td>12</td>
<td>2032</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>SFBUF</td>
<td>64</td>
<td>112</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>SPBUF</td>
<td>21</td>
<td>176</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIBUF</td>
<td>360</td>
<td>632</td>
<td>0</td>
<td>60</td>
<td>120</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T1BUF</td>
<td>16</td>
<td>1004</td>
<td>0</td>
<td>32</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T2BUF</td>
<td>8</td>
<td>2028</td>
<td>0</td>
<td>32</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XDBUF</td>
<td>5</td>
<td>697</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**This value can change. The IO buffer can be different sizes. VTAM obtains the base number of buffers that actually fit on a 4K page.**

### Overriding buffer pool values

All buffer pool values that you specify override the previously specified values, while buffer pools that you do not code default to the IBM-supplied values.

In the following example, LPBUF values are specified in the default start option list. When VTAM is started, the system operator selects start option list ST, which contains new LPBUF values which will override the previous values. VTAM processes these values in the order shown below. For more details about processing, see [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/en/SSEXJ1_7.4.0/com.ibm.zos.v2r11.snm.doc/vsam_diagnosis/zos_communication_server_sna_network_implementation_guide.html).

**Overriding buffer pool example:**
In this example, the operator enters LIST=ST. Note that the double commas denote values that are not coded.

IBM default \[\text{LPBUF}= (9,1344,0,,1,2)\]
ATCSTROO \[\text{LPBUF}= (128,,1,,0,0)\]
ATCSTRST (LIST=ST) \[\text{LPBUF}= (8,,0,F,,)\]

The resultant values for LPBUF are: \((8,1344,0,F,0,0)\).

**Specifying IOBUF buffer values**

If you want to maximize real storage usage, choose a buffer size \((\text{bufsize})\) that fits efficiently onto a page. The following calculations can be made to determine an optimum value for your environment. When selecting a value, you need to know the average PIU size to have a starting point for \(\text{bufsize}\).

**Note:** For type 2 local SNA devices, the actual data size is 16 bytes less than BUFSIZE.

To calculate how many buffers fit on a page (no encryption specified):

1. Determine the \(\text{bufsize}\) value.
2. Add 91 bytes to the specified \(\text{bufsize}\) value.
3. Round the sum up to a doubleword boundary.
4. Divide 4096 by the value.
5. The quotient indicates the number of buffers that can fit on 1 page (for the given \(\text{bufsize}\) value).
   The remainder indicates the number of bytes that are not used on each page. However, these unused bytes are taken into consideration for the \(\text{xpanlim}\) value that you specify. Keep this number as low as possible.

If the encryption start option is specified (ENCRYPTN = YES, 24, or 31), add 98 to the \(\text{bufsize}\) value instead of 91.

Table 69 reflects the IOBUF formula and can be used when coding \(\text{bufsize}\). For greatest efficiency, choose the high number in the range, because the higher number has a smaller remainder.

**Note:** The minimum \(\text{bufsize}\) used by VTAM is 256. If a \(\text{bufsize}\) of less than 256 is specified, VTAM overrides the value with a value of 256 and displays no message indicating this has occurred. D NET,BFRUSE,BUFFER=IO will, however, show the resulting change.

**Table 69. IOBUF buffer size and number of buffers per page**

<table>
<thead>
<tr>
<th>Number of buffers per page</th>
<th>Range of buffer sizes for MVS (with 4k paging)</th>
<th>Range of buffer sizes for MVS using data encryption facility (with 4k paging)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1958–3992</td>
<td>1951–3992</td>
</tr>
<tr>
<td>2</td>
<td>1270–1957</td>
<td>1263–1950</td>
</tr>
<tr>
<td>3</td>
<td>934–1269</td>
<td>927–1262</td>
</tr>
<tr>
<td>4</td>
<td>726–933</td>
<td>719–926</td>
</tr>
<tr>
<td>5</td>
<td>590–725</td>
<td>583–718</td>
</tr>
</tbody>
</table>
Table 69. IOBUF buffer size and number of buffers per page (continued)

<table>
<thead>
<tr>
<th>Number of buffers per page</th>
<th>Range of buffer sizes for MVS (with 4k paging)</th>
<th>Range of buffer sizes for MVS using data encryption facility (with 4k paging)</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>494–589</td>
<td>487–582</td>
</tr>
<tr>
<td>7</td>
<td>422–493</td>
<td>415–486</td>
</tr>
<tr>
<td>8</td>
<td>358–421</td>
<td>351–414</td>
</tr>
<tr>
<td>9</td>
<td>318–357</td>
<td>311–350</td>
</tr>
<tr>
<td>10</td>
<td>278–317</td>
<td>270–310</td>
</tr>
<tr>
<td>11</td>
<td>256–277</td>
<td>256–270</td>
</tr>
</tbody>
</table>

**CACHETI start option**

CACHETI = number_of_minutes

range: 0–1440 minutes

Specifies the minimum number of minutes that procedure-correlation identifiers (PCIDs) are cached after VTAM processes a network search request. Cached PCIDs are used to recognize previously processed search requests, thereby preventing duplicate processing.

Note that if you set the CACHETI value too low, duplicate searching can result. Conversely, if you set the value too high, storage use can increase because of VTAM retaining the control blocks used to save the PCIDs.

**CDRDYN start option**
Specifies whether the host is authorized to dynamically define CDRSC representations of cross-domain, cross-network, or APPN resources when a session request is received from or sent to an adjacent SSCP or CP.

You can change the value of CDRDYN with the MODIFY VTAMOPTS command while VTAM is running.

**Note:**
1. For information specific to cross-domain sessions, see "CDRDYN" on page 154.
2. The CDRDYN start option applies only to the host CDRM. The CDRDYN start option value, or a MODIFY VTAMOPTS value, overrides only the CDRDYN value on the hosts’ CDRM definition statement.
3. Using the CDRDYN start option instead of the CDRDYN operand allows greater flexibility with the DYNLU start option.

**CDRDYN=NO**
   Causes the failure of any cross-domain, cross-network, or APPN session request for a cross-domain, cross-network, or APPN resource that has no predefined CDRSC entry.

**CDRDYN=YES**
   Authorizes dynamic definition of cross-domain, cross-network, or APPN resources by this host, eliminating the need to predefine certain CDRSCs. Both origin logical units and destination logical units can be dynamically defined.

### CDRSCTI start option
dependencies: applies only to dynamic CDRSCs
range: 1 second–7 days

Specifies the minimum amount of time for which the system retains a dynamically defined cross-domain resource after the last session with the resource has been terminated.

There are several considerations when specifying a value for CDRSCTI for an interchange node. See z/OS Communications Server: SNA Network Implementation Guide for details.

For a description of the retention time for APPN resources, see “DIRTIME start option” on page 829.

timeout_value can be specified in the following notation, where \( n \) is an integer:

- `n` or `nS` Specifies the number of seconds a resource is retained
- `nM` Specifies the number of minutes a resource is retained
- `nH` Specifies the number of hours a resource is retained
- `nD` Specifies the number of days a resource is retained

**CDSERVR start option**
CDSERVR start option

CDSREFER = NO
    Specifies that this network node is not a central directory server.

CDSREFER = YES
    Specifies that this network node is a central directory server.

CDSREFER start option

Notes:
1  CDSREFER is meaningful only if the NODETYPE=NN and CDSERVR=NO start options are also used.

range: 0–65535

Specifies how many of the nearest central directory servers (CDS) are used for Locate searches and central resource registration. “Nearest” refers to the least-weight CP-CP session paths to a CDS. If CDSREFER=0, then all CDSs in the network are used.

You can change the value of CDSREFER with the MODIFY VTAMOPTS command while VTAM is running.

CINDXSIZ start option
CINDXSIZ = 8176

range: 28–16777199 bytes

Specifies the maximum size of the ISTCIT and ISTCONVT index tables.

**CMPMIPS start option**

CMPMIPS = 100

(1)

Notes:
1. The CMPMIPS start option is meaningful only if the value for CMPVTAM is greater than 1.

range: 0–100

Used by VTAM to determine the amount of time the adaptive compression tables are in adaptive mode versus being static:

- When an adaptive data compression table is in adaptive mode, CMPMIPS is used to determine the amount of time the compression table remains in adaptive mode before becoming static. When a compression table is static, no new entries are made to the table.
- When an adaptive data compression table is static, CMPMIPS is used to calculate the lowest compression efficiency rate that a session must maintain to keep the compression tables static.

The higher the value specified, the greater the amount of time spent in adaptive mode. The lower the value, the sooner a compression table becomes static.

You can change the value of CMPMIPS with the MODIFY VTAMOPTS command while VTAM is running.

Note:
1. Adaptive mode uses CPU cycles; a static table does not use CPU cycles. However, when a table is static, compression efficiency decreases.
2. If CMPMIPS=0, no compression processing is performed.
3. If CMPMIPS=100, adaptive mode compression is continuous and the compression table is never static.
4. CMPMIPS=50 provides the most effective beginning balance between compression efficiency and CPU usage.

**CMPVTAM start option**
Specifies the maximum compression level allowed for sessions involving the host's application programs. Specify one of the following values for compression_level:

0  No compression
1  Run-length encoding (RLE)
2  Small adaptive compression table
3  Medium adaptive compression table
4  Large adaptive compression table

You can change the value of CMPVTAM with the MODIFY VTAMOPTS command while VTAM is running.

**CNMTAB start option**

Specifies the name of the communication network management (CNM) routing table to be used by this VTAM to route unsolicited request units for further processing. The value specified for cnm_table_name is the 1–8 character name of a user-defined CNM routing table.

For more information about coding a user-defined CNM routing table, see z/OS Communications Server: SNA Customization.

**CNNRTMSG start option**
Specifies whether VTAM is to suppress messages when an optimal CNN route exists and is not chosen during session activation. A CNN route that is not optimal is one in which the CNN entry TG subarea and the CNN exit TG subarea chosen do not match and another active TG is found to the adjacent network node that has the same TG characteristics as the CNN entry TG chosen and the same subarea as the CNN exit TG subarea.

You can change the value of CNNRTMSG with the MODIFY VTAMOPTS command while VTAM is running.

**CNNRTMSG=SUPPRESS**
- Specifies that VTAM suppress CNN routing messages. This value can be abbreviated as SUPP.

**CNNRTMSG=NOSUPP**
- Specifies that VTAM does not suppress CNN routing messages. This value can be abbreviated as NOSUP.

### COLD and WARM start option

Defines the status to which the VTAM configuration restart facility restores each major node identified by the CONFIG start option when VTAM was last executing.

**COLD**
- Instructs VTAM to restore each major node to its initial status as defined by the user. VTAM issues the equivalent of:

  ```
  VARY NET,ACT,ID=major_node_name,SCOPE=U
  ```

  for each major node identified by the CONFIG start option.

**WARM**
- Instructs VTAM to restore the major node to its status prior to VTAM’s last termination. VTAM issues the equivalent of:

  ```
  VARY NET,ACT,ID=major_node_name,WARM
  ```

  for each major node identified by the CONFIG start option.
Contains a list of definition members in the VTAM definition library that are to be activated when VTAM is started. For more information on configuration lists, see z/OS Communications Server: SNA Network Implementation Guide.

**CONFIG=00**

Files a user-defined configuration list under the name ATCCON00.

VTAM always uses this predefined list unless a system operator uses the CONFIG option to select another list, or another list of start options specifies a different CONFIG start option. If a default configuration list does not exist when VTAM is started, an error message is sent to the system or domain operator. VTAM initialization continues without the configuration list.

**Note:** The configuration list named by CONFIG replaces the default configuration list. This differs from start option list processing where LIST, if coded, is merged with the 00 list.

**CONFIG=file_id**

Provides 1–2 alphanumeric characters that specify the file (or member) in the VTAM definition library that contains a list of major nodes that are activated when VTAM is started.

File the list of major nodes under the name ATCCONxx.

**CONFIG=file_name**

The 3–8 character file name of the configuration restart VSAM file. The file name must have been coded in the NODELST option during VTAM’s execution prior to the failure or termination of VTAM.

This file contains a list of the major nodes that were active and a list of the dynamic reconfiguration data set (DRDS) files and sets of PATH statements that were in effect at the time of failure or termination of VTAM.

Include a DD statement using this data definition name in the VTAM start procedure.

**CONNTYPE start option**
Notes:

1. CONNTYPE is meaningful only if the NODETYPE start option is also used.

Specifies for an APPN PU (type 2.1) whether a connection from it is established as a LEN connection or attempted as an APPN connection. CONNTYPE applies to end nodes and network nodes.

You can change the value of CONNTYPE with the MODIFY VTAMOPTS command while VTAM is running.

**CONNTYPE=APPN**

Specifies that the connection is attempted as an APPN connection. When you code CONNTYPE=APPN, XID rules for parallel TGs are enforced. So, if you have PUs with duplicate CPNAMEs, you should make them unique before this node becomes an APPN node.

**CONNTYPE=LEN**

Specifies that the connection is established as a LEN connection. CONNTYPE=LEN specifies that this connection does not support CP-CP sessions and CP name change, and that VTAM does not check for duplicate CPNAMEs.

**CPCDRSC start option**
Specifies whether VTAM allows applications or LUs to initiate and establish sessions to a dynamic LEN CP ILU. A dynamic LEN CP ILU is an ILU with a resource name that is the same as its adjacent link station CPNAME.

**CPCDRSC=NO**

Specifies that VTAM does not allow applications or LUs to initiate and establish sessions with dynamic LEN CP ILUs. However, the LEN CP ILU can initiate and establish the session successfully.

**CPCDRSC=YES**

Specifies that VTAM allows applications and LUs to initiate and establish sessions with a dynamic LEN CP ILU.

**CPCP start option**

Specifies whether an APPN node supports CP-CP sessions with an adjacent node.

You can change the value of CPCP with the MODIFY VTAMOPTS command while VTAM is running.

**Rule:** When the CPCP start option is changed with the MODIFY VTAMOPTS command, model major nodes need to be cycled to pick up the new value.

**CPCP=**

- **YES** Specifies that CP-CP sessions on all connections are supported.
- **NO** Specifies that CP-CP sessions are not supported for any connection, leased or switched.
- **LEASED** Specifies that CP-CP connections are supported for leased connections, but not supported for switched connections.
- **SWITCHED** Specifies that CP-CP sessions are supported for switched connections, but not supported for leased connections.

**CSALIMIT start option**
CSALIMIT = 0

Specifies that the maximum amount of CSA storage to be used by VTAM is 90% of the system CSA limit. See z/OS Communications Server: SNA Operation, DISPLAY BFRUSE, to find out what the system CSA limit is for your system.
CSALIMIT=value
Specifies the maximum amount of CSA storage to be used by VTAM. value can be specified in any of the following forms:

nK
n is the number of 1024-byte increments that is the maximum amount of CSA that can be used by VTAM. The value of n is rounded up to the next multiple of 4.

qM
q is the number of 1-megabyte increments that is the maximum amount of CSA that can be used by VTAM.

CSALIMIT=(value,F)
Forces the specified value, n, nK, or qM, to be used as the absolute CSA usage limit. If VTAM's CSA usage reaches this CSALIMIT value, all subsequent CSA requests by VTAM components are rejected until the current CSA value falls below the CSALIMIT value. If sufficient LBUF storage is available, a modify CSALIMIT command can be used to increase the CSALIMIT value.

If the ,F is not specified, VTAM may use CSA until only 25% of system CSA storage remains available for use. If the ,F is specified, VTAM will not use more CSA than the CSALIMIT specified.

CSA24 start option

CSA24=0
Specifies that no limit is enforced on the amount of 24-bit addressable CSA used by VTAM.

CSA24=value
Specifies the maximum amount of CSA storage to be used by VTAM. value can be specified in any of the following forms:

nK
n is the number of 1024-byte increments that is the maximum amount of 24-bit addressable CSA that can be used by VTAM. The value of n is rounded up to the next multiple of 4.
\( q \)

\( q \) is the number of 1 MB increments that is the maximum amount of 24-bit addressable CSA that can be used by VTAM.

**CSDUMP start option**

![Diagram of CSDUMP start option](image)

**Notes:**
1. Specify the CSDUMP start option twice to set both message and sense code triggers.

**CSDUMP sense code trigger:**

![Diagram of CSDUMP sense code trigger](image)

**Rules:**
1. When an error message is received on any parameter of the CSDUMP start option, the remaining parameters for this CSDUMP start option are ignored. You are required to re-enter the complete CSDUMP start option.
2. When the same parameter is entered multiple times on a CSDUMP sense code trigger, only the last occurrence is accepted.
CSDUMP message trigger:

```
<table>
<thead>
<tr>
<th>MESSAGE=</th>
<th>message_id_number</th>
</tr>
</thead>
<tbody>
<tr>
<td>IST1504I</td>
<td>REMOTE=NO</td>
</tr>
<tr>
<td>IST1504I</td>
<td>message_value_field</td>
</tr>
<tr>
<td>(message_id_number, message_value_field)</td>
<td></td>
</tr>
</tbody>
</table>
```

Notes:
1. MSGVALUE is valid only when the MESSAGE operand is used, and MESSAGE specifies either message IST2391I or IST2406I.

Rules:
- When an error message is received on any parameter of the CSDUMP start option, the remaining parameters for this CSDUMP start option are ignored. You are required to re-enter the complete CSDUMP start option.
- When the same parameter is entered multiple times on a CSDUMP message trigger, only the last occurrence is accepted.

You can use the CSDUMP start option to set up a trigger that invokes a dump of the current address space when a particular sense code or message is issued. You can set only one sense code and one message trigger simultaneously; you cannot set two different message or sense code triggers at the same time. You can change a message trigger or a sense code trigger using the MODIFY CSDUMP command.

Either the MESSAGE operand or the SENSE operand must be specified after the CSDUMP start option or the CSDUMP start option is invalid.

**MATCHLIM=matchlim_value**
- Specifies that the CSDUMP trigger is to be disabled after the matchlim_value value matches the value specified on the message trigger. The matchlim_value value is an integer in the range 1-255. The default is 1.

**MESSAGE=message_id_number**
- Specifies the ID number of the message that triggers a dump. Message numbers must be in the format ISTxxxI, ISTxxxxI, ISTxxxE, ISTxxxxE, or IVTxxxxI.

**Rule:** If the start option MSGLEVEL=BASE is specified, or if MSGLEVEL=BASE is specified in a USS operator message table, then the message displayed on the console is the pre-version 4 VTAM message. If you specify CSDUMP with a message trigger, the message ID number is checked before it is swapped to the base message_id_number value. Therefore, the message_id_number value used with the CSDUMP command must be the version 4 VTAM message number for CSDUMP to match the message_id_number value and take the dump. See
**MESSAGE**=(*message_id_number*,*message_value_field*,...)

Specifies the message variable text that can be used to trigger a dump. Instead of just matching a message number, this field causes the trigger to be more specific. If variable text is specified, then a dump is taken only when the message and variable text match. If a variable text field is blank, then it is considered to be a wildcard. See [z/OS Communications Server: SNA Messages](#) for more information about message text for VTAM operator messages.

**Rules:**

1. Use an underscore as a substitute for a space in the message text fields that contain variable values. You do not need to pad the end of the variable to fill out the text field; however, in a few rare cases messages can contain text fields (mostly numbers) in which the variable is right justified and does not completely fill the variable length. In these cases, when you are specifying the CSDUMP operand, the *message_value_field* value must be padded to the left with underscores. For example, the first variable in the IST1461I message is a 3-byte field which is right justified. If a 2-byte number is displayed, an underscore must precede the number in the command in order for the dump command to function correctly. See the following example:

   ```
   CSDUMP,MESSAGE=(IST1461I,_'21)
   ```

2. Each message has a fixed number of *message_value_field* values. Each *message_value_field* value has a maximum length.

3. The specified number of *message_value_field* values must be less than or equal to the number of *message_value_field* values in the specified message.

4. The specified length of the *message_value_field* value must be less than or equal to the maximum length of the specified *message_value_field* value.

Leading *message_value_field* values can be skipped using a comma (,) for each *message_value_field* value. The trailing *message_value_field* values are not required.

See [z/OS Communications Server: SNA Messages](#) to determine the number of message value fields and their maximum lengths.

**REMOTE**

Specifies whether to request a dump of the remote VTAM when an XCF link connecting the two VTAMs becomes inoperative. You can specify this operand only when you specify MESSAGE=IST1504I.

**REMOTE=NO**

A dump of the remote VTAM is not requested when an XCF link becomes inoperative. This is the default behavior.

**REMOTE=YES**

A dump of the remote VTAM is requested when an XCF link becomes inoperative. The remote VTAM must be z/OS V1R9 or later for the dump to be taken.

**RNICTRLE**

Specifies that a diagnostic dump of an IBM 10GbE RoCE Express feature needs to be taken under certain conditions. The RNICTRLE operand can be used only with the MESSAGE trigger.

**RNICTRLE=MSGVALUE**

*MSGVALUE* is valid only when the MESSAGE operand is used, and
MESSAGE specifies either message IST2391I or IST2406I. Specifying the
MSGVALUE keyword allows VTAM to collect diagnostic dump
information for the 10GbE RoCE Express feature identified in these
messages.

RNICTRLE=RNIC_TRLEName
The format of RNIC_TRLEName must be IUTyyyy, where yyyy is the
Peripheral Component Interconnect Express (PCIe) function ID (PFID) that
identifies the 10GbE RoCE Express feature, and y is the port number used
on the 10GbE RoCE Express interface. The value of y can be 1 or 2.

Usage
The 10GbE RoCE Express diagnostic dump is taken in addition to any
other dumps that CSDUMP produces. After the 10GbE RoCE Express
diagnostic dump is produced, recovery of the 10GbE RoCE Express feature
is attempted.

Notes: No 10GbE RoCE Express diagnostic dump is taken in either of the
following cases:
• The TRLE is not active when CSDUMP produces the dump.
• A specific TRLE value is coded for RNICTRLE but the TRLE is not an
  RDMA over Converged Ethernet (RoCE) TRLE.

Rules:
• When the 10GbE RoCE Express feature operates in a dedicated RoCE
  environment, the diagnostic dump deactivates the 10GbE RoCE Express
  feature, and causes an inoperative condition for all users.
• When the 10GbE RoCE Express feature operates in a shared RoCE
  environment, the diagnostic dump only affects the TCP/IP stack that
  configured the PFID value included in the value of RNIC_TRLEName. Other
  TCP/IP stacks that use the RoCE Express feature are not affected.

Guideline: Ensure that multiple 10GbE RoCE Express interfaces are active
with the same physical network ID to avoid loss of connections during a
CSDUMP operation. For more information, see High availability considerations
in z/OS Communications Server: IP Configuration Guide.

RU=ru_code
Specifies the response unit code that contains the sense code specified. The RU
operand can be used only with the SENSE operand. If the RU operand is
specified, then the dump is triggered only if the RU and SENSE codes occur
together. The RU code must be 2, 4, or 6 characters in length.
See z/OS Communications Server: SNA Data Areas Volume 1 for valid RU
codes or see SNA Formats.

SENSE=sense_code
Specifies the sense code that will trigger a dump. This value must be 8
characters in length.
See z/OS Communications Server: IP and SNA Codes for valid sense codes.

TCPNM=TCP/IP_Jobname
Specifies that a dump of the TCPIP job should be taken when the
corresponding sense or message trigger occurs. The current address space is
also specified. The TCP/IP_Jobname value must be 1-8 characters in length.

Examples:
• Set the message trigger to take the dump:
  CSDUMP,MESSAGE=IST1386I
• Set the message trigger to take the dump of the current address space and the
dump of the TCPIP job:
  CSDUMP,MESSAGE=IST1386I,TCPNM=TCPCS
• Set the message trigger with the message and first message_value_field value to
take the dump:
  CSDUMP,MESSAGE=IST169I,react
• Set the message trigger with the message and first two message_value_field values
to take the dump:
  CSDUMP,MESSAGE=IST169I,react,pua
• Set the message trigger with the message and first and fifth message_value_field
values to take the dump:
  CSDUMP,MESSAGE=IST252I,rct,,,,pua
• Set the sense code trigger to take a dump of the current address space:
  CSDUMP,SENSE=08090000
• Set the message trigger and the sense trigger to take a dump:
  CSDUMP,MESSAGE=IST169I,react,pua,
  CSDUMP,SENSE=08090000
• Set the sense code to take a dump of the current address space and a dump of
the TCPIP job:
  CSDUMP,SENSE=08090000,RU=818641,TCPNM=TCPCS
• Set the message trigger with the message and remote option to take a dump of
the current address space and the remote VTAM connected through an XCF link
to another VTAM:
  CSDUMP,MESSAGE=IST1504I,REMOTE=YES

**DATEFORM start option**
Specifies the format of the start date for an automatic scheduled IPL.

**Note:** DATEFORM or DATEFRM is allowed.

If the year is omitted, the current year is used. The MODIFY LOAD command is rejected if the value entered for the date is any of the following conditions:

- Not valid
- Greater than 90 days from the current date or time
- Prior to current date or time

**Note:** You will encounter a MOSS clock problem if you set the IPL to occur after 15 days. For MOSS clock accuracy, the IPL should be less than or equal to 15 days. Otherwise, you can set the IPL up to 90 days in advance.

**DATEFORM=DMY**

Represents the date format DD/MM/YY. The valid delimiters are a slash (/) or a period (.).

**DATEFORM=MDY**

Represents the date format MM/DD/YY. The valid delimiters are a slash (/) or a dash (-).

**DATEFORM=YMD**

Represents the date format YY/MM/DD. The valid delimiters are a slash (/), a period (.), or dash (-).

**DIALRTRY start option**
Specifies whether VTAM should continue searching for a switched resource after a
dial–out failure. Searching should be allowed to continue if the switched resource
is also defined (an possibly already active) on another VTAM. Searching should
not be allowed to continue if the switched resource is defined only on this VTAM.
You can change the value of DIALRTRY with the MODIFY VTAMOPTS command
while VTAM is running.

**DIALRTRY=NO**
Specifies that VTAM should not continue searching for a switched resource
after a dial–out failure.

**DIALRTRY=YES**
Specifies that VTAM should continue searching for a switched resource after a
dial–out failure.

**DIRSIZE start option**

![Diagram](image)

Notes:
1. DIRSIZE is meaningful only if the NODETYPE=NN start option is also used.
   range: 0, or 1000–2147483647

   Specifies the maximum number of dynamic APPN resources that VTAM stores in
   the directory services database. Once the number specified is reached, storage from
   the oldest dynamic resources is freed and reused.

   0 specifies that no limit is enforced for the number of dynamic APPN resources.

   You can change the value of DIRSIZE with the MODIFY VTAMOPTS command
   while VTAM is running.

**DIRTIME start option**
DIRTIME start option

Notes:
1. DIRTIME is meaningful only if the NODETYPE=NN start option is also used.

- Dependencies: Applies only to APPN resources
- Range: 25 hours–2 years (730D)

Specifies the amount of time that a dynamic APPN resource can remain unused in the directory services database before VTAM deletes it. Dynamic resources older than the time specified are deleted.

You can specify the time in seconds (S), hours (H), or days (D).

DIRTIME is valid only when NODETYPE=NN, and applies only to dynamic APPN resources. See "CDRSCTI start option" on page 812 for a description of the retention time for subarea resources.

You can change the value of DIRTIME with the MODIFY VTAMOPTS command while VTAM is running.

DISCNTIM start option
DISCNTIM = (15,0)

- DISCNTIM = (time_period1, time_period2)

<table>
<thead>
<tr>
<th>time_period1:</th>
<th>5–65535 seconds or IMMED</th>
</tr>
</thead>
<tbody>
<tr>
<td>time_period2:</td>
<td>0–255 seconds</td>
</tr>
</tbody>
</table>

Specifies the amount of time VTAM delays deactivation of an SSCP-PU session when there are no LU-LU session requests.

**For Subarea and Non-HPR APPN Sessions:** DISCNTIM is valid only for PU types 2 and 2.1 that have DISCNT=DELAY specified on the PU definition statement.

*time_period1* specifies the amount of time VTAM delays deactivation of an SSCP-PU session after all LU-LU sessions terminate. IMMED specifies the SSCP-PU session will be terminated immediately when the last LU-LU session terminates.

*time_period2* specifies the amount of time after establishment of an SSCP-PU session, VTAM delays deactivation of the session if no LU-LU sessions have been established. When 0 is specified, VTAM will not deactivate the line if no LU-LU session is established after the line is connected.

You can change the value of DISCNTIM with the MODIFY VTAMOPTS command while VTAM is running.

**DLRORDER start option**
Specifies the way VTAM locates a DLUR PU when the connection is initiated by the DLUR. If you do not code DLRORDER, VTAM searches for a DLUR PU by the station identifier (IDBLK and IDNUM operands on the PU definition statement for the switched major nodes) first. If VTAM does not find the PU by the station identifier (STATNID), it searches for the DLUR PU by CPNAME since the second parameter defaults to FIRST.

If the second parameter is defined as ONLY, the search using CPNAME will not be done. If the PU is not found using STATNID, and dynamic PUs are allowed, a dynamic PU will be created and connection processing will continue. If dynamic PUs are not allowed, the PU connection processing will fail.

You can change the value of DLRORDER with the MODIFY VTAMOPTS command while VTAM is running.

Tip: Some DLURs have the capability of sending the PU name to the DLUS. In this case, the DLUS VTAM tries to find the PU by name before looking for the PU by CPNAME or station ID, and the DLRORDER setting is ignored. Refer to the DLUR implementation procedures to see if this function is supported.

**DLRORDER=(value_1,value_2)**

- **value_1=CPNAME**
  Specifies that VTAM searches for a DLUR PU by the CPNAME.

- **value_1=STATNID**
  Specifies that VTAM searches for a DLUR PU by the station identifier (default).

- **value_2=FIRST**
  If a switched PU is not found using the value specified by value_1, the alternate value of value_1 will be used to search for a switched PU.

- **value_2=ONLY**
  If a switched PU is not found using the value specified by value_1, a search using the alternate value of value_1 will not be performed.

**Note:** For regular (non-DLUR) switched PUs, see the SWNORDER start option.

**DLRTCB start option**
DLRTCB = number_of_task_control_blocks

range: 1–32767

Specifies the largest number of task control blocks (TCBs) used by VTAM for the following functions:

- Dump-Load-Restart
- File I/O services
- Cryptographic services
- Configuration checkpoint services
- WTOR message services

If you code 0, VTAM uses the default value of 32.

The DLRTCB start option is closely related to the MAXHNRES start option. The MAXHNRES start option controls the number of subtasks that VTAM allows for Enterprise Extender host name resolution requests. The MAXHNRES start option value is internally added to the value of the DLRTCB start option value, but the value cannot exceed the DLRTCB upper limit.

**Restriction:** The DLRTCB and MAXHNRES start options cause control blocks to be built below the 16-megabyte line. Therefore, the availability of storage might limit the value that you can code, and conversely, the value that you code might impact the availability of the storage. Also, if several dump-load-restart subtasks are active at the same time, the value that you code for this start option might affect performance. A value that is too low can cause VTAM initialization to stall or suspend a session initiation for a function that requires a TCB until a TCB becomes available.

**Results:**

- If VTAM is initialized as an APPN network node (NODETYPE=NN) and you specify a value less than 5 for the DLRTCB start option, then the value for DLRTCB is set to 5. Two TCBs are needed to process APPN directory services data sets and topology and routing services data sets, and at least three TCBs are needed for other dump-load-restart functions. If VTAM is initialized as a pure subarea node (NODETYPE start option not specified) and you specify a value less than 3 for the DLRTCB start option, then the value for DLRTCB is set to 3.
- No more than 2 TCBs will be used for cryptographic processing.
- Issuing a D NET, VTAMOPTS command displays the specified DLRTCB value; it does not display the combined DLRTCB and MAXHNRES value.

**DLURSAW start option**
Notes:
1  DLURSAW is meaningful only if the NODETYPE=NN start option is also used.

Specifies whether there will be an NLDM application started that needs to receive information about any RTP path switches involving a DLUR that is the end point of an RTP pipe. This process is to update the NLDM of RSCV updates when the DLUR’s pipe is path switched. The endpoint of the HPR pipe that path switched, is not this VTAM. By default this capability will be enabled in the host. Whether path switch updates are actually received by NLDM also depends on the capability and configuration of the DLUR.

Note: If SAVERSCV=NO is coded, then the session RSCV portion of the DLUR session awareness data is freed when the session is active. Thus, if NLDM is not active at the time, the session RSCV is not available at NLDM warm start. Conversely, if SAVERSCV=YES is specified, then the session RSCV is saved for NLDM warm start, at a cost of increased z/OS storage utilization. SAVERSCV has no influence on z/OS saving the HPR session awareness data that the DLUR supplied. It is maintained as long as a single dependent LU has a session that had been reported as being associated with that HPR pipe.

**DLURSAW=NO**

Specifies that DLUS/R function will not report the RSCV updates to NLDM when the DLUR's pipe is path switched.

Note: This could result in NLDM not getting RSCV updates and the DISPLAY SESSIONS PATHINFO=YES display would be incomplete.

**DLURSAW=YES**

Specifies that DLUS/R function will report the RSCV updates to NLDM when the DLUR’s pipe is path switched, if the DLUR is capable of reporting this information and is configured to do so.

**DSIRFMSG start option**
Controls the display of IST663I, IST664I, and IST889I messages when searches that might not result in a session being established fail to locate the target resource. The FSIRFMSG and RSIRFMSG start options can be used in conjunction with DSIRFMSG to generate additional messages as part of this message group.

- The FSIRFMSG option controls whether the IST1704I, IST1705I, IST894I, and IST895I messages are displayed.
- The RSIRFMSG option controls whether the IST1460I, IST1461I, IST2102I, IST2103I, and IST2104I messages are displayed.

**DSIRFMSG=ALLSSCP**
- Specifies that messages are to be issued in all SSCP s.

**DSIRFMSG=OLUSSCP**
- Specifies that messages are to be issued in the SSCP of the original logical unit (OLU) only.

**DSIRFMSG=NONE**
- Specifies that messages are not to be issued in any SSCP.

You can change the value of DSIRFMSG with the MODIFY VTAMOPTS command while VTAM is running.

**Recommendation:** Because of the volume of messages that can be generated, it is not recommended that this option be enabled during normal operation. Instead, it is recommended that this option be enabled (using the MODIFY VTAMOPTS command) on all necessary hosts only when trying to diagnose specific problems. Once the problem has been diagnosed or documentation has been collected, this option should be disabled once again (using the MODIFY VTAMOPTS command).

**DSPLYDEF start option**
Notes:
1. If the DSPLYMAX start option value is less than 100, that value is the default for DSPLYDEF.

### DSPLYDEF

- Dependencies: None
- Range: 1–value of DSPLYMAX

Limits the number of messages displayed when the following commands are issued without the MAX operand:

- DISPLAY ADJSSCPS
- DISPLAY APINGTP
- DISPLAY AUTOLOG
- DISPLAY CDRSCS
- DISPLAY CPCP
- DISPLAY EEC
- DISPLAY EXIT
- DISPLAY GROUPS
- DISPLAY LINES
- DISPLAY MAJNODES
- DISPLAY PENDING
- DISPLAY RTPS
- DISPLAY SRCHINFO
- DISPLAY STATS
- DISPLAY TABLE,SCOPE=ALL
- DISPLAY TGPS
- DISPLAY TRL

You can change the value of DSPLYDEF with the MODIFY VTAMOPTS command while VTAM is running.

**DSPLYMAX start option**
DSPLYMAX = 65535

Limits the value that can be specified for the DSPLYDEF start option and for the MAX operand on the following commands:

- `DISPLAY ADJSSCPs`
- `DISPLAY APINGDTP`
- `DISPLAY APINGTP`
- `DISPLAY APPLS`
- `DISPLAY AUTOLOG`
- `DISPLAY CDRMS`
- `DISPLAY CDRSCS`
- `DISPLAY CLSTRS`
- `DISPLAY CPCP`
- `DISPLAY DIRECTRY,ID=*\.name`
- `DISPLAY EE`
- `DISPLAY EEDIAg`
- `DISPLAY EXIT`
- `DISPLAY GRAFFIN`
- `DISPLAY GROUPS`
- `DISPLAY INOPCODE`
- `DISPLAY LINES`
- `DISPLAY LUGROUPS,SCOPE=ALL`
- `DISPLAY MAJNODES`
- `DISPLAY PATHTAB`
- `DISPLAY PENDING`
- `DISPLAY RSCLIST`
- `DISPLAY RTPS`
- `DISPLAY SESSIONS,LIST=ALL`
- `DISPLAY SRCHINFO`
- `DISPLAY STATIONS`
- `DISPLAY STATS`
- `DISPLAY STORUSE`
- `DISPLAY TABLE,SCOPE=ALL`
- `DISPLAY TERMS`
- `DISPLAY TGPS`
- `DISPLAY TOPO,LIST=UNRCHTIm`
- `DISPLAY TRL`
- `DISPLAY USERVAR`

**DSPLYWLD start option**
Specifies whether wildcards are permitted in certain DISPLAY commands. See z/OS Communications Server: SNA Operation to determine which display commands are governed by this start option.

You can change the value of DSPLYWLD with the MODIFY VTAMOPTS command while VTAM is running.

**DSPLYWLD=FULLWILD**
Specifies that wildcards are permitted in DISPLAY commands from all network operators. Wildcards are permitted in DISPLAY commands from program operator applications whose APPL definition statement indicates DSPLYWLD=YES.

**DSPLYWLD=NOWILD**
Specifies that wildcards are not permitted in any DISPLAY commands.

**DSPLYWLD=OPERONLY**
Specifies that wildcards are permitted in DISPLAY commands from the network operator, but not from program operator applications.

**DSPLYWLD=POAONLY**
Specifies that wildcards are permitted in DISPLAY commands from program operator applications whose APPL definition statement indicates DSPLYWLD=YES. Wildcards are not permitted from the network operator.

**DUPDEFS start option**
Specifies whether VTAM should continue searching for the target resource when the resource has been found, but is not active. The search is continued, because it may find a duplicate active definition for the resource. This start option should be set on the VTAM that owns the resource or on the VTAM acting as interchange node for resources located in subarea. You can change the value of DUPDEFS with the MODIFY VTAMOPTS command while VTAM is running.

**DUPDEFS=ALL**

Specifies that resources may be defined on more than one node in the network. If this node receives a search for a resource that is not active, VTAM should continue searching for the resource because an active instance may reside somewhere else in the network.

Tip: There is no support to continue a subarea search for an application. An application is always a subarea resource. If a session request is initiated from the host where an APPL is defined, and if the APPL definition is not in an ACTIV state (for example: CONCT, NEVAC, or INACT), then that session request will fail with a sense code that indicates that the PLU or the SLU is not active.

**DUPDEFS=NONE**

Specifies that VTAM should not continue searching for a resource when it is found but not active. This value would be coded if there are not duplicate definitions in the network.

**DUPDEFS=APPL**

Specifies that applications may be defined on more than one node in the network. If this node receives a search for an application that is not active, VTAM should continue searching for the application because an active instance may reside somewhere else in the network.

Tip: There is no support to continue a subarea search for an application. An application is always a subarea resource. If a session request is initiated from the host where an APPL is defined, and if the APPL definition is not in an ACTIV state (for example: CONCT, NEVAC, or INACT), then that session request will fail with a sense code that indicates that the PLU or the SLU is not active.

**DUPDEFS=DEPLU**

Specifies that dependent LUs may be defined on more than one node in the network. If this node receives a search for a dependent LU that is not active, VTAM should continue searching for the dependent LU because an active instance may reside somewhere else in the network.

**DYNADJCP start option**
Notes:

1. DYNADJCP is meaningful only if the NODETYPE start option is also used.

Specifies whether adjacent control point (ADJCP) minor nodes are allowed to be created dynamically and placed in ISTADJCP. The ADJCP minor node provides control and management of connections between this node and an adjacent APPN type 2.1 node. This option can be overridden at the connection level (that is, on a link station).

By specifying this option as NO and overriding its value on specific link stations, you can limit connections to a specific set of CPs over some link stations, and allow connections to any CP over other link stations.

DYNADJCP=NO
ADJCP minor nodes are not created dynamically. When a connection is established, the adjacent control point minor node must be defined and active or the connection is discontinued.

DYNADJCP=YES
ADJCP minor nodes are created as needed. Because some VTAM LEN implementations do not have an ADJCP major node, this option defaults to YES to provide compatibility for all type 2.1 connections.

Restrictions:

• If an HPR pipe is to be brought up, the DYNADJCP option is ignored.
• When you define a PU with CPNAME, the ADJACENT CP will automatically be built and stored in ISTADJCP when the PU is activated, even when still in CONCT status. When the first link to the CPNAME is activated, the DYNADJCP=NO option does not take effect for pipes towards that CPNAME. With DYNADJCP=NO, the links from unknown CPNAMEs require a VBUILD TYPE=ADJCP major node that defines every CPNAME you want to link with.

DYNASSSCP start option
Specifies whether VTAM dynamically routes session establishment requests to all active adjacent SSCPs if no appropriate adjacent SSCP table is defined.

Note: If you have coded a node to function as an end node or network node (NODETYPE=EN or NN, and HOSTSA is not coded), then DYNASSCP should not be coded because it has no effect. If you code it, error messages will be issued, but the node will come up.

**DYNASSCP=NO**
Specifies that VTAM does not perform dynamic session establishment request routing. If you have not coded the appropriate adjacent SSCP tables and have not used the SSCPDYN start option, or have not specified the CDRM owner, session establishments might fail.

**DYNASSCP=YES**
 Specifies that VTAM determines the adjacent SSCPs when routing a session establishment request across a domain or across a network. If VTAM does not locate an appropriate adjacent SSCP table, it dynamically routes the session establishment request to all active adjacent SSCPs until the correct SSCP is found.

**DYNDLGMD start option**
Specifies the default logon mode name used for dynamic cross-domain resources when the resource is acting as the secondary logical unit (SLU) and a logon mode name is not otherwise provided.

You can change the DYNDLGMD value with the MODIFY VTAMOPTS command while VTAM is running.

You can also change the default logon mode name used for a specific dynamic cross-domain resource with the MODIFY DEFAULTS or MODIFY RESOURCE command.

**DYNDLGMD=logmode_name**
- Specifies the default logon mode name for dynamic cross-domain resources.

**DYNDLGMD=NONE**
- Specifies that there is no default logon mode name for dynamic cross-domain resources.

### DYNHPPFX start option

**Notes:**

1. Two character prefix.

If DYNHPPFX is not specified, then dynamic PUs created for HPR will be prefixed with the three characters, CNR. If DYNHPPFX=cc where cc is the two character prefix that you want, then dynamic PUs created for HPR will be prefixed with the two characters, cc. In these two cases, respectively, the dynamic PU names will have the format of CNRxxxxx or ccxxxxxx.

**DYNHPPFX=prefix**
- Specifies a two character prefix that is used in creating the dynamic PUs created for HPR. A unique value will then be generated and concatenated to the defined prefix to create an eight character PU name.

### DYNLU start option
Allows you to dynamically allocate cross-domain resource (CDRSC) definitions for resources being treated as independent LUs by this VTAM. (The cross-domain resources might actually represent application programs, dependent LUs or independent LUs in some other node’s domain.)

You can override this value at the connection level by coding DYNLU on specific adjacent link stations (PU definition statements). The adjacent link station determines the boundary function link that the resource uses to access the network.

Likewise, for an adjacent control point major node, you can override the value of the DYNLU start option by specifying a different value on the DYNLU operand on the ADJCP definition statement.

**DYNLU=NO**

Specifies that VTAM will not use dynamically allocated CDRSC representations for sessions over adjacent link stations.

If you code DYNLU=NO, you are required to define all resources that can be accessed over adjacent link stations. If you do not define a given resource, sessions to or from that resource will fail.

**DYNLU=YES**

Specifies that VTAM can use dynamically allocated CDRSC representations for sessions over adjacent link stations.

If you code DYNLU=YES, you are not required to predefine all resources that can be accessed over adjacent link stations.

For subarea nodes and APPN-only nodes (HOSTSA is not specified), DYNLU=YES is meaningful only when CDRDYN=YES is coded, as the start option (either specified as YES or by the default) or as the operand.

**Result:** The DYNLU value is associated with an adjacent APPN node when the first link to that adjacent node is activated. When the DYNLU value is associated with the adjacent node, that value is propagated to all other links to that adjacent node as they are activated, regardless of the DYNLU value specified for each link. See Table 2 on page 24 for more information about determining the source of the DYNLU value assigned to an adjacent CP and attached resources.

**DYNMODTB start option**
Specifies the name of a logon mode table used for dynamic cross-domain resources to correlate each logon mode name with a set of session parameters when the resource acts as the SLU.

You can change the DYNMODTB value with the MODIFY VTAMOPTS command or the MODIFY TABLE,OPTION=LOAD command while VTAM is running.

You can also override this value for a specific dynamic cross-domain resource with the MODIFY TABLE,OPTION=ASSOCIATE command while VTAM is running.

DYNMODTB=table_name

Specifies the name of the default logon mode table used for dynamic cross-domain resources. The name specified must be the name of a logon mode table created as described in "Logon mode table" on page 1026.

DYNMODTB=NONE

Specifies that there is no default logon mode table for dynamic cross-domain resources.

**DYNPUPFX start option**

DYNPUPFX=CN

Specifies the default.

DYNPUPFX=prefix

Specifies a two character prefix that is used in creating the dynamic PUs. This prefix will be used when generating names for these dynamically built PUs. The prefix defined by the start option will be used if DYNPUPFX is not coded on the group statement for the line under which the PU is built. A unique value will then be generated and concatenated to the defined prefix to create an eight character PU name.

**DYNVNPFX start option**
Specifies the prefix to be used when generating the name for dynamic connection network PUs.

**DYNVNPFX=CNV**

When the DYNVNPFX start option is not specified, the default value is the three character prefix CNV. The PU names will then take the format CNVxxxxx, where xxxxx is a unique value which is generated and concatenated to the default prefix to create the eight character PU name.

**DYNVNPFX=prefix**

Specifies a two character prefix that is used to generate the names for dynamic connection network PUs. This prefix will be used if DYNVNPFX is not coded on the virtual node GROUP definition statement. The PU names will take the format ppxxxxx, where xxxxx is a unique value which is generated and concatenated to the specified prefix pp to create the eight character PU name.

**EEHPRANR start option**

Notes:
1. EEHPRANR is meaningful only when the NODETYPE=NN start option is also used.

Determines whether VTAM maintains diagnostic EE counters for inbound HPR packets that are ANR routed through this network node. These counters are visible through the DISPLAY EE command output and through Network Management Interface application (SNAMGMT) EE displays.

**EEHPRANR=NO**

Specifies that VTAM does not maintain diagnostic EE counters for inbound HPR packets that are routed by using the ANR protocol through this node. However, VTAM maintains diagnostic EE counters for outbound HPR packets that are routed by using the ANR protocol through this node.

**EEHPRANR=YES**

Specifies that VTAM maintains diagnostic EE counters for both inbound and outbound HPR packets that are routed by using the ANR protocol through this node.

Note: In environments with large numbers of RTP pipes, specifying EEHPRANR=YES might increase CPU usage to maintain these statistics.
For Enterprise Extender (EE), this option determines whether VTAM accepts inbound UDP packets that are destined to one of the five EE ports (defined by IPPORT), that specify an ephemeral source port. You can modify this start option.

**EEPORTCK=NO**

Specifies that VTAM accepts inbound UDP packets that are destined to one of the five EE ports, which specify an ephemeral source port.

**EEPORTCK=YES**

Specifies that VTAM does not accept inbound UDP packets that are destined to one of the five EE ports, which specify an ephemeral source port. If the source and destination ports do not match, the packet is immediately discarded.

**Result:** Some EE platforms use an ephemeral source port for EE communications. Specifying EEPORTCK=YES in these configurations might result in loss of connectivity to these EE platforms. Similar connectivity issues might result if your EE communications uses network address port translation (NAPT).

### EEVERIFY start option

Specifies whether VTAM should automatically send an LDLC probe to the remote partner to determine if all ports are accessible during the activation of the EE connection. This option might also specify the time interval at which VTAM should send the probe on active EE connections.

You can change the EEVERIFY value with the MODIFY VTAMOPTS command while VTAM is running. The MODIFY VTAMOPTS command changes the value of all active connections immediately.

You can display the EEVERIFY value by issuing the DISPLAY VTAMOPTS command.

**Rule:** EEVERIFY is meaningful only if VTAM provides RTP-level HPR support. The NODETYPE start option must be specified and the RTP value must be specified on the HPR start option.

**EEVERIFY=ACTIVATE**

Specifies that VTAM should automatically send an LDLC probe to the remote partner to determine whether all ports are accessible during the activation of the EE connection. If one or more ports are not reachable or are blocked, then VTAM issues an error message and does not activate the EE connection.
EEVERIFY=NEVER
Specifies that VTAM should never automatically send an LDLC probe to the remote partner to determine if all ports are accessible during the activation of the EE connection.

EEVERIFY=Number_of_minutes
Specifies that VTAM should automatically send an LDLC probe during the activation of the EE connection and that VTAM should continue to periodically send LDLC probes at intervals specified by the Number_of_minutes value while the connection is active. Valid values for Number_of_minutes are in the range 1 - 1440 minutes (1440 minutes specifies 1 day). If one or more ports are not reachable or are blocked at activation, then VTAM issues an error message and does not activate the EE connection. If one or more ports are not reachable after the connection is active, VTAM issues an error message, but the connection remains active.

ENCRPREF start option

ENCRPREF=None
Specifies the prefix to be used for common cryptographic architecture (CCA) key-encrypting key (KEK) labels.

You can change the ENCRPREF value with the MODIFY VTAMOPTS command while VTAM is running.

ENCRPREF=None
Specifies that no prefix is to be used for the common cryptographic architecture (CCA) key-encrypting key (KEK) labels.

Note: The default or the IBM Integrated Cryptographic Service Facility (ICSF) in CCA and the Transaction Security System (TSS) program product running in PCF mode is no prefix.

ENCRPREF=value
Specifies the 1–8-character prefix to be used for common cryptographic architecture (CCA) key-encrypting key (KEK) labels.

Note: Transaction Security System (TSS) program product running in CCA mode has a configuration data set. The PCF-KEY-PREFIX command specifies the first key label token name (the first 1-8 characters of the key label). The default for PCF-KEY-PREFIX is $$CUSB$$.

ENCRYPTN start option
Notes:

1. ENCRYPTN=CCA needs to be coded when Triple Des Encryption is desired.

Specifies that 24-bit capable or 31-bit capable cryptography is installed.

Note: If the ENCRYPTN start option is not specified, VTAM will determine if cryptography is active and the addressing mode it supports.

**ENCRYPTN=YES**

Specifies that VTAM supports cryptography and determines the addressing mode at VTAM initialization.

At initialization:

- If the currently active cryptographic product supports 24-bit addressing mode, VTAM supports cryptographic products that support 24-bit or 31-bit addressing mode and VTAM I/O buffers are placed in 24-bit storage.
- If the currently active cryptographic product supports 31-bit addressing mode, VTAM only supports cryptographic products that support 31-bit addressing mode and VTAM I/O buffers are placed in 31-bit storage.
- If no cryptographic product is currently active, VTAM only supports cryptographic products that support 31-bit addressing mode and VTAM I/O buffers are placed in 31-bit storage.

**ENCRYPTN=NO**

Specifies that VTAM does not support cryptography, regardless of whether or not a cryptographic product is currently active.

**ENCRYPTN=CCA**

Specifies that VTAM use a common cryptographic architecture (CCA) product to satisfy cryptography requests. If a CCA product is not available at start time the request is rejected and the session fails. VTAM supports this product using 24-bit or 31-bit addressing mode.

Sessions that require TDES24 (Triple DES encryption) will fail if the CCA product is not available. In addition, the start option ENCRYPTN=CCA is required if TDES24 is to be used.

**ENCRYPTN=CUSP**

Specifies that VTAM use an IBM Cryptographic Unit Support (CUSP) program product to satisfy a request for cryptographic support. If a CUSP product is not available at start time, the request is rejected and the session fails. VTAM supports CUSP using 24-bit or 31-bit addressing mode.

**ENCRYPTN=24**

Specifies that VTAM supports cryptography in 24-bit mode. VTAM I/O buffers are placed in 24-bit storage. VTAM uses cryptographic products that support either 24-bit or 31-bit storage addresses.
ENCRTPN=31
Specifies that VTAM supports cryptography in 31-bit mode. VTAM I/O buffers are placed in 31-bit storage. VTAM only uses cryptographic products that support 31-bit addressing mode. If VTAM receives a cryptography request and the only available cryptographic product supports 24-bit addressing mode, the request fails.

ENHADDR start option

| ENHADDR=NO | ENHADDR=YES |

Specifies whether VTAM can assign element addresses greater than 65,535. These addresses are assigned to resources establishing sessions within this subarea and as a base address to application programs within this subarea.

ENHADDR=NO
Specifies that you cannot use high-order element addresses for application programs.

If you specify ENHADDR=NO all element addresses that VTAM obtains for resources in this subarea come out of the 64k element pool. This includes each active (connectable) APPL (two if PARSESS=YES), each activated local LU and PU, and each dynamically defined LU and PU (once connected). We must also obtain a session partner element address for each session in which we are the endpoint or intermediate routing point (unless the session path to that partner is over a FID_4 link - because, in this case, the element address will originate from another subarea's address space). APPN session partner element addresses come out of this subarea's 64K element address pool if you specify ENHADDR=NO. When addresses in the high-order element address pool are available, every SSCP_LU and SSCP_PU element address for DLUR/DLUS, EE line and PU element addresses, and RTP PU element addresses will be allocated from the high-order pool, even if ENHADDR =NO is specified.

ENHADDR=YES
Specifies that you can use high-order element addresses for application programs.

If you specify ENHADDR=YES and have an application program that used the network-qualified address pair control vector X'15', see z/OS Communications Server: New Function Summary.

SNA PUs have an architected element address pool limit of 64k, however, VTAM has created an additional pool of address spaces whose total is 16348k. In addition, VTAM must still obtain an element address from the 64k pool if the session path, in or out of the host, is over a FID_4 link.

A SNA network address is in the form of a four byte subarea address and a two byte element address (as in 'ssssss eeee'), however, only the two low-order bytes of the subarea address are actually used. The mechanism VTAM uses to create and access a different element address space is to use the two high-order bytes as an index field (as in iiiissss eeee).

ENSEMBLE start option
Notes:

1 The ENSEMBLE setting is used to either permit or deny connectivity to the intraensemble data network (IEDN) and the intranode management network (INMN) by allowing or denying activation of OSX and OSM interfaces.

When the IBM zEnterprise® System (zEnterprise) node is configured at the Hardware Management Console (HMC) as a member of an ensemble, then the ENSEMBLE start option determines whether this specific z/OS image should be included as a member of the ensemble. When the zEnterprise node is not configured at the HMC as a member of an ensemble, this start option has no meaning. See z/OS Communications Server: IP Configuration Guide for details about configuring a zEnterprise node to participate in an ensemble.

ENSEMBLE=NO

Specifies that this z/OS Communications Server does not permit connectivity to either the intraensemble data network (IEDN) or the intranode management network (INMN) by denying activation of OSX and OSM interfaces. The default value for the ENSEMBLE start option is NO.

ENSEMBLE=YES

Specifies that this z/OS Communications Server permits connectivity to the IEDN and to the INMN. Access to these networks is through OSA-Express3 or later features that are configured with the following CHPID types:

- OSX (IEDN)
- OSM (INMN)

When the zEnterprise node is configured at the HMC as a member of an ensemble, you can change the value of the ENSEMBLE start option with the MODIFY VTAMOPTS command while VTAM is running.

Result: The ENSEMBLE setting is used to either permit or deny activation attempts for OSX and OSM interfaces. Modifying the ENSEMBLE start option from YES to NO does not cause z/OS Communications Server to take action on any active OSX or OSM interfaces.

ESIRFMSG start option
Controls the display of IST891I, IST892I, and IST893I. These messages are issued when a session initiation request fails and extended sense data exists.

The ESIRFMSG start option is subordinate to the SIRFMSG and LSIRFMSG start options and SIRFMSG coded on a CDRSC or APPL. Therefore, you must code SIRFMSG as ALLSSCP or OLUSSCP or LSIRFMSG as ALNNNS or OLUNNS to receive the ESIRFMSG messages.

You can change the value of ESIRFMSG with the MODIFY VTAMOPTS command while VTAM is running.

**ESIRFMSG=ALLSSCP**
- Specifies that messages are issued in all SSCPs.

**ESIRFMSG=OLUSSCP**
- Specifies that messages are issued only in the origin logical unit (OLU) SSCP.

**ESIRFMSG=NONE**
- Specifies that no messages are issued in any SSCP.

**Tip:** When ESIRFMSG is used in conjunction with LSIRFMSG, then ALLSSCP is equivalent in function to ALNNNS and OLUSSCP is equivalent in function to OLUNNS.

**EXPFLTRM start option**
Specifies the number of data flow control (DFC) and session control (SC) request units (RU), that a boundary-attached LU can send without receiving a response from the partner LU. If an LU-LU session exceeds this threshold, VTAM deactivates the LU-LU session. If the `expedited_flow_count` defaults to 0 or is coded 0, the function is disabled.

Expedited data flows for an LU-LU session are not governed by the SNA pacing rules. As a result, if a hardware error, or a 3270 emulator logic error occurs, excessive expedited data might flow into VTAM. These unpaced flows may cause excessive resources (storage, CPU cycles) to be consumed to process these requests.

**Guideline:** Coding a value too small can cause LU-LU sessions to be deactivated. Some environments might need to specify a larger value for EXPFLTRM to avoid unwanted LU-LU session deactivations.

**Rule:** When an LU-LU session meets or exceeds the defined EXPFLTRM threshold, the session is deactivated and message IST2113I is issued.

**FLDTAB start option**
Specifies whether VTAM uses a message-flooding prevention table to suppress duplicate messages.

**FLDTAB=ISTMSFLD**
Indicates that VTAM uses the internal message flooding table.

**FLDTAB=NONE**
Indicates that VTAM will not suppress duplicate messages that are sent to the operator console or system hardcopy log.

**FLDTAB=table_name**
Indicates that VTAM uses a user-specified message-flooding prevention table to determine which duplicate messages are to be suppressed. `table_name` is the 1–8 character name of the table stored in the VTAM definition library.

If an attempt to load the specified table fails, VTAM issues an error message and uses the VTAM internal table.

If you specify ISTMSFLD as the table name, VTAM will use the IBM-supplied table located in the VTAM definition library.

**FSIRFMSG start option**

Controls the display of IST894I, IST895I, IST1704I, and IST1705I. These messages are issued when a session initiation request fails because trial and error routing, using an adjacent SSCP table, exhausted the table and was unable to locate the destination LU.

**Requirement:** The FSIRFMSG start option is subordinate to the SIRFMSG, DSIRFMSG, and LSIRFMSG start options and SIRFMSG coded on a CDRSC or APPL. Therefore, you must code SIRFMSG or DSIRFMSG as ALLSSCP or OLUSSCP or LSIRFMSG as ALLNNS or OLUNNS to receive the FSIRFMSG messages.

You can change the value of FSIRFMSG with the MODIFY VTAMOPTS command while VTAM is running.

**FSIRFMSG=ALLSSCP**
Specifies that messages are issued in all SSCPs.

**FSIRFMSG=OLUSSCP**
Specifies that messages are issued only in the origin logical unit (OLU) SSCP.

**FSIRFMSG=NONE**
Specifies that no messages are issued in any SSCP.
Tip: When FSIFMSG is used in conjunction with LSIFMSG, then ALLSSCP is equivalent in function to ALLNNS and OLSSCP is equivalent in function to OLUNNS.

**GVBKDLY start option**

Determines whether VTAM internally paces the GIVEBACK subcommands (GIVEBACKs) that are generated from a VARY INACT,TYPE=GIVEBACK command. The GVBKDLY start option only controls GIVEBACK commands that use wildcard values for the ID operand to deactivate dependent LU requesters (DLURs). Specifying GVBKDLY=YES causes VTAM to internally delay scheduling the GIVEBACK subcommands. It results in more efficient processing when the wildcard value for the ID operand on a VARY INACT,TYPE=GIVEBACK command matches many DLURs. This start option can be modified.

**Note:** The use of wildcard values on GIVEBACK commands is controlled by the value of the VARYWLD start option.

**GVBKDLY=NO**

Specifies that VTAM does not internally pace the GIVEBACK subcommands that are generated from a VARY INACT,TYPE=GIVEBACK command.

**GVBKDLY=YES**

Specifies that VTAM internally paces the GIVEBACK subcommands that are generated from a VARY INACT,TYPE=GIVEBACK command. The GVBKDLY start option only controls GIVEBACK commands that use wildcard values for the ID operand to deactivate dependent LU requesters (DLURs).

**Note:** VTAM issues message IST2416I after it schedules the last GIVEBACK that is generated from the previous command. If you have multiple wildcard patterns to use for the VARY INACT command, wait until the IST2416I message is displayed before you issue another VARY INACT,TYPE=GIVEBACK command with a different wildcard pattern.

**GWSSCP start option**
Specifies that an SSCP can be a gateway SSCP. If you code GWSSCP, the SSCP can still process CDRSCs defined with the NETID operand.

GWSSCP=NO
Specifies that this SSCP cannot:
- Reroute initiation requests, so it cannot be an intermediate SSCP in the session setup path.
- Perform the gateway functions.

GWSSCP=YES
Allows the SSCP to:
- Reroute initiation requests.
- Perform gateway functions in a session setup path.

Note that GWSSCP=YES is meaningful only for subarea nodes and APPN interchange nodes (NODETYPE=NN and HOSTSA is coded).

If you code GWSSCP=YES on the receiving CDRM definition, NETID is required on the NETWORK definition statement of all external CDRMs.

If you have coded a node to function solely as an end node or network node (NODETYPE=EN or NN, and HOSTSA is not coded), then you should not code GWSSCP. If you code GWSSCP in this case (specifying either YES or NO), it is ignored and an informational message is issued.

If you have coded a node to function as a migration data host (NODETYPE=EN and HOSTSA is coded), then you should code GWSSCP=NO. If you code GWSSCP=YES, or allow it to default to YES, VTAM forces GWSSCP to NO and issues an informational message.

**HNTSIZE start option**
HNTSIZE = 4080

Specifies the size of the host node table used to find element addresses. This start option is no longer user defined. If specified in the VTAM start list, the value will be forced to 4080 bytes.

HOSTNAME start option

HOSTNAME = fully_qualified_hostname

Specifies the default host name associated with the local TCP/IP stack to be used for Enterprise Extender connections involving this node. The HOSTNAME you code should be a fully qualified TCP/IP domain name.

Recommendation: The value of hostname can be resolved to either an IPv4 address or an IPv6 address, but it is highly recommended that hostname resolve to a single VIPA address.

If the local TCP/IP stack has multiple static IPv4 VIPAs, use the value of hostname to optionally provide a host name for remote Enterprise Extender nodes to perform name-to-address resolution to acquire the appropriate static VIPA, or NAT address, to use in order to communicate with this host. The value of hostname will also be resolved by this node to acquire the static IPv4 VIPA to use for Enterprise Extender connections. The value of the IPADDR start option, or the first VIPA defined to the TCP/IP stack identified by TCPNAME, may be used as alternatives for identifying the default IPv4 static VIPA address for Enterprise Extender purposes.

You can specify a node-wide value for HOSTNAME to use IPv6 protocols for some or all of your EE connections. The value of hostname will be used by this node to perform name-to-address resolution in order to determine the appropriate static IPv6 VIPA for use with Enterprise Extender connections. Remote Enterprise Extender nodes wanting to communicate with this node will also perform name-to-address resolution using hostname to acquire the appropriate static VIPA required to ultimately reach this node. If the local TCP/IP stack has multiple static IPv6 VIPAs, use the value of hostname to indicate the appropriate static VIPA to use in order to communicate with this host. Alternatively, you can use the value of the IPADDR start option to identify the default IPv6 static VIPA address for EE connections.

HOSTNAME can also be specified on the Enterprise Extender XCA major node GROUP definition statement. Specifying the HOSTNAME operand on the Enterprise Extender XCA major node GROUP definition statement may be necessary under the following conditions:

• Your Enterprise Extender configuration uses both IPv4 and IPv6 addressing protocols.
• Your Enterprise Extender configuration uses multiple static VIPAs.

Note:
1. The fully qualified hostname may be up to 64 characters in length. See z/OS Communications Server: SNA Network Implementation Guide for more information about HOSTNAME length considerations.

2. The HOSTNAME start option is used only when the XCA major node PORT definition statement is coded as MEDIUM=HPRIP, and neither the HOSTNAME nor IPADDR operand is coded on the GROUP definition statement.

3. HOSTNAME and IPADDR cannot be modified using one MODIFY VTAMOPTS command. If both start options are specified on the same MODIFY command, they will both be ignored and message IST1917I will be generated.

4. The HOSTNAME start option can be dynamically modified using MODIFY VTAMOPTS under certain conditions. An attempt to modify HOSTNAME to be a fully qualified hostname will be failed with message IST448I if a static VIPA address has been obtained from the TCP/IP stack named in the TCPNAME start option and is being used for active Enterprise Extender connections. All active Enterprise Extender lines must be deactivated before the MODIFY VTAMOPTS, HOSTNAME=fully qualified hostname will be accepted.
   If the MODIFY command is successful, displays of VTAM start options will show the new value immediately; however, the new value will not be used until all Enterprise Extender lines whose GROUP definition statements do not have HOSTNAME or IPADDR explicitly coded are inactive. Any subsequent activation for an Enterprise Extender line whose GROUP definition statement does not have HOSTNAME or IPADDR explicitly coded will make use of the new HOSTNAME start option value. If the IPADDR start option is currently in use when the MODIFY VTAMOPTS is performed to activate HOSTNAME, the IPADDR start option is set to 0.0.0.0 as part of processing the MODIFY VTAMOPTS.

   The value of NONE can be specified for the HOSTNAME value on the MODIFY VTAMOPTS command as a mechanism for clearing the value of the current HOSTNAME start option. This option will fail, however, with message IST448I, if the Enterprise Extender XCA major node is active.

   Enterprise Extender major node GROUP definition statements that have HOSTNAME or IPADDR explicitly coded are unaffected by setting the HOSTNAME start option, and are unaffected by the MODIFY VTAMOPTS processing.

5. If neither the HOSTNAME nor IPADDR operand is specified on any of the GROUP definition statements within the Enterprise Extender XCA major node, then either the HOSTNAME, TCPNAME or IPADDR start option must be specified in order to initiate an Enterprise Extender link. If VTAM has been activated without the specification of one of these options, then the MODIFY VTAMOPTS command can be used to set one of these options. If an Enterprise Extender link activation is attempted without HOSTNAME, TCPNAME or IPADDR specified, then the link activation will fail with the IST1890I message.

6. If HOSTNAME or IPADDR is specified on any GROUP definition statement within the Enterprise Extender (XCA) definition, it must be specified on all Enterprise Extender GROUP definition statements (or defaulted from the VTAM start options).

7. The TCPNAME start option is required in a CINET environment if any of the following conditions exist:
   - The IPADDR parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
• The HOSTNAME parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
• The HOSTNAME keyword is specified as a VTAM start option.

If OMVS is not initialized at Enterprise Extender line activation, a Common INET environment is assumed. In this case, if an Enterprise Extender link activation is attempted without TCPNAME specified, then the link activation will fail with message IST1897I.

8. If both the IPADDR and the HOSTNAME start options are specified on the VTAM START command, the value of the IPADDR start option is ignored.
9. HOSTNAME is meaningful only if the NODETYPE start option is also used.
10. See z/OS Communications Server: IP Configuration Guide for information about UNIX Systems Services security considerations and required security definitions.

HOSTPU start option

HOSTPU = ISTPUS
HOSTPU = host_subareaPu_name

dependencies: see notes

Specifies the network name of the VTAM host subarea physical unit for this host. It is recommended that you code HOSTPU if you are using the NetView program or NLDM. If you do not code HOSTPU, VTAM uses ISTPUS as the host PU name.

Note:
1. The name for HOSTPU must not be the same as the name for this CDRM or the name coded for the SSCPNAME start option for this SSCP. If the name for HOSTPU is the same as the name for SSCPNAME, HOSTPU is set to ISTPUS. This host PU name should be network-unique.
2. You are prompted for the correct name if a reserved name other than ISTPUS is specified on the HOSTPU start option or if the HOSTPU name and the SSCPNAME are the same.
3. If you do not use the default name (ISTPUS), the subarea can still be defined to the system as ISTPUS. The name that you use as the HOSTPU value is interchangeable with the name ISTPUS in VTAM commands.

HOSTSA start option
Notes:
1. HOSTSA specifies the subarea number of this VTAM. If HOSTSA is not coded, then a default subarea number of 1 is used.

   range: 1– to the maximum number of subareas allowed for the network

If HOSTSA= is coded and SACONNS= is not coded or SACONNS=YES is coded, then this VTAM will provide subarea function. If HOSTSA= is not coded, or HOSTSA= is coded and SACONNS=NO is coded, then this VTAM will not provide subarea function.

The combination of start options HOSTSA, NODETYPE, and SACONNS determines the overall node configuration and level of function support. The start options and resulting node configurations are intended to provide a smooth migration for existing subarea hosts.

See Table 70 for a summary of the node type functions. When the node is configured as a network node or end node, attempts to activate FID4 (PU types 4 or 5) connections will fail. However, if the node is a migration data host or an interchange node, FID4 connections are allowed.

Table 70. Node type functional summary

<table>
<thead>
<tr>
<th>Node type</th>
<th>NODETYPE</th>
<th>HOSTSA</th>
<th>SACONNS</th>
<th>CP-CP sessions</th>
<th>SSCP-SSCP sessions</th>
<th>NCP ownership</th>
<th>Interchange function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subarea Node</td>
<td>(not coded)</td>
<td>n</td>
<td>n/a</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Interchange Node</td>
<td>NN</td>
<td>n</td>
<td>(not coded) or YES</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Migration Data Host</td>
<td>EN</td>
<td>n</td>
<td>(not coded) or YES</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Network Node</td>
<td>NN</td>
<td>(not coded)</td>
<td>n/a</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Network Node</td>
<td>NN</td>
<td>n</td>
<td>NO</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>End Node</td>
<td>EN</td>
<td>(not coded)</td>
<td>n/a</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>End Node</td>
<td>EN</td>
<td>n</td>
<td>NO</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
</tbody>
</table>

Note: 1. n represents a subarea host number.

Note: If you want a node to function solely as an end node or network node (NODETYPE=EN or NN), then HOSTSA should not be coded or SACONNS=NO must be coded.

HOSTSA must be less than or equal to the maximum number of subareas specified (MXSUBNUM start option). See “MXSUBNUM start option” on page 900 for more information on the MXSUBNUM start option.
**Note:** If you plan to incorporate a non-VTAM host, VTAM's SSCPID and HOSTSA start options values should be coded so that both values are either higher or lower than the non-VTAM host.

**HOTIOTRM start option**

```
HOTIOTRM = 0
HOTIOTRM = percentage_of_buffer_pool
```

range: 0 or 10–99

Specifies the percentage of the current size of the IO buffer pool that a single session must have allocated to it to cause VTAM to take corrective action. Corrective action differs depending on the type of session involved:

- When the session type is SSCP-LU: deactivate the LU.
- When the session type is SSCP-PU: deactivate the PU.
- When the session type is LU-LU: terminate all sessions between the two logical units.

If you omit HOTIOTRM or specify 0, VTAM will not terminate sessions based on IO buffer pool usage.

You can change the value of HOTIOTRM with the MODIFY VTAMOPTS command while VTAM is running.

**Note:**

1. It is very important that a proper expansion limit (XPANLIM) be specified on the IO buffer pool when HOTIOTRM is specified. An expansion limit allows VTAM to detect one or more hot sessions prior to complete CSA exhaustion. See the description of the IOBUF buffer pool for a complete description of the XPANLIM parameter.

2. Investigate past VTAM console logs for VTAM IST930I messages when there did not seem to be a problem. Specifying HOTIOTRM greater than the percentages in these past IST930I messages will cause VTAM to not terminate sessions that “normally” use large amounts of the IO buffer pool. Usually, these session have not had proper session level pacing values specified. You might wish to consider specifying session pacing that will reduce the amount of IO buffers used by these sessions.

3. If past console logs are not available, or no IST930I messages are found, specify a HOTIOTRM value of 50.

**HPR start option**
Notes:

1. HPR is meaningful only if NODETYPE is also used.

Specifies the level of High-Performance Routing (HPR) support provided by VTAM.

**HPR=ANR**

Indicates this VTAM (and NCP in a composite network node) provides ANR-level HPR support. Coding HPR=YES on a CDRM or PU definition statement (or allowing it to default to YES) results in ANR-level HPR support over that connection.

HPR=ANR is not valid at an end node (NODETYPE=EN).

**HPR=RTP**

Indicates this VTAM provides RTP-level HPR support. Coding HPR=YES on a CDRM or PU definition statement (or allowing it to default to YES) results in RTP-level HPR support over that connection.

**HPR=NONE**

Indicates this VTAM (and NCP in a composite network node) provides no HPR support.

**HPR=(RTP,ANR)**

Indicates this VTAM provides RTP-level HPR support only if HPR=YES is coded on the CDRM or PU definition statement. Letting the HPR operand default on a CDRM or PU definition statement results in only ANR-level HPR support over that connection (except for dynamic XCF connections, which are always treated as having RTP-level HPR support).

HPR=(RTP,ANR) is not valid at an end node (NODETYPE=EN).

**Note:** The HPR=(RTP,ANR) start option is recommended only for migration purposes. It can cause unexpected results in HPR route selection. See [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/en/POWER5/zos/zos_communication_server) for additional information.

**HPR=(RTP,NONE)**

Indicates this VTAM provides RTP-level HPR support only if HPR=YES is coded on the CDRM or PU definition statement. Letting the HPR operand default on a CDRM or PU definition statement results in no HPR support over that connection.

**HPR=(ANR,NONE)**

Indicates this VTAM provides ANR-level HPR support only if HPR=YES is coded on the CDRM or PU definition statement. Letting the HPR operand default on a CDRM or PU definition statement results in no HPR support over that connection.

HPR=(ANR,NONE) is not valid at an end node (NODETYPE=EN).
Note: When end to end path switching is performed, a BN, which is not an endpoint or the NNS for the endpoint, may be asked to compute an HPR route for a piece of the end to end route. To support this function, the BN needs to provide RTP-level HPR support. Ensure the BN and HPR start options are coded accordingly.

**HPRARB start option**

```
HPRARB –= RESPMODE
(1)
HPRARB –= BASE
```

Notes:
1. This start option was provided by APAR OW36113 for use in migration to VTAM V4R5. Do not use this option unless you use the default value of RESPMODE.

Specifies the algorithm to be used in calculations for adaptive pacing.

**HPRCLKRT start option**

```
HPRCLKRT –= STANDARD
HPRCLKRT –= ADAPTIVE
```

The HPR clock is the mechanism that processes the timer requests for all RTP pipes. The HPRCLKRT start option specifies whether the HPR clock runs in standard or adaptive mode. These modes control how reactive VTAM is to the packet loss that occurs on RTP pipes that run over Enterprise Extender.

**Guideline:** Use the default value of HPRCLKRT=STANDARD when you are not running Enterprise Extender.

**HPRCLKRT=ADAPTIVE**

Specifies that the HPR clock operates in adaptive mode. Running the HPR clock in adaptive mode enables the clock to toggle between standard and high modes of operation when the clock is servicing RTP pipes that are experiencing packet loss. When packet loss occurs, the clock speed adapts to run in high mode. The RTP pipes that are experiencing packet loss are scheduled more frequently, which enables the RTP pipes to recover more quickly. There might be some CPU overhead for running the HPR clock in high mode. When packet loss is no longer occurring, the clock returns to the standard mode.

The adaptive mode is designed to increase the overall performance of RTP pipes when packet loss is occurring. However, RTP performance and CPU overhead are system dependent and might vary.

**Restriction:** If you code HPRCLKRT=ADAPTIVE, the following restrictions apply:
- This mode is meaningful only in Enterprise Extender configurations that have a defined capacity of 1 Gb or higher access speeds.
- Do not use this mode for uniprocessor configurations or in configurations that are CPU bound.
**HPRLKRT=STANDARD**

Specifies the HPR clock is to operate in standard mode. The HPR clock will run at the standard interval to service RTP pipes. Standard mode is the default mode for the HPR clock rate.

**HPRITMSG start option**

![HPRITMSG Diagram]

Specifies which HPR activation and deactivation messages VTAM should issue. Because some of the messages affected by this start option have different versions, the exact message that is affected in the output depends on the value specified on the MSGLEVEL start option. If the current value of the HPRITMSG is not appropriate for your system, you can modify the HPRITMSG value by using the MODIFY VTAMOPTS command.

**Rule:** HPRITMSG is meaningful only if VTAM provides RTP-level HPR support. At a minimum, the NODETYPE start option must also be coded and the RTP parameter must be specified on the HPR start option.

**HPRITMSG=BASE**

**RTP activation when running with MSGLEVEL=BASE|V4R1|V4R2|V4R3|V4R4|CS390**

When an RTP pipe is activated, VTAM issues the following message:

IST1488I ACTIVATION OF RTP puname AS role TO cnetid.cpname

**RTP inactivation when running with MSGLEVEL=BASE**

When an RTP pipe is deactivated, VTAM might issue the following messages:

IST1488I INACTIVATION OF RTP puname AS PASSIVE TO cnetid.cpname
IST619I ID = nodename FAILED -- RECOVERY IN PROGRESS
IST129I UNRECOVERABLE OR FORCED ERROR ON NODE nodename -- VARY INACT SCHED
IST105I nodename NODE NOW INACTIVE
IST871I RESOURCE puname DELETED

**RTP inactivation when running with MSGLEVEL=V4R1**

When an RTP pipe is deactivated, VTAM might issue the following messages:

IST1488I INACTIVATION OF RTP puname AS PASSIVE TO cnetid.cpname
IST619I ID = nodename FAILED -- RECOVERY IN PROGRESS
IST129I UNRECOVERABLE OR FORCED ERROR ON NODE nodename -- VARY INACT SCHED
IST105I nodename NODE NOW INACTIVE
IST1136I VARY INACT puname SCHEDULED - UNRECOVERABLE ERROR
IST1133I puname IS NOW INACTIVE, TYPE = PU_T2.1
IST871I RESOURCE puname DELETED

**RTP inactivation when running with MSGLEVEL=V4R2|V4R3|V4R4|CS390**

When an RTP pipe is deactivated, VTAM might issue the following messages:

IST1488I INACTIVATION OF RTP puname AS PASSIVE TO cnetid.cpname
IST1416I ID = puname FAILED - RECOVERY IN PROGRESS
IST1136I VARY INACT puname SCHEDULED - UNRECOVERABLE ERROR
IST1133I puname IS NOW INACTIVE, TYPE = PU_T2.1
IST871I RESOURCE puname DELETED

**HPRITMSG=ENHANCED**

Chapter 4. VTAM start options 863
RTP activation when running with
MSGLEVEL=BASE|V4R1|V4R2|V4R3|V4R4|CS390

When an RTP pipe is activating, VTAM issues the IST1488I message group. This message group identifies the RTP PU name, whether this is the active or passive end of the pipe, the partner CP name, priority, associated APPN Class of Service, and the APPN route that the pipe is traversing.

IST1488I ACTIVATION OF RTP puname AS role TO cnetid.cpname
[IST1962I APPNCOS = appncos_name- PRIORITY = NETWORK]
[IST1963I APPNCOS = appncos_name- PRIORITY = HIGH]
[IST1964I APPNCOS = appncos_name- PRIORITY = MEDIUM]
[IST1965I APPNCOS = appncos_name- PRIORITY = LOW]
IST1480I RTP END TO END ROUTE - RSCV PATH
IST1460I TGN CPNAME TG TYPE HPR
IST1461I tgn cpname tgyype hpr
.
IST314I END

RTP inactivation when running with
MSGLEVEL=BASE|V4R1|V4R2|V4R3|V4R4|CS390

When an RTP pipe is deactivating, VTAM issues the IST1488I message group. This message group identifies the RTP PU name, whether this is the active or passive end of the pipe, the partner CP name, and the priority and associated APPN Class of Service.

IST1488I INACTIVATION OF RTP puname AS role TO cpnetid.cpname
[IST1962I APPNCOS = appncos_name- PRIORITY = NETWORK]
[IST1963I APPNCOS = appncos_name- PRIORITY = HIGH]
[IST1964I APPNCOS = appncos_name- PRIORITY = MEDIUM]
[IST1965I APPNCOS = appncos_name- PRIORITY = LOW]
IST314I END

Result: If an RTP pipe is being deactivated by a VNET,INACT,ID=rtppuname,F command, message IST105I or message IST1133I will be issued to the console in addition to the new IST1488I message group. This ensures that a response is sent back to the operator who issued the command.

HPRNCBPBF start option
Specifies whether HPR can be used although it may cause session data to travel through an NCP twice. If HPR is used, session data may pass through the NCP as HPR data on the RTP connection AND also as non-HPR data.

**HPRNCPBF=NO**
- Specifies HPR will not be used if session data might travel through an NCP twice.

**HPRNCPBF=YES**
- Specifies that HPR can be used although it may cause session data to travel through an NCP twice.

When coding HPRNCPBF=YES, in general your considerations are possible decrease in network performance but better session availability.

You can change the value of HPRNCPBF with the MODIFY VTAMOPTS command while VTAM is running. If you change the value from YES to NO, no new RTP connections are activated that would cause session data to go through an NCP twice, but existing such RTP connections may in some cases be used.


**HPRPSDLY start option**
Notes:
1. HPRPSDLY is meaningful only if VTAM provides RTP-level HPR support.

Specifies the time that elapses before RTP pipes enter the path switch state.

The ARB flow control algorithm is sensitive to minor variations in the round-trip time and to unpredictable response times from the partner. This sensitivity might cause RTP endpoints to prematurely enter the path switch state. Although this situation does not affect availability, it does consume CPU cycles and also causes a significant number of path switch messages to be written to the console log.

The HPRPSDLY start option affects all RTP pipes with endpoints on this host. This start option affects only path switches that are the result of an unresponsive partner (path switches that generate the message IST1818I PATH SWITCH REASON: SHORT REQUEST RETRY LIMIT EXHAUSTED). The HPRPSDLY start option does not control path switches that are initiated as a result of a TG INOP, F RTP commands, or the PSRETRY function. If you do not code this operand, a \( ps\_delay \) value of 0 is used.

**HPRPSDLY=ps\_delay**

- Specifies the minimum amount of time, in seconds, that local RTP endpoints must delay before entering the path switch state as the result of an unresponsive partner. During this time, the RTP endpoint periodically tries to contact the partner in an effort to avoid switching paths. The value 0 indicates that the RTP nodes will enter the path switch state when a predetermined number of retry attempts have been unsuccessful.

When RTP pipes are directly attached to Enterprise Extender (EE), you can specify the HPR path switch delay value at a more granular level. For predefined EE connections, define this parameter on the PU definition in the switched major node. For EE connection networks, define this parameter on the connection network GROUP definition statements in the EE XCA major node. For dial-in EE connections with associated PUs that are dynamically created, define this parameter on the model major node (DYNTYPE=EE) PU definition statement.

Tips:
- This start option can be useful when the RTP partner is on a node that is CPU-constrained or is running in a virtualized environment. In both of these situations, allowing additional time for the RTP partner to respond might avoid unnecessary processing associated with a path switch.
- If alternate routes exist, specifying a long delay time might cause unnecessary delays for the sessions that are using RTP pipes to this host.

You can change the value of the HPRPSDLY operand with the MODIFY VTAMOPTS command while VTAM is running. The new HPRPSDLY value is used immediately by both new and existing RTP connections, if applicable.

**HPRPSMSGS start option**
Specifies whether VTAM should enable the path switch message reduction and summarization function. This function limits the number of HPR path switch started message groups that can be issued during a path switch event interval (60 seconds). The corresponding path switch completed or failed message group for a given RTP pipe is issued only if the path switch started message group could be issued for that RTP pipe. A path switch event time interval is initiated when any path switch event (started, failed, or completed) occurs, and a path switch event time interval has not already been started.

At the end of the time interval, a summary of the path switch events is issued to the console. The path switch message summary is always issued to the console at the end of the time interval, whether or not messages were suppressed. The summary includes all path switch events that occurred during the specified path switch event time interval, including path switch event information that was issued to the console. If the ps_msg_count value uses the default value ALL or is coded as ALL, the path switch message reduction and summarization function is disabled. When the value ALL is coded or used as the default, all path switch message groups can be issued. If the current value of the HPRPSMSG operand is not appropriate for your system, you can modify the HPRPSMSG value using the MODIFY VTAMOPTS command.

**Rule:** HPRPSMSG is meaningful only if VTAM provides RTP-level HPR support. At a minimum, the NODETYPE start option must also be coded and the RTP parameter must be specified on the HPR start option.

**Results:**
- When you modify the value of the HPRPSMSG start option, the new value goes into effect immediately. If path switches are actively occurring while this start option is dynamically changing, the display output presented to you might not be consistent.
- The path switch message summary is not issued to the console when VTAM is halting.

This start option does not directly control the IST1494I (Completed) or IST1494I (Failed) message groups. If the IST1494I (Started) message group was issued for a specific RTP pipe, then the IST1494I (Completed) or IST1494I (Failed) message group is issued. If the IST1494I (Started) message group was not issued for a specific RTP pipe, then the IST1494I (Completed) or IST1494I (Failed) message group is not issued.

The path switch message reduction function is useful in large installations that have hundreds or thousands of RTP endpoints on z/OS Communications Server. In the event of a large network outage, this function can prevent VTAM from flooding the system console log with redundant path switch messages. Although this function reduces the amount of path switch information that is issued to the console log, the summary information attempts to supply the operator with enough information to understand the size and scope of the outage.

**Tip:** The HPR path switch message reduction function might inhibit some path switch completion messages from being issued. The IST1494I message group supplies the APPN route that the RTP pipe is using after the path switch.
Another way to obtain the APPN route for a specific RTP pipe is to issue the D NET,ID=rtppuname command. For more information about this command, see z/OS Communications Server: SNA Operation.

Restriction: The HPR path switch message reduction function is not associated with the message flooding prevention table. For predictable results, do not code the IST1494I message in the message-flooding prevention table when the HPR path switch message reduction function is enabled. For more information about message-flooding prevention and the IBM-supplied message-flooding prevention table, see z/OS Communications Server: SNA Resource Definition Reference.

Rule: The HPR path switch message reduction and summarization function does not affect HPR path switch messages that are initiated by the MODIFY RTP command. When you issue the MODIFY RTP command, you will see all related path switch messages.

The HPR path switch summary messages provide information for up to fifty partner CPs in ten network IDs.

Path switch messages:

When an RTP pipe enters the path switch state, the following message group is issued to the console log:

IST1494I PATH SWITCH STARTED FOR RTP puname TO cpname
IST1817I PATH SWITCH REASON: RTP CONNECTION UNAVAILABLE
IST1818I PATH SWITCH REASON: SHORT REQUEST RETRY LIMIT EXHAUSTED
IST1819I PATH SWITCH REASON: TG INOP
IST1820I PATH SWITCH REASON: MODIFY RTP COMMAND ISSUED
IST1821I PATH SWITCH REASON: AUTO PATH SWITCH FOR PSRETRY
IST1822I PATH SWITCH REASON: UNKNOWN
IST1937I PATH SWITCH REASON: INITIATED BY REMOTE PARTNER
IST2239I PATH SWITCH REASON: MNPS ENDPOINT RECOVERY
IST2335I PATH SWITCH REASON: XMIT STALL DETECTED
IST314I END

When an RTP pipe successfully completes the path switch process, the following message group is issued to the console log:

IST1494I PATH SWITCH COMPLETED FOR RTP puname TO cpname
IST1480I RTP END TO END ROUTE - RSCV PATH
IST1460I TGN CPNAME TG TYPE HPR
IST1461I tgn cpname tgtype hpr
.
.
.
IST314I END

When an RTP pipe fails to successfully complete a path switch, the following message group is issued to the console:

IST1494I PATH SWITCH FAILED FOR RTP puname TO cpname
IST1495I NO ALTERNATE ROUTE AVAILABLE
IST314I END

Path switch message summary examples:

At the end of an HPR path switch event time interval, an IST2191I message group is issued to the system console. For complete information about the IST2191I message group, see z/OS Communications Server: SNA Messages.

The HPR path switch summary messages provide information for up to fifty partner CPs in ten NETIDs.
The path switch message event time interval is set when the first pipe enters the path switch state. In some cases, when RTP path switches complete quickly, the path switch summary message group might contain information related to the path switch started events and the path switch completion events. This situation might occur when HPRPSMSG=10 is specified and 20 RTP pipes enter and successfully complete the path switch within 60 seconds. Message group IST1494I (Started) is issued to the console for the first ten RTP pipes. Message group IST1494I (Started) is not issued to the console for the next ten RTP pipes. When all the RTP pipes successfully complete the path switch process, the message group IST1494I (Completed) is issued for the first ten RTP pipes (the RTP pipes for which the message group IST1494I (Started) was previously issued). Because all of these path switch events occur within the same 60-second interval, a total of ten IST1494I (Started) and ten IST1494I (Completed) message groups are issued to the console. When the 60-second time interval ends, the IST2191I path switch summary message group, similar to the following example, is issued to the console log.

```
IST2191I HPR PATH SWITCH SUMMARY FROM 03/21/06 AT 09:45:14
IST924I -------------------------------------------------------------
IST2192I STARTED = 20
IST2193I TGINOP = 20 SRQTIMER = 0 PSRETRY = 0
IST2194I PARTNER = 0 MNPS = 0 UNAVAILABLE = 0
IST2195I NETWORK = 5 HIGH = 5 MEDIUM = 5 LOW = 5
IST924I -------------------------------------------------------------
IST2196I COMPLETED = 20
IST2195I NETWORK = 5 HIGH = 5 MEDIUM = 5 LOW = 5
IST924I -------------------------------------------------------------
IST2197I FAILED = 0
IST2195I NETWORK = 0 HIGH = 0 MEDIUM = 0 LOW = 0
IST924I -------------------------------------------------------------
IST2198I NETID STARTED COMPLETED FAILED
IST2199I CPNAME NET HI MED LOW NET HI MED LOW NET HI MED LOW
IST2200I --------- --------------- --------------- ---------------
IST2201I NETA5555 5 5 5 5 0 0 0 0
IST2201I SSCP7A 1 1 1 1 1 1 1 1 0 0 0 0
IST2201I SSCP3A 1 1 1 1 1 1 1 1 0 0 0 0
IST2201I SSCP7B 1 1 1 1 1 1 1 1 0 0 0 0
IST2201I SSCP2AB 1 1 1 1 1 1 1 1 0 0 0 0
IST2201I SSCP99 1 1 1 1 1 1 1 1 0 0 0 0
IST924I -------------------------------------------------------------
IST2206I 40 PATH SWITCH EVENTS FOR 5 CPS IN 1 NETIDS
IST314I END
```

When path switches do not complete quickly, the path switch information associated with a given RTP pipe might be spread across multiple path switch summary message groups. This situation might occur when HPRPSMSG=10 and HPRPST=(4M,4M,2M,2M) are specified, and 20 RTP pipes (five low priority, five medium priority, five high priority, five network priority) enter the path switch state. Message group IST1494I (Started) is issued to the console for the first ten RTP pipes. Message group IST1494I (Started) is not issued to the console for the next ten RTP pipes. In this example, an alternate route is not available. All the RTP pipes remain in the path switch state until their respective HPRPST timers expire. When all the timers have expired, the path switches fail. Because the HPRPST values for the various priorities are specified as different values, the path switches fail at different times. When the current path switch message event interval ends, a summary of twenty path switch started events, which are similar to the following example, is issued to the console.

```
IST2191I HPR PATH SWITCH SUMMARY FROM 03/21/06 AT 09:45:14
IST924I -------------------------------------------------------------
IST2192I STARTED = 20
IST2193I TGINOP = 20 SRQTIMER = 0 PSRETRY = 0
IST2194I PARTNER = 0 MNPS = 0 UNAVAILABLE = 0
IST2195I NETWORK = 5 HIGH = 5 MEDIUM = 5 LOW = 5
IST924I -------------------------------------------------------------
IST2196I COMPLETED = 20
IST2195I NETWORK = 5 HIGH = 5 MEDIUM = 5 LOW = 5
IST924I -------------------------------------------------------------
IST2197I FAILED = 0
IST2195I NETWORK = 0 HIGH = 0 MEDIUM = 0 LOW = 0
IST924I -------------------------------------------------------------
IST2198I NETID STARTED COMPLETED FAILED
IST2199I CPNAME NET HI MED LOW NET HI MED LOW NET HI MED LOW
IST2200I --------- --------------- --------------- ---------------
IST2201I NETA5555 5 5 5 5 0 0 0 0
IST2201I SSCP7A 1 1 1 1 1 1 1 1 0 0 0 0
IST2201I SSCP3A 1 1 1 1 1 1 1 1 0 0 0 0
IST2201I SSCP7B 1 1 1 1 1 1 1 1 0 0 0 0
IST2201I SSCP2AB 1 1 1 1 1 1 1 1 0 0 0 0
IST2201I SSCP99 1 1 1 1 1 1 1 1 0 0 0 0
IST924I -------------------------------------------------------------
IST2206I 40 PATH SWITCH EVENTS FOR 5 CPS IN 1 NETIDS
IST314I END
```
In this example, after 2 minutes elapsed, the five network and five high
priority RTP pipes failed. After the attempts failed, a new 60-second path
switch interval began. The message group IST1494I (Failed) was issued
only for the RTP pipes for which the message group IST1494I (Started) was
issued previously. When the 60-second interval expired, the path switch
summary message group IST2191I was issued:

IST2191I HPR PATH SWITCH SUMMARY FROM 04/27/06 AT 06:23:11
IST924I -----------------------------------------------------------------------------------------------------------
IST2192I STARTED = 0
IST2193I TGINOP = 0 SRQTIMER = 0 PSRETRY = 0
IST2194I PARTNER = 0 MNPS = 0 UNAVAILABLE = 0
IST2195I NETWORK = 5 HIGH = 5 MEDIUM = 5 LOW = 5
IST924I -----------------------------------------------------------------------------------------------------------
IST2196I COMPLETED = 0
IST2197I FAILED = 10
IST924I -----------------------------------------------------------------------------------------------------------
IST2198I CPNAME NET HI MED LOW NET HI MED LOW NET HI MED LOW
IST2199I --------- --------------- --------------- ---------------
IST2200I NETA 0000 0000 0000
IST2201I SSCP3A 0000 0000 0000
IST2201I SSCP7A 0000 0000 0000
IST2201I SSCP99 0000 0000 0000
IST2201I SSCP7B 0000 0000 0000
IST2201I SSCP2AB 0000 0000 0000
IST924I -----------------------------------------------------------------------------------------------------------
IST2206I 10 PATH SWITCH EVENTS FOR 5 CPS IN 1 NETIDS
IST314I END

After 4 minutes elapsed, the final ten RTP pipe path switch attempts failed.
The following information shows a summary display:

IST2191I HPR PATH SWITCH SUMMARY FROM 04/27/06 AT 06:25:11
IST924I -----------------------------------------------------------------------------------------------------------
IST2192I STARTED = 0
IST2193I TGINOP = 0 SRQTIMER = 0 PSRETRY = 0
IST2194I PARTNER = 0 MNPS = 0 UNAVAILABLE = 0
IST2195I NETWORK = 5 HIGH = 5 MEDIUM = 5 LOW = 5
IST924I -----------------------------------------------------------------------------------------------------------
IST2196I COMPLETED = 0
IST2197I FAILED = 10
IST924I -----------------------------------------------------------------------------------------------------------
IST2198I CPNAME NET HI MED LOW NET HI MED LOW NET HI MED LOW
IST2199I --------- --------------- --------------- ---------------
IST2200I NETA 5 5 5 5 0 0 0 0 0 0 0 0
IST2201I SSCP3A 1 1 1 1 0 0 0 0 0 0 0 0
IST2201I SSCP7A 1 1 1 1 0 0 0 0 0 0 0 0
IST2201I SSCP99 1 1 1 1 0 0 0 0 0 0 0 0
IST2201I SSCP7B 1 1 1 1 0 0 0 0 0 0 0 0
IST2201I SSCP2AB 1 1 1 1 0 0 0 0 0 0 0 0
IST924I -----------------------------------------------------------------------------------------------------------
IST2206I 10 PATH SWITCH EVENTS FOR 5 CPS IN 1 NETIDS
IST314I END
HPRPST start option

\[
\text{HPRPST} = (\text{low\_limit}, \text{med\_limit}, \text{high\_limit}, \text{network\_limit})
\]

range: 5 secs - 24 hours (low, medium, high, and network limits)

Specifies the maximum amount of time during which VTAM attempts to perform a path switch before it ends a connection, when HPR=RTP. The value can be expressed as \(nS\) (seconds), \(nM\) (minutes), or \(nH\) (hours). If you do not specify an \(S\), \(M\), or \(H\), seconds are assumed. When VTAM determines that it is necessary to switch paths for the RTP logical connection, it starts a path switch timer.

You can specify different time limits for low, medium, high, and network transmission priority connections. For example, you might allow only a 20-second (20S) limit for interactive connections to remain open during a path switch, but allow an hour (1H) limit for batch connections to remain open while waiting for restoration of a path that is unavailable as a result of maintenance activities.

Although there are no strict rules for coding HPRPST, consider the following conditions:

- An interval that is too short might not allow enough time for a path switch attempt to succeed and find a new route. This could be a problem, particularly when the network is congested.
- An interval that is too long might use up storage and could leave an LU in a state in which other LUs or applications cannot communicate with it.
- Set the network transmission priority to a substantially lower value than the other transmission priorities. This allows CP-CP sessions (which carry signals that flow at network priority) to complete a path switch before the path-switch timer for an RTP connection carrying an LU-LU session expires.
- When specifying the HPRPST value, take into consideration HPRPSDLY, which allows you to set a path switch delay.
  - The HPRPSDLY start option affects all RTP pipes with endpoints on the host where it is specified.
- For RTP pipes that are directly attached to Enterprise Extender, also take into consideration the EE-only path switch delay values that can be specified for individual EE connections with the HPRPSDLY operand. Consider coding the HPRPSDLY operand value EEDELAY for these connections. When you specify the EEDELAY value, and EE is the only path to the RTP partner, specify the HPRPST value to be sufficiently large to enable the EE connection to be re-established. A sufficiently large value enables the RTP pipe and its associated LU-LU sessions to be successfully recovered. The HPRPSDLY operand is available on the following major nodes:
  - For predefined EE connections, define this parameter on the PU definition in the switched major node.
  - For EE connection networks, define this parameter on the connection network GROUP definition statements in the EE XCA major node.
  - For dial-in EE connections that have associated PUs that are dynamically created, define this parameter on the model major node (DYNTYPE=EE) PU definition statement.

- Although HPR connections that are established for some sessions with multinode persistent session (MNPS) applications might have a network transmission priority, these connections use the HPRPST value that is specified for the high transmission priority limit. See z/OS Communications Server: SNA Network Implementation Guide for more information about recovering MNPS sessions.

You can change the value of HPRPST with the MODIFY VTAMOPTS command while VTAM is running. The new HPRPST values do not affect existing RTP connections, but are used for any new RTP connections that are activated.

**HPRSESLM start option**
Rule: HPRSESLM is meaningful only if VTAM provides RTP-level HPR support. At a minimum, the NODETYPE start option must also be coded and the RTP parameter must be specified on the HPR start option.

Use the HPRSESLM start option to limit the number of sessions that VTAM assigns to a single RTP connection (pipe). When the number of sessions assigned to an RTP pipe reaches the specified \textit{sess\_lim} value, new sessions are no longer assigned to that RTP pipe until the number of sessions falls below the specified \textit{sess\_lim} value; instead, another RTP pipe is chosen or created for the new sessions. Limiting the number of sessions on a single RTP pipe can result in significant performance improvement by allowing concurrent traffic on multiple RTP pipes.

\textbf{HPRSESLM=NOLIMIT}

Specifies that VTAM should not enforce a limit on the number of sessions assigned to an RTP pipe.

\textbf{HPRSESLM=sess\_lim}

Specifies that VTAM should limit the number of sessions assigned to an RTP pipe to the number specified by the \textit{sess\_lim} value. The \textit{sess\_lim} value is a decimal integer in the range 1000 - 2147483647.

\textbf{HPRSESLM=DISABLED}

Specifies that VTAM should completely disable the HPRSESLM function to allow this level of VTAM to coexist with older releases of VTAM. HPRSESLM=DISABLED is meaningful only on interchange nodes (ICNs). If you specify HPRSESLM=DISABLED on any other type of APPN node, IST1070I is displayed and HPRSESLM is set to NOLIMIT.

If HPRSESLM is not set to DISABLED, then you can change the value of HPRSESLM (to any valid value except DISABLED) while VTAM is running by issuing the \texttt{MODIFY VTAMOPTS} command. If HPRSESLM is set to DISABLED, then you can only change the HPRSESLM value by stopping and restarting VTAM. Although changing HPRSESLM does not affect any sessions that have already been assigned to an RTP pipe, the changes to HPRSESLM take effect immediately. If you specify HPRSESLM=NOLIMIT, then VTAM does not enforce a session limit for all new and existing RTP pipes when new sessions are established. If you specify HPRSESLM=sess\_lim, then VTAM immediately begins enforcing the specified session limit (for all new and existing RTP pipes) when new sessions are established.

The \textit{sess\_lim} value specified for HPRSESLM does not represent an absolute session limit. The actual number of sessions assigned to an RTP pipe can exceed the specified \textit{sess\_lim} value during normal operation.

- If multiple sessions (that are between the same two APPN nodes and that are requesting the same class of service) are initiated at the same time, then most or all of these sessions can be assigned to the same RTP pipe and result in the...
specified \textit{sess\_lim} value being exceeded. This is because of a time delay between when an RTP pipe is assigned to a session and when the active session count for that RTP pipe is incremented.

- Although VTAM does not assign new sessions to an RTP pipe that has reached or exceeded the specified \textit{sess\_lim} value, the APPN node on the other end of that RTP pipe can continue to add new sessions to that RTP pipe. The node on the other end of an RTP pipe might continue adding new sessions in the following circumstances:
  - The node on the other end of the RTP pipe is not a VTAM node and does not support the HPRSESLM (or equivalent) function.
  - The node on the other end of the RTP pipe is a VTAM running an earlier (pre-V1R10) release of z/OS Communications Server and does not support the HPRSESLM function.
  - The node on the other end of the RTP pipe is a VTAM running z/OS Communications Server V1R10 or later but has specified a larger HPRSESLM value than this VTAM. (For best results, specify the same HPRSESLM value on all of your VTAMs.)
  - The nodes on both ends of the RTP pipe are VTAMs running z/OS Communication Server V1R10 or later and have specified the same HPRSESLM value, but both nodes are adding sessions to the same RTP pipe at approximately the same time.

- The HPRSESLM start option is not enforced for RTP pipes that are created for interchange node (ICN) sessions. ICN sessions (sessions that traverse both APPN and subarea portions of the SNA network through an interchange node) impose special requirements for choosing an RTP pipe that has an ICN as one or both of the endpoints. In some cases, the ICN and the partner RTP node must both choose the RTP pipe that will be used for an ICN session, but they must choose it at different times. If these two nodes do not choose the same RTP pipe for the ICN session, then the ICN session fails with sense code 0897000A when the BIND arrives at the ICN over the wrong RTP pipe. To prevent this type of problem from occurring because the two nodes involved have different \textit{sess\_lim} values specified, the HPRSESLM start option is not enforced for RTP pipes created for ICN sessions.

\textbf{Rule:} Except when HPRSESLM=DISABLED is specified, VTAM ICNs running z/OS Communications Server V1R10 or later always separate ICN session onto their own RTP pipes by using a unique network connection endpoint (NCE) when creating RTP pipes for ICN sessions. This unique ICN NCE prevents pure APPN (non-ICN) sessions from sharing the same RTP pipe as ICN sessions. In a typical mixed APPN and subarea network, there are generally many more pure APPN sessions than ICN sessions. Although VTAM does not enforce the HPRSESLM start option for ICN sessions, separating ICN sessions onto their own RTP pipes prevents ICN sessions from overloading the RTP pipes that were created for pure APPN sessions and minimizes the possibility of placing too much session traffic on RTP pipes that were created for ICN sessions.

\textbf{HPRSTALL start option}
**Rule:** HPRSTALL is meaningful only if VTAM provides RTP-level HPR support. At a minimum, the NODETYPE start option must also be coded and the RTP parameter must be specified on the HPR start option.

Specifies the amount of time that an RTP pipe can remain continuously stalled before VTAM stops it.

An RTP pipe stalls when the partner RTP requests retransmission of at least one network layer packet (NLP), but the request cannot be honored. A stall also occurs when the partner RTP repeatedly requests transmission of the same NLP. When both conditions are in effect simultaneously, each stall is timed separately.

You must specify the time in full seconds, in the range 30 - 3600. If you omit HPRSTALL or specify 0, VTAM does not automatically stop stalled RTP pipes.

You can change the HPRSTALL value with the MODIFY VTAMOPTS command while VTAM is running. For RTP pipes that are already stalled when the option is modified, the time is measured from when the stall began, not from when you issued the command.

**HSRTSIZE start option**
HSRTSIZE = number_of_queue_pointers

range: 1–2097148

Specifies the number of queue pointers in the symbol resolution table (SRT) for the network containing the VTAM host node.

**INITDB start option**

(1)

**Notes:**
1 INITDB is meaningful only if the NODETYPE=NN start option is also used.

Specifies whether the directory services and the topology and routing services databases are loaded when VTAM is started.

**INITDB=ALL**
   Specifies that both databases are loaded.

**INITDB=DIR**
   Specifies that only the directory services database is loaded.

**INITDB=TOPO**
   Specifies that only the topology and routing services database is loaded.

**INITDB=NONE**
   Specifies that neither database is loaded.

**INOPCODE start option**
Three positional parameters must be specified:

- The name of the VTAM module containing inopcode. A value of ALL indicates all modules containing inopcodes. A null value also indicates all modules containing inopcodes.
- The inopcode within modulename to which the dump attribute is assigned. A value of ALL indicates all inopcodes within the module or modules. A null value also indicates all inopcodes within the module or modules. When specifying an InOpCode, always specify three digits by including any leading zeros.
- The dump attribute to be assigned:

  - **INOPCODE=DUMPENABLE**
    Enables the dump attribute for modulename and inopcode. May be abbreviated as DUMPE.

  - **INOPCODE=DUMPDDISABLE**
    Disables the dump attribute for modulename and inopcode. May be abbreviated as DUMPD.

  - **INOPCODE=DUMPDEFAULT**
    Restores the IBM default dump attribute for modulename and inopcode. May be abbreviated as DUMPDEF.

When modulename detects an inoperative condition, it assigns a predefined inopcode. Subsequently, if INOPDUMP is active for the resource on which the condition was detected and the dump attribute for the modulename inopcode is DUMPENABLE (and an INOPDUMP was not previously taken for the resource) an INOPDUMP is forced. The INOPCODE start option (and MODIFY command) provide more granular control of the INOPDUMP function.

**Tip:** The dump attribute keyword DUMPDEFAULT is provided primarily for the MODIFY INOPCODE command.

**Restrictions:**
- When specifying an InOpCode for the second parameter, always specify three digits by including any leading zeros.
- If ALL is specified for modulename, then ALL must be specified for inopcode.

**INOPDUMP start option**
INOPDUMP = OFF

Specifies that dumps are not taken.

INOPDUMP = ON

Specifies that dumps are taken.

**IOINT start option**
**IOINT** start option

\[
\text{IOINT} = \text{number of seconds}
\]

range: \(0-5366000\)

Specifies the number of seconds after which any outstanding response for most request units sent by VTAM is identified with a VTAM message.

Coding IOINT=0 deactivates the function. If you code a value less than 60, IOINT=60 (1 minute) is assumed.

You can change the value of IOINT with the MODIFY VTAMOPTS command or the MODIFY IOPD command while VTAM is running.

**IOMSGLIM** start option

\[
\text{IOMSGLIM} = \text{number of message pairs}
\]

range: \(1-100\)

Limits the number of pairs of IST530I and IST1436I messages that are written for individual subareas by the IOPD facility. If the number of pending I/O operations for a subarea is greater than the value specified, then the IOPD facility writes only one message pair for each type of pending I/O operation for that subarea instead of one message pair per pending operation. The default value ensures that a message pair is written for each pending I/O operation.

**Note**: The default IOMSGLIM start option value is 100. This will cause a summary message, IST9821, to be issued when the number of IST530I message groups exceeds 100, preventing the console or PPO to be flooded.

You can change the value of IOMSGLIM with the MODIFY VTAMOPTS command while VTAM is running.

**IOPURGE** start option
IOPURGE = timeout_value

Specifies an interval after which outstanding I/O requests are purged. The types of outstanding I/O requests that are checked are:

- CDINIT requests
- Direct search list requests
- APPN search requests
- HPR route setup requests
- HPR route test requests

Before this function (or when IOPURGE is set to zero), outstanding I/O requests could remain outstanding indefinitely. For example, when an intermediate host that is used in session routing goes down, it cannot respond to a session establishment request from an LU. The LU then remains hung as it waits indefinitely for a response. However, when an IOPURGE value expires, the outstanding request is canceled (a negative acknowledgement is sent) the LU is freed, and VTAM continues to search for a session establishment for the LU on an alternate path.

It is recommended that you set an IOPURGE value of three minutes or more, or set the value for twice as long as session setup usually takes.

In some circumstances, for example logging on a remote LU by way of a slow modem, session setup might take longer and the IOPURGE should be adjusted upward accordingly. In these circumstances, if you set IOPURGE too low, a session request is canceled before a response can be sent and received, and the session request will have to be repeated.

Setting too low a value could also generate unnecessary searches. For instance, using the above example of the intermediate routing host: the host might be slow to respond, rather than inactive, and setting too low a value for IOPURGE could cancel a request that would have completed and cause VTAM to continue to search for a session establishment.

If you specify a value less than 30 seconds, VTAM uses IOPURGE=30 (seconds).

You can change the value of IOPURGE with the MODIFY VTAMOPTS command or the MODIFY IOPURGE command while VTAM is running.

n or ns
   Specifies the number of seconds that can pass before a response is assumed to be lost.

nM
   Specifies the number of minutes that can pass before a response is assumed to be lost.

nH
   Specifies the number of hours that can pass before a response is assumed to be lost.
nD  Specifies the number of days that can pass before a response is assumed to be lost.

0  Specifies that you do not want to use this function.

**IPADDR start option**

```
IPADDR = IP_address
```

Specifies the IP address at the local TCP/IP stack. If the local TCP/IP stack has multiple static IPv4 VIPA addresses, use IPADDR to specify which of those static VIPA addresses you want other Enterprise Extender nodes to use in order to communicate with this host. Alternatively, the HOSTNAME start option can be coded to provide a host name to be resolved into the static VIPA address. If neither IPADDR nor HOSTNAME are specified, the default is to use the first static IPv4 VIPA address in the TCP/IP stack’s (the referenced TCPNAME start option stack) HOME list.

The IPADDR operand can also be specified on the Enterprise Extender XCA major node GROUP definition statement. Specifying the IPADDR operand on the EE XCA major node GROUP definition statement is necessary in the following cases:

- When your Enterprise Extender configuration requires the use of multiple static VIPAs
- When HOSTNAME was coded as a start option and the value of hostname resolves to an IP address different from the IP address that is to be used for this EE XCA major node GROUP definition statement.

The address must be specified in dotted decimal format for an IPv4 address (for example, IPADDR=4.6.41.7) or in colon-hexadecimal format for an IPv6 address (for example, IPADDR=2000::67:1:2).

**Note:**

1. The IPADDR start option is used only when the IPADDR and HOSTNAME operands are not specified on the XCA major node (MEDIUM=HPRIP) GROUP definition statements, and the HOSTNAME start option was not coded.
2. The IPADDR start option can be dynamically modified using MODIFY VTAMOPTS under certain conditions. An attempt to modify IPADDR to be a nonzero IP address will be failed with message IST448I if a static VIPA address has been obtained from the TCP/IP stack named in the TCPNAME start option and is being used for active Enterprise Extender connections. All active Enterprise Extender lines must be deactivated before the MODIFY VTAMOPTS,IPADDR=IP address will be accepted.

If the MODIFY command is successful, displays of VTAM start options will show the new value immediately; however, the new value will not be used until all Enterprise Extender lines whose GROUP definition statements do not have HOSTNAME or IPADDR explicitly coded are inactive. Any subsequent activation for an Enterprise Extender line whose GROUP definition statement does not have HOSTNAME or IPADDR explicitly coded will make use of the new IPADDR start option value. If the HOSTNAME start option is currently in use when the MODIFY VTAMOPTS is performed to activate IPADDR, the HOSTNAME start option is reset (for example, treated as if MODIFY VTAMOPTS,HOSTNAME=NONE was also coded) as part of processing the MODIFY VTAMOPTS.
A value of 0.0.0.0, or an IPv6 IP address of all zeros, usually written as ::, can be specified for the IPADDR value on the MODIFY VTAMOPTS command as a mechanism for clearing the value of the current IPADDR start option. This option will be failed, however, with message IST448I, if the Enterprise Extender XCA major node is active.

3. HOSTNAME and IPADDR cannot be modified using one MODIFY VTAMOPTS command. If both start options are specified on the same MODIFY command, they will both be ignored and message IST1917I will be generated.

4. If the IPADDR and the HOSTNAME operands are not specified on any of the GROUP definition statements within the Enterprise Extender XCA major node, then either the HOSTNAME, TCPNAME or IPADDR start option must be specified in order to initiate an Enterprise Extender link. If VTAM has been activated without the specification of one of these options then the MODIFY VTAMOPTS command can be used to set one of these options. If an Enterprise Extender link activation is attempted without HOSTNAME, TCPNAME or IPADDR specified then the link activation will fail with the IST1890I message.

5. If either the IPADDR or the HOSTNAME operand is specified on any GROUP definition statement within the Enterprise Extender (XCA) definition, it must be specified on all Enterprise Extender GROUP definition statements (or defaulted from the VTAM start options).

6. The TCPNAME start option is required in a CINET environment if any of the following conditions exist:
   - The IPADDR parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
   - The HOSTNAME parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
   - The HOSTNAME keyword is specified as a VTAM start option.

   If OMVS is not initialized at Enterprise Extender line activation, a Common INET environment is assumed. In this case, if an Enterprise Extender link activation is attempted without TCPNAME specified, then the link activation will fail with message IST1897I.

7. If both the IPADDR and the HOSTNAME start options are specified on the VTAM START command, the value of the IPADDR start option is ignored.

8. IPADDR is meaningful only if the NODETYPE start option is also used.

**IPINFO start option**
Specifies whether VTAM should externalize information from the TCP/IP Information Control Vector (CV X'64') associated with applications, logical units, and cross-domain resources that represent TN3270 clients. The TCP/IP Information Control Vector may have been received from the TN3270 client during resource activation (for TN3270 clients that are owned by this host) or on cross-domain session establishment flows (for TN3270 clients that are owned by other hosts). The IPINFO start option setting applies to all TCP/IP information received by VTAM, whether it is received from a locally owned resource or a cross-domain resource.

For TN3270 clients that are owned by this host, the associated TCP/IP information, if saved, is maintained as long as the resource remains session capable. For TN3270 clients that are owned by other hosts, the associated TCP/IP information, if saved, is maintained as long as one or more sessions exist to the TN3270 client.

You can dynamically change the value of IPINFO using the MODIFY VTAMOPTS command.

**IPINFO=NONE**

 Specifies that TCP/IP information should not be maintained by VTAM at all. Therefore, it cannot be displayed or provided to other products, exits, or nodes.

**IPINFO=DISPONLY**

 Specifies that TCP/IP information should be maintained by VTAM and used for VTAM DISPLAY purposes. The TCP/IP information will not be sent to exits or to CMIP; will not be available to the USS component for substitution into USS messages; and will not be sent to cross-domain hosts.

**IPINFO=SNDLOCAL**

 Specifies that TCP/IP information should be maintained by VTAM and used for VTAM DISPLAY purposes. Additionally, VTAM will also make this TCP/IP information available to the USS component for substitution into USS messages, the Session Management Exit, the TIVMON Exit, the Common Management Information Protocol (CMIP), and locally owned applications using the LOGON exit.

**IPINFO=SENDALL**

 Specifies that TCP/IP information should be maintained by VTAM and used for VTAM DISPLAY purposes. VTAM will make this TCP/IP information available to the USS component for substitution into USS Messages, the Sessions Management Exit, the TIVMON exit, the Common Management Information Protocol (CMIP), and locally owned applications using the LOGON exit. VTAM will also include TCP/IP information on cross-domain session establishment flows involving TN3270 clients, so that other nodes can make use of this information.

**IQDCHPID start option**
IQDCHPID = chpid specifies the (hexadecimal) CHPID (Channel Path ID) that is to be used for HiperSockets communications from this LP (Logical Partition) to other LPs within this central processor complex (CPC) for TCP/IP Dynamic XCF support. This start option allows the user to designate or reserve a specific IQD CHPID for use by dynamic XCF HiperSockets. This option does not affect user defined HiperSockets devices. HiperSockets is used for TCP/IP Dynamic XCF support. All instances of TCP/IP started within this z/OS Communications Server LP will use this same CHPID Dynamic XCF HiperSockets device. VTAM will dynamically construct an HiperSockets TRLE using the HiperSockets subchannel devices associated with this CHPID. The dynamic TRLE created will have a name of IUTIQDIO.

A TRLE that has the reserved name IUTIQDIO indicates it is the HiperSockets device that is built and used exclusively for dynamic XCF support. The PORTNAME will have a name of IUTIQDxx where xx = the IQD CHPID selected by VTAM. All LPs within this CPC using z/OS CS HiperSockets support that specify the same CHPID will gain internal connectivity by the way of dynamic XCF using HiperSockets. If the CPC is enabled to use multiple channel subsystems and the LPs are not using the same CSS channel subsystem, then the IQD CHPID must be defined (in HCD) to span the applicable CSSs. The CHPID is specified as a single hexadecimal byte or 2 digits (for example, F2). The IQDCHPID start option can also be modified. To understand the effects of modifications to IQDCHPID on an active system, see the IQDCHPID option of the MODIFY VTAMOPTS command in [z/OS Communications Server: SNA Operation]. This start option is valid only if the associated processor hardware support is available. When the processor support is not available, this start option is ignored and cannot be modified or displayed.

The three possible definitions are:

- IQDCHPID = ANY is the default and indicates that any HiperSockets CHPID can be used for the Dynamic XCF HiperSockets device. If ANY is specified or defaulted, then:
  - When the first HiperSockets device is activated, VTAM will use the first acceptable HiperSockets CHPID detected (having at least 3 subchannel devices associated with the CHPID) to dynamically build the IUTIQDIO TRLE.
  - Within HCD (or IOCDS), only one HiperSockets CHPID must be defined for this Logical Partition. If VTAM finds more than one HiperSockets CHPID, the HiperSockets activation will not be allowed (because it is considered an ambiguous definition). Users who require (and therefore define in HCD/IOCDS) a single HiperSockets CHPID for a given Logical Partition can use the default of ANY.

- IQDCHPID = NONE will prevent VTAM from activating the IUTIQDIO TRLE which is associated with Dynamic XCF HiperSockets device. The NONE option does not affect (block) the use of user defined HiperSockets devices. All subsequent attempts to activate HiperSockets from any TCP/IP stack within this LP will be prevented (rejected).
• IQDCHPID = chpid is the (hexadecimal) value for a specific HiperSockets (IQD) CHPID which is to be used to build the IUTIQDIO TRLE used only for Dynamic XCF. If the CHPID is specified, it must match one of the IQD CHPIDs that was also defined using HCD (or IOCDS). This LP must be eligible to connect to this specific CHPID and must have a minimum of three subchannel devices associated with this CHPID. The value must be specified as a single-byte hexadecimal number (for example, IQDCHPID=F2). This option is intended for users who must define multiple HiperSockets CHPIDs within HCD [for example, users who plan to use both dynamically defined HiperSockets devices (Dynamic XCF) and user defined HiperSockets devices].

Note:
1. In order to use HiperSockets communications, this processor must have the necessary hardware support. This start option cannot be specified, modified, or displayed on a processor without the hardware support.
2. HiperSockets activation requires a minimum of three subchannel devices and uses a maximum of ten subchannel devices. The HiperSockets subchannel devices associated with the selected iQD CHPID will be grouped together into a single MPC group (two read/write control devices and up to eight data devices). The subchannel devices must be online.

IQDIOSTG start option

Notes:
1. This option only affects iQDIO devices that use a MFS of 64k. The smaller frame sizes will always use 126 SBALs.

Specifies how much storage VTAM keeps available for read processing for all HiperSockets data devices that use a MFS (Maximum Frame Size) of 64k. The HiperSockets MFS is defined in HCD. The HiperSockets storage units are defined in QDIO SBALs (QDIO read buffers). Each SBAL is 64k. For most users, the default setting will be the most suitable option. The storage used for this read processing is allocated from CSM data space 4k pool, and is fixed storage.

HiperSockets devices that are defined with a smaller MFS (16k, 24k, or 40k) are not affected by this start option. Those devices will use 126 SBALs.

The recommended values can be configured by specifying MAX, AVG, or MIN, which are predefined constants (number of SBALs) that are most appropriate for this type of adapter. MAX allows for the best performance (for example, throughput), but requires more storage. Conversely, MIN could be used for devices with lighter workloads or where system storage might be constrained. Also, instead of using one of the 3 predefined values, users can instead define a specific value for IQDIOSTG (the exact number of SBALs), such as IQDIOSTG=100. The supported range is 8 (minimum supported value) to 126 (maximum supported value). The predefined and possible user defined values for IQDIOSTG are:
IQDIOSTG=MAX
   Specifies the recommended maximum (126 SBALs for a total of 7.8 MB). This is
   the default setting.

IQDIOSTG=AVG
   Specifies the recommended average (96 SBALs for a total of 6 MB).

IQDIOSTG=MIN
   Specifies the recommended minimum (64 SBALs for a total of 4 MB).

IQDIOSTG=nnn
   Specifies the user defined specific number of SBALs, valid range of 8 - 126 (see
   Note 1).

Note:
1. If you elect to define a specific value versus the recommended values, carefully
   evaluate your system storage and performance needs for your HiperSockets
   devices. The amount of storage used is (NNN x 64k) x number of active iQDIO
   data devices.

2. You can use VTAM tuning stats to evaluate your needs and usage. Under a
   sample (typical) workload, the NOREADS counter should remain low (close to
   0). If this count does not remain low you might need to consider a higher
   setting for IQDIOSTG. RMF™ can also be used to evaluate the correct setting
   for your environment. RMF records send failures, which can be an indication
   that the target LP (logical partition) does not have enough storage (read
   SBALs).

3. You can override the IQDIOSTG value for a given HiperSockets device by
   using the READSTORAGE parameter on the IPAQIDIO LINK, the IPAQIDIO
   INTERFACE, or the IPAQIDIO6 INTERFACE statement in the TCP/IP profile.

IRNSTRGE start option
IRNSTRGE = 0
IRNSTRGE = size

range: 1–2147483647 bytes

Defines the maximum size of the virtual area in VTAM storage that can save host intermediate routing node (IRN) transmissions. If zero (the default) is used, the amount of storage is unlimited.

### ISTCOSDF start option

ISTCOSDF = INDLU
ISTCOSDF = APPL
ISTCOSDF = DEPLU
ISTCOSDF = INDLU
ISTCOSDF = NONE

Specifies the resource types that can use the ISTCOSDF logmode entry. This entry is used when the logmode name specified for the session is not found.

You can change the value of ISTCOSDF with the MODIFY VTAMOPTS command while VTAM is running.

- **ISTCOSDF=ALL**: Indicates that the ISTCOSDF logmode entry will be used by the following resources:
  - Application programs
  - Dependent LUs
  - Independent LUs

- **ISTCOSDF=APPL**: Indicates that ISTCOSDF is restricted to use by application programs.

- **ISTCOSDF=DEPLU**: Indicates that ISTCOSDF is restricted to use by dependent LUs.

- **ISTCOSDF=INDLU**: Indicates that ISTCOSDF is restricted to use by independent LUs.

- **ISTCOSDF=NONE**: Indicates that the function is disabled.

### LIMINTCP start option
LIMINTCP—number_of_seconds

dependencies:                      LIMRES=YES
range:                             0–65535

Specifies the interval to retain a free SNASVCMG session on limited resources for a control point. The control point SNASVCMG session is used for some network management flows. LIMRES=YES must be coded on applicable GROUP, LINE, or PU definition statements for LIMINTCP to be effective.

If you do not code LIMINTCP, the limited resource function is not supported for network management flows. This function is supported only on network management flows using the CP SNASVCMG session. It is not supported on APPN CP sessions or dependent LU server sessions. Note that in a subarea environment, CP refers to the SSCP.

You can change the value of LIMINTCP with the MODIFY VTAMOPTS command while VTAM is running. However, though you can modify the value of LIMINTCP, there is not a way to dynamically turn off its function after VTAM has started.

**LIST start option**

\[-LIST=\text{start\_option\_list\_id}\]

(1)

**Notes:**

1. LIST can be entered by a VTAM operator only. If LIST is coded in an ATCSTRxx file, it is considered to be an error and is ignored.

Specifies which list of predefined start options is used to start VTAM. Each ATCSTRxx list is filed in the VTAM definition library. You can choose any two alphanumeric characters or national characters (#,@,&) for the value of xx to identify user-defined lists of start options.

For more information about start option lists, see z/OS Communications Server: SNA Network Implementation Guide

**LISTBKUP start option**
LISTBKUP = start_option_list_id

Notes:
1. LISTBKUP can be coded in a start option file only. If you enter it on the
   START command or at an operator prompt, VTAM will ignore it.

Specifies what action is taken when VTAM detects an error in a start file.

LISTBKUP=start_option_list_id
   Specifies the 2-letter start list identifier (ATCSTRyy) of another start file to
   process in place of the file in error. All options in the start file in error are set
   to the values before the file was processed and ATCSTRyy is processed.

   If the backup file contains errors, LISTBKUP=yy is ignored to prevent possible
   looping. However, LISTBKUP=DEFAULTS or LISTBKUP=PROMPT is
   processed in a backup file.

LISTBKUP=DEFAULTS
   Causes the options in the start file in error to be set to the values they had
   before the file was processed. VTAM then continues processing.

LISTBKUP=PROMPT
   Causes all valid options in the start file to be set. The operator is prompted to
   enter overrides to the options that are not valid.

LSIRFMSG start option
Controls the display of IST663I, IST664I, and IST889I messages when an APPN locate search fails to find the destination LU. The ESIRFMSG and FSIRFMSG start options can be used in conjunction with LSIRFMSG to generate additional messages as part of this message group.

- ESIRFMSG controls whether IST891I, IST892I, and IST893I are displayed.

LSIRFMSG is meaningful only if the NODETYPE=NN start option is also used.

You can change the value of LSIRFMSG with the MODIFY VTAMOPTS command while VTAM is running.

**LSIRFMSG=ALLNNS**

Specifies that APPN locate search failure messages are always to be issued by this network node regardless of node role.

**LSIRFMSG=OLUNNS**

Specifies that APPN locate search failure messages are to be issued by this network node, only when acting as the network node server of the OLU or when acting as the origin directory server of the search.

**LSIRFMSG=NONE**

Specifies that APPN locate search failure messages are not to be issued by this network node.

**Guideline:** Because of the volume of messages that can be generated, it is not recommended that this option be enabled during normal operation. Instead, it is recommended that this option be enabled (using the MODIFY VTAMOPTS command) on all necessary hosts only when trying to diagnose specific problems. Once the problem has been diagnosed or documentation has been collected, this option should be disabled once again (using the MODIFY VTAMOPTS command).

**Tip:** When ESIRFMSG or FSIRFMSG start options are used in conjunction with LSIRFMSG, then ALLSSCP is equivalent in function to ALLNNS and OLUSSCP is equivalent in function to OLUNNS.

**MAINTLVL start option**
MAINTLVL = maintenance_level

Specifies which maintenance level is running on a host. You can code up to eight characters to identify the maintenance level, and the identifier can be displayed by an operator.

If you do not specify a maintenance level identifier, blanks are displayed. Any characters except the equals sign (=), comma (,), and left and right parentheses are allowed.

MAXEETST start option

MAXEETST = 500

MAXEETST = max_ee_connectivity_tests

range: 1-1000

Specifies the maximum number of Enterprise Extender connectivity tests (DNET,EEDIAG,TEST=YES commands) that are allowed to run concurrently. When this maximum is reached, all subsequent EE connectivity tests are rejected until some of the previous test commands complete processing. If the current value of the MAXEETST is not appropriate for your system, you can modify the MAXEETST value by using the MODIFY VTAMOPTS command.

MAXHNRES start option
MAXHNRES start option

Specifies the maximum number of host name resolution requests that can run concurrently. This start option controls host name resolution requests for the following functions:

- EE line activation
- Dialing an EE switched PU
- Host name filters specified on D NET,EE commands
- Host name filters specified on D NET,EEDIAG commands

The MAXHNRES start option is closely related to the DLRTCB start option. The MAXHNRES start option controls the number of subtasks that VTAM allows for Enterprise Extender host name resolution requests. The MAXHNRES start option value is internally added to the value of the DLRTCB start option value, but the resulting value cannot exceed the DLRTCB upper limit. This enables VTAM to govern the number of TCBs used to perform EE host name resolutions.

Restriction: The DLRTCB and MAXHNRES start options cause control blocks to be built below the 16-megabyte line. Therefore, the availability of storage might limit the value that you can code, and conversely, the value you code might impact the availability of the storage.

Results:

- When the total number of resolver requests for D NET,EE and D NET,EEDIAG commands reaches the MAXHNRES value, all subsequent D NET,EE and D NET,EEDIAG commands that specify host name filters are rejected until some of the previous resolver requests complete.
- The number of concurrent requests allowed for EE display command processing is limited to 80% of the MAXHNRES value. This ensures that 20% of the task control blocks remain available for line activation or dial processing.
- When a D NET,EE or D NET,EEDIAG command is entered with both a local and remote host name filter, VTAM initiates two host name resolution requests. Therefore, one display command can account for two host name resolutions that run concurrently.

MAXLOCAT start option
Notes:

1  MAXLOCAT is meaningful only if NODETYPE is specified.

   range: 100–65535

Specifies the maximum locate congestion threshold value to be reached before VTAM suspends sending APPN search requests to an adjacent CP. The maximum locate congestion threshold is determined by the following formula:

\[(\text{queued requests plus queued replies for the adjacent CP}) \times (\text{number of seconds since last response from adjacent CP})\]

VTAM resumes sending APPN search requests to an adjacent CP when a minimum congestion threshold is reached. The minimum threshold is based on the number of queued searches waiting to be sent to the adjacent CP. When the queued searches are reduced to 10% of the number on the queue when the maximum locate congestion threshold was reached, new APPN search requests are again sent to the adjacent CP.

Even after VTAM suspends sending APPN search requests to an adjacent CP, VTAM continues to queue APPN search replies to the adjacent CP. If the number of queued requests plus the number of queued replies reaches 200% of the MAXLOCAT value and a minimum time interval has been exceeded, VTAM stops the CP-CP session to the adjacent CP. VTAM stops the CP-CP session with a sense code that can be tried again and retries the CP-CP session. The minimum time interval is based on the time since VTAM issued message IST1601I, which indicates that routing to this node has been suspended. If the IOPURGE value is less than the default of 180 seconds, then the minimum time interval is 180 seconds. Otherwise, the minimum time interval is the current IOPURGE value.

If you specify a value less than 100, VTAM uses MAXLOCAT=100 as the MAXLOCAT value.

**MAXLURU start option**
MAXLURU = 6144

ru_length

range: 1–65535 bytes

Specifies the maximum length, in bytes, for request units (RUs) in LU-LU sessions that have a VTAM application as an endpoint.

You can change the value of MAXLURU with the MODIFY VTAMOPTS command while VTAM is running.

MAXSSCPS start option

MAXSSCPS = 10

number_of_SSCPs

range: 1–255

Specifies the maximum number of SSCPs on a specific path that are searched for the destination SSCP before alternate paths are tried. This number affects the rerouting of both session initiation requests (CDINIT) and direct search routing requests (INQUIRE APPSTAT).

You can change the value of MAXSSCPS with the MODIFY VTAMOPTS command while VTAM is running.

MAXSUBA start option
MAXSUBA = 15

MAXSUBA = maximum_subarea_number

dependencies: required if ENA not supported
range: 1–255

Specifies the highest subarea value that can be assigned to any node within this network for communicating with NCPs and SSCP s that do not support extended network addressing (ENA).

This operand is required if any host or NCP in the network does not support ENA.

If you code MAXSUBA, the value must be the same for all nodes (including NCP) in the same network.

Note: If you have coded a node to function solely as an end node or network node (NODETYPE=EN or NN), then MAXSUBA should not be coded because it has no effect. If you code it, error messages will be issued, but the node will come up.

The MAXSUBA value is rounded up to the next higher 2^n-1. For example, if MAXSUBA is coded as 10, the next higher 2^n-1 is 15. Therefore, VTAM uses a MAXSUBA value of 15.

In networks that have no pre-ENA nodes, or where you do not want to communicate with any such nodes, you do not need to code MAXSUBA.


**MIHTMOUT start option**
**MIHTMOUT start option**

Specifies the time (in tenths of a second) that allows you to adjust the missing interrupt handling interval during channel-to-channel communication between two VTAMs or between VTAM and an IBM 3172 Interconnect Controller.

**Note:** This start option does not apply to attachments between two hosts that use the multipath channel (MPC) feature. See “Multipath channel support” on page 85 and in the [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com) for information on MPC.

You can change the value of MIHTMOUT with the MODIFY VTAMOPTS command while VTAM is running.

**MPCACT start option**

Specifies whether activations of MPC subchannel groups are to be suspended if the minimum number of read and write subchannels is not available, either because they are offline or because no valid path exists to the connecting host. VTAM automatically resumes activation when the minimum number of subchannels becomes available. At least one read device and one write device are needed to activate an MPC group.

**MPCACT=WAIT**

Specifies that VTAM is to suspend activations of MPC subchannel groups if the minimum number of read and write subchannels is not available, either because they are offline or because no valid path exists to the connecting host. VTAM automatically resumes activation when the minimum number of subchannels becomes available.

**MPCACT=NOWAIT**

Specifies that VTAM is to fail activations of MPC subchannel groups if the minimum number of read and write subchannels is not available. The system operator must manually try these activations again after the minimum number of subchannels becomes available.

**MSGLEVEL start option**
Specifies the version of the messages to use for those messages listed in z/OS Communications Server: SNA Messages. If the MSGVL operand is specified on the USSMSG macroinstruction, it overrides the value specified on the MSGLEVEL start option.

**Note:** You can abbreviate the MSGLEVEL start option to MSGLVL.

You can change the value of MSGLEVEL with the MODIFY VTAMOPTS command while VTAM is running.

**MSGLEVEL=BASE**

Specifies that for all messages, except those that are coded otherwise in the USS table, VTAM issues the pre-V4R1 message that this message number replaces.

See z/OS Communications Server: SNA Messages for a list of messages replaced by a new message or a message group.

If you specify MSGLEVEL=BASE and there are duplicate names in your SNA interconnected networks, you might not know which resource the message applies to because the message does not use a network-qualified name.

**MSGLEVEL=V4R1**

Specifies that VTAM issues the V4R1 message or messages, except for those that are coded otherwise in the USS table.

See z/OS Communications Server: SNA Messages for a list of messages replaced by a V4R1 message or message group.

**MSGLEVEL=V4R2**

Specifies that VTAM issues the V4R2 message or messages, except for those that are coded otherwise in the USS table.

See z/OS Communications Server: SNA Messages for a list of messages replaced by a V4R2 message or message group.

**MSGLEVEL=V4R3**

Specifies that VTAM issues the V4R3 message or messages, except for those that are coded otherwise in the USS table.

See z/OS Communications Server: SNA Messages for a list of messages replaced by a V4R3 message or message group.

**MSGLEVEL=V4R4**

Specifies that VTAM issues the V4R4 message or messages, except for those that are coded otherwise in the USS table.

See z/OS Communications Server: SNA Messages for a list of messages replaced by a V4R4 message or message group.

**MSGLEVEL=CS390**

Specifies that VTAM issues the z/OS Communications Server message or messages, except for those that are coded otherwise in the USS table.
MSGMOD start option

Specifies whether VTAM should identify the issuing VTAM module in a message.

You can change the value of MSGMOD with the MODIFY VTAMOPTS command or the MODIFY MSGMOD command while VTAM is running.

**MSGMOD=NO**
- Specifies that VTAM does not identify the issuing VTAM module in a VTAM message.

**MSGMOD=YES**
- Specifies that VTAM inserts the last five characters of the name of the VTAM module that issued a VTAM message into each VTAM message following the message identifier.

**Note:** Messages might be truncated if you specify this option.

MULTIPATH start option
Notes:
1 MULTPATH is meaningful only if the NODETYPE start option is also specified.

You can configure TCP/IP to use multiple paths for outbound packets. Use the MULTPATH start option to enable or disable this function for Enterprise Extender (EE). You can change the value of MULTPATH by issuing the MODIFY VTAMOPTS command while VTAM is running.

MULTPATH=NO
Disables multipath routing support for EE connections regardless of the TCP/IP profile IPCONFIG MULTIPATH or IPCONFIG6 MULTIPATH setting in a TCP/IP stack.

MULTPATH=TCPVALUE
Indicates that VTAM accepts the setting of the TCP/IP stack that is associated with EE to control multipath routing for EE connections.

The setting of the TCP/IP profile IPCONFIG MULTIPATH or IPCONFIG6 MULTIPATH determines the multipath routing control in a TCP/IP stack.

**MXSAWBUF start option**
MXSAWBUF = number_of_buffers

range: 1–2147483647

Specifies the maximum number of buffers used for session awareness (SAW).

**MXSSCPRU start option**

MXSSCPRU = ru_length

range: 1–65535 bytes

Specifies the maximum length for RUs in SSCP sessions (SSCP-PU, SSCP-LU, and SSCP-SSCP).

**MXSUBNUM start option**

Specifies the maximum subarea number supported by a network to which a host VTAM is interconnected. If the value coded is not one of the values listed above, VTAM will not accept this start option for processing.

**Note:** The value specified on the HOSTSA start option cannot exceed the value specified for MXSUBNUM.

**NACPROBE start option**
Specifies whether VTAM takes an FFST probe dump when VTAM does not complete the CLOSE ACB processing for an application in a timely manner. The title of the dump is “NACCT -- CLOSE TIMER EXPIRED FOR applname”. Alternatively, you can use this option to specify the time interval at which VTAM takes an FFST probe dump when VTAM does not complete the CLOSE ACB processing for another application in a timely manner.

VTAM must complete the CLOSE ACB processing in a timely manner; otherwise, VTAM takes an FFST probe dump to notify the user and forces the CLOSE ACB processing to complete.

Issue the MODIFY VTAMOPTS command to change the NACPROBE value when VTAM is running. The MODIFY VTAMOPTS command changes the value immediately.

Issue the DISPLAY VTAMOPTS command to display the NACPROBE value.

**NACPROBE=NODUMP**

Specifies that VTAM does not take an FFST probe dump when VTAM does not complete the CLOSE ACB processing for an application in a timely manner.

**NACPROBE=DUMP**

Specifies that VTAM takes an FFST probe dump when VTAM does not complete the CLOSE ACB processing for an application in a timely manner. If the CLOSE ACB processing for another application does not complete in a timely manner, VTAM does not take another FFST probe dump if the previous FFST probe dump was taken less than 30 minutes ago.

**NACPROBE=Number_of_minutes**

Specifies that VTAM takes an FFST probe dump when VTAM does not complete the CLOSE ACB processing for an application in a timely manner only if the interval specified by the Number_of_minutes value has been exceeded since a previous FFST probe dump was taken for another application.

Valid values for Number_of_minutes are in the range 1 - 1440 minutes (1440 minutes specifies 1 day).

**Note:** If VTAM does not take an FFST probe dump because of the NACPROBE start option value, VTAM forces the CLOSE ACB processing to complete and issues the following message:

IST2364I CLOSE ACB TIMER EXPIRED FOR applname

**NCPBUFSZ start option**

Chapter 4. VTAM start options  901
Specifies the request unit size in bytes (I/O buffer size minus the size of the SNA header) used by VTAM when loading or performing a static dump of a remote NCP.

**Note:**
1. This value does not apply to a MODIFY LOAD command. The length of the IPLTEXT RU is always 2048 plus the length of the SNA header.
2. This value applies only to static dumps or loads of 3720, 3725, and 3745 communication controllers. VTAM ignores this value when loading or dumping a 3705, and uses a data size of 512 bytes.

**NETID start option**

Specifies the required 1–8 character identifier of the network containing the host.

**Note:** The use of the national character, #, is discouraged, because this character might not be present on keyboards of terminals produced in some countries. Because # might not be available on all terminals, SNA architecture specifically excludes # from the list of valid characters that can be used for defining network names (NETIDs). Although VTAM allows you to use #, other products might enforce this restriction.

NETID should be unique to each network within a set of SNA interconnected networks. Within each network, each SSCP must have the same NETID. Besides NETID, a gateway configuration requires GWSSCP=YES and an SSCPNAME.

**NMVTLOG start option**
Specifies when VTAM is to record network management vector transport (NMVT) records in LOGREC.

**NMVTLOG=ALWAYS**
Specifies that VTAM always records NMVT alerts in LOGREC.

**NMVTLOG=NEVER**
Specifies that VTAM never records NMVT alerts in LOGREC.

**NMVTLOG=NPDA**
Specifies that VTAM records NMVT alerts in LOGREC only when the NetView hardware monitor, NPDA, is not active.

### NNSPREF start option

Specifies whether or not network node server dynamic switching is enabled for this end node.

**Note:** NNSPREF is valid only if the NODETYPE=EN start option is also used.

**NNSPREF=NONE**
Specifies that dynamic switching of the network node server will not be enabled.

If NNSPREF=NONE is specified through MODIFY VTAMOPTS, dynamic switching will be disabled. Existing CP-CP sessions with the current NNS will not be affected.

**NNSPREF=network_node_server**
Specifies the end node’s preferred network node server. The name specified can be fully qualified with a network ID. If the network node server specified is not qualified with a network ID, the network ID of the end node will be used as the default. By specifying this start option with the name of a preferred network node server, the EN will dynamically switch CP-CP sessions from its current network node server to the preferred network node server when a CP-capable physical link is activated with the preferred network node server.

In addition, when CP-capable links specified in the VTAM configuration list are automatically activated at VTAM startup, only the link to the preferred network node server (if one exists) will trigger CP-CP session activation. If CP-CP sessions to the preferred network node server have not been initiated by the end of VTAM initialization, the network node server (NETSRVR) list will then be used to select a network node server.

This start option is modifiable using the MODIFY VTAMOPTS command. The network node specified for NNSPREF on the MODIFY VTAMOPTS command does not have to be predefined in the active network node server list and there
does not have to be a nameless entry to handle it in the absence of an explicit entry. The value specified on the MODIFY command effectively gets dynamically added to the active NETSRVR list. This entry becomes the preferred entry and will have a higher priority than any other entry in the current NETSRVR list. The switching to the new preferred network node server will take place automatically if there is already an active CP-capable physical link to the new server.

Note: The switching could cause new session requests for resources on the EN to fail until the switching is complete.

**NODELST start option**

```
[ NODELST = name ]
```

Provides the 3–8 character name of the configuration restart file in which VTAM maintains a list of all currently active major nodes.

**NODETYPE start option**

```
[ NODETYPE = EN | NR ]
```

Notes:

1. NODETYPE enables APPN function. The combination of HOSTSA, NODETYPE, and SACONNS determines the configuration (subarea node, interchange node, migration data host, network node, or end node).

Possible node configurations and their functional ability are summarized in Table 70 on page 859.

Specifies the node type that is presented to other APPN nodes. When NODETYPE is not specified, the node operates as a subarea node and all type 2.1 connections are managed as LEN connections. Also, the new start options, definition statements, and operands pertinent to APPN function are ignored.

When NODETYPE is specified, the HOSTSA and/or SACONNS start options determine whether this VTAM will also function as a subarea node. If HOSTSA= is coded and SACONNS= is not coded or SACONNS=YES is coded, then this VTAM will provide subarea function. If HOSTSA= is not coded, or HOSTSA= is coded and SACONNS=NO is coded, then this VTAM will not provide subarea function.

If you want a pure APPN node, then you should not code the HOSTSA start option or you must code SACONNS=NO. Additionally, the DYNASSCP, GWSSCP, and MAXSUBA start options should not be coded because they have no effect. If you code them, the node will come up, but you will get error messages.

**NODETYPE=EN**

Specifies that the node is an end node. When NODETYPE=EN and subarea function has been enabled (see above), the node operates as a migration data host. If subarea function has not been enabled, the node operates as a pure end
node. When the node is configured as a migration data host or end node, attempts to activate an NCP will fail. NCPs can be contacted, however.

**NODETYPE=NN**

Specifies that the node is a network node. When NODETYPE=NN and subarea function has been enabled (see above), the node operates as an interchange node. If subarea function has not been enabled, the node operates as a pure network node. Interchange nodes can own NCPs and their resources.

**NOTRACE start option**

For a description of the NOTRACE start option, see “TRACE for MODULE, STATE (with OPTION), or VTAM internal trace” on page 955 and “TRACE for buffer, I/O, NCP line, QDIOSYNC, SIT, SMS, or STATE (with ID) traces” on page 947.

**NQNMODE start option**

```
NQNMODE -- NAME
```

Specifies whether VTAM defines cross-network resources by both their non-network-qualified names and their network-qualified names or only by their network-qualified names.

If NQNMODE is specified on the CDRSC definition statement, it overrides the value specified on the NQNMODE start option. Dynamic CDRSCs and independent LUs which are defined using LOCADDR=0 are defined with the NQNMODE start option.

You can change the value of NQNMODE with the MODIFY VTAMOPTS command while VTAM is running.

**NQNMODE=NAME**

Specifies that the cross-network resource is defined by both its network-qualified name and its non-network-qualified name. The same-network CDRSC is always considered to be defined using NQNMODE=NAME because it is always identified by its non-network-qualified name. This is the mode in which pre-V4R1 releases of VTAM operate.

If a CDRSC is defined using NQNMODE=NAME (either by definition or start option), then no other resource can be defined to that host by the same non-network-qualified name.

**NQNMODE=NQNAME**

Specifies that the cross-network resource is defined by its network-qualified name and is not defined to this host by its non-network-qualified name.

If you have duplicate names in your SNA interconnected networks and prefer not to use other forms of alias name translation, you can specify (either by definition or start option) NQNMODE=NQNAME.

**NSRTSIZE start option**
NSRTSIZE = (netid, size)

range (size): 1–2097148

Specifies the number of queue pointers in the symbol resolution table (SRT) for the network identifier or identifiers you specify. VTAM uses the value when it allocates an SRT directory size for a specific network identifier. NSRTSIZE allows you to override OSRTSIZE for a maximum of nine network identifiers.

You can code a maximum of nine netid, size pairs. netid can be any network identifier other than the VTAM host network identifier. If you specify the host network identifier, it is ignored and size is used.

For network identifiers not in the list, the value specified or defaulted for OSRTSIZE is used. The OSRTSIZE start option defines the directory size for all network identifiers not defined by the NSRTSIZE option.

**NUMTREES start option**

NUMTREES = number_of_routing_trees

Notes:
1. NUMTREES is meaningful only if the NODETYPE=NN start option is also used.

range: 2–10000

Specifies a limit on the number of routing trees retained in the topology and routing services tree cache. Topology and routing services uses the trees when it receives requests for session routes. When the limit is reached, the oldest unused routing trees are discarded.

Setting a high number for NUMTREES makes session setup faster, but takes up more storage than a low number. If needed, you can free some storage by setting a lower number for NUMTREES, but session setup will then take longer because routes will have to be calculated.

You can change the value of NUMTREES with the MODIFY VTAMOPTS command while VTAM is running.

**OSIEVENT start option**
Specifies whether VTAM topology notifications are generated for CMIP events. CMIP services generates and processes a great number of status notifications from resources even though only a small percentage of the notifications are needed by a manager application program.

The OSIEVENT start option lets you adjust the number of events reported, according to the requirements of the manager application programs. For details on how you specify the requirements of manager application programs and how they send requests to CMIP services, see "z/OS Communications Server: CMIP Services and Topology Agent Guide".

You can change the value of OSIEVENT with the MODIFY VTAMOPTS command while VTAM is running. If you have coded OSIMGMT=YES, you can specify OSIEVENT=NONE or OSIEVENT=ALL with the command. If you have coded OSIMGMT=NO, you can specify any value.

**OSIEVENT=ALL**

Specifies that the VTAM topology agent generates notifications for all CMIP events. Specify OSIEVENT=ALL only if this VTAM is managed by a CMIP application program that uses filters to collect VTAM topology information but the filters do not match the patterns recognized by CMIP services. These patterns are described under in "z/OS Communications Server: CMIP Services and Topology Agent Guide".

With OSIEVENT=ALL, VTAM processes all event reports from network resources in that VTAM, regardless of the reporting flags set by the manager application program for that VTAM. The manager application program sees only the event reports matching the filters it created.

Be aware that specifying OSIEVENT=ALL can affect VTAM performance. However, the OSIEVENT=ALL start option allows the use of filters that do not match the patterns recognized by CMIP services.

**OSIEVENT=NONE**

Specifies that the VTAM topology agent creates no notifications for CMIP events, regardless of the reporting flags set by the manager application program for that VTAM.

**OSIEVENT=PATTERNS**

Specifies that the VTAM topology agent creates notifications only if there are active filters that CMIP services recognizes as requesting information for specific resources or classes of resources.

OSIEVENT=PATTERNS allows the use of certain filters that notify VTAM which specific resources, or classes of resources, should be monitored for event reports in that VTAM.

With this value, you can limit CPU cycles and storage use to the minimum needed to process the event reports from the selected resources.

**OSIMGMT start option**
Specifies whether the VTAM topology agent code and the CMIP services code is to be loaded and activated for use.

You can change the value of OSIMGMT with the MODIFY VTAMOPTS command while VTAM is running.

OSIMGMT=YES
   Specifies that the VTAM topology agent code and the CMIP services code is to be loaded and activated for use. If you want to support CMIP application programs on VTAM, including the VTAM topology agent, specify YES for this start option.

   If you specify OSIMGMT=YES, a directory definition file is required. See “CMIP services directory definition file” on page 1008 for information on coding the directory definition file. For further information on the CMIP services directory definition file, see z/OS Communications Server: SNA Network Implementation Guide.

OSIMGMT=NO
   Specifies that the VTAM topology agent code and the CMIP services code is not to be loaded. If you do not want to use the VTAM topology agent or CMIP services on this host, specify NO.

**OSITOPO start option**
Specifies whether the VTAM topology agent reports logical lines and all types of cross domain resources (CDRSCs) to CMIP manager application programs.

You can change the value of OSITOPO with the MODIFY VTAMOPTS command while VTAM is running, with the following restriction. When the current value of OSIMGMT=NO, you can specify any value for OSITOPO on the command. But, when the current value for OSIMGMT=YES, you can specify only OSITOPO=ILUCDRSC.

**OSITOPO=ALLCDRSC**

Specifies that the VTAM topology agent reports all CDRSCs to the CMIP manager application program as part of the local topology data. This value includes:

- Independent LUs (dynamic or predefined)
- Predefined CDRSCs
- Dynamic CDRSCs (except dynamic aliases)

**OSITOPO=ILUCDRSC**

Specifies that the VTAM topology agent reports all independent LUs (dynamic or predefined), but does not report other types of CDRSCs.

**OSRTSIZE start option**
OSRTSIZE = number_of_queue_pointers

range: 1–2097148

Specifies the number of queue pointers in the symbol resolution table (SRT) directory for networks other than the VTAM host node's network.

**PDTRCBUF start option**

PDTRCBUF = number_of_buffers

range: 2–255

Specifies the number of buffers to build for a session monitor request.

You can change the value of PDTRCBUF with the MODIFY VTAMOPTS command while VTAM is running.

**PIUMAXDS start option**
PIUMAXDS = 200

PIUMAXDS = calculation_factor

range: 0–2147483647

A factor used in an equation to calculate the maximum buffer limit for PIU data space.

You can change the value of PIUMAXDS with the MODIFY VTAMOPTS command while VTAM is running.

**PLUALMSG start option**

PLUALMSG = NOSUPP

PLUALMSG = SUPPRESS

Specifies whether VTAM is to suppress duplicate session setup failure messages associated with LOGAPPL failures to the same PLU.

You can change the value of PLUALMSG with the MODIFY VTAMOPTS command while VTAM is running.

**PLUALMSG=NOSUPP**

Specifies that VTAM does not suppress the session setup failure messages.

**PLUALMSG=SUPPRESS**

Specifies that VTAM suppresses duplicate session setup failure messages.

**PMTUD start option**
Notes:
1 PMTUD is meaningful only if the NODETYPE start option is also used.

A TCP/IP stack can use path MTU discovery to dynamically learn the maximum MTU for the total IP network connection. Use the PMTUD start option to enable and disable path MTU discovery for IPv4 Enterprise Extender (EE) connections and IPv6 EE connections. You can change the value of PMTUD with the MODIFY VTAMOPTS command while VTAM is running.

**PMTUD=TCPVALUE**
Indicates that VTAM accepts the TCP/IP stack (associated with EE) setting to control path MTU discovery for both IPv4 and IPv6 EE connections.

**IPv4**
For IPv4 EE connections, the TCP/IP stack associated with EE is enabled to use path MTU discovery with the PATHMTUDISCOVERY operand on the IPCONFIG profile statement. For more information about the TCP/IP PATHMTUDISCOVERY setting, see [z/OS Communications Server: IP Configuration Reference](https://www.ibm.com/support/docview.ws/docview/153937).

**IPv6**
For IPv6 EE connections, path MTU discovery is always enabled in the TCP/IP stack. The TCPVALUE setting indicates that VTAM accepts the TCP/IP stack (associated with EE) setting. Therefore, path MTU discovery is also enabled by default for IPv6 EE connections.

**PMTUD=NO**
Disables path MTU discovery for IPv4 EE connections and IPv6 EE connections.

Requirement: Path MTU discovery uses ICMP fragmentation-needed errors to detect the PMTUD for a path. If you use path MTU discovery, you must permit ICMP errors to flow to all hosts along the path of a connection. PATHMTUDISCOVERY does not function if a firewall blocks ICMP errors.

Tip: If you do not want path MTU discovery enabled for EE, but you want to limit the maximum packet size that EE will transmit, consider specifying the MTU operand available on the following major nodes:
- For EE connection networks, you can define the MTU operand on the connection network GROUP definition statements in the EE XCA major major node.
- For dial-in EE connections with associated PUs that are dynamically created, you can define the MTU operand on the model major node (DYNTYPE=EE) PU definition statement.
- For predefined EE connections, you can define the MTU operand on the PU definition statement in the switched major node.

**PPOLOG start option**
Specifies how much information is to be recorded in the primary program operator (PPO) log. When a PPO is running, the following events are always recorded in the PPO log:

- Commands issued by the program operator.
- Messages issued by VTAM in response to those commands.
- Unsolicited VTAM messages (for example, messages pertaining to alerts).

**Note:** When the NetView program is running, it is always the PPO.

The PPOLOG start option allows you to record the following additional events in the PPO log:

- Commands entered at the system console.
- Messages issued by VTAM in response to those commands.

PPOLOG is supported by the NetView program. It should be used only if the PPO can support the option.

You can change the value of PPOLOG with the MODIFY VTAMOPTS command or the MODIFY PPOLOG command while VTAM is running.

**PPOLOG=NO**

Specifies that VTAM commands entered at the system console and messages VTAM issues in response are not recorded in the PPO log.

**PPOLOG=YES**

Specifies that all VTAM commands entered at the system console (except START and HALT) and all messages VTAM issues in response are recorded in the PPO log.

**Note:** Messages that are suppressed by the MODIFY SUPP command or by the SUPP start option are recorded in the PPO log, but do not appear on the system console. However, if multiple console support (MCS) is included in the system, all suppressed messages, originally destined for an MVS console, are sent to the hardcopy log.

**PROMPT and NOPROMPT start option**
Notes:

1. A VTAM operator cannot enter the PROMPT or NOPROMPT start option; it can be coded only in ATCSTR00. The value coded in ATCSTR00 is ignored if start options are entered on the START command or if VTAM finds an error in a start list. Upon finding an error in a start list, VTAM prompts the operator so that the operator can specify the option correctly.

Specifies whether VTAM prompts the VTAM operator to enter VTAM start options. PROMPT and NOPROMPT can be coded only in the ATCSTR00 predefined list of start options. You cannot code PROMPT in any other predefined start option list. If you include PROMPT or NOPROMPT in a predefined list other than ATCSTR00, the option will be ignored.

If you code NOPROMPT in ATCSTR00, VTAM does not prompt the operator. If NOPROMPT appears in ATCSTR00, you cannot override it by including PROMPT on the START command or in a predefined list. Do not put NOPROMPT in ATCSTR00 unless you are sure that a VTAM operator will never want to change the VTAM start options.

Note: If start options are entered with the START command, PROMPT is ignored.

PSRETRY start option
Specifies the interval between attempts to switch a rapid transport protocol (RTP) session to another path that can be equal to or better than the current path.

The following values indicate the transmission priority interval:

- **low tp int**
  Specifies the time interval for low transmission priority RTP connections.

- **med tp int**
  Specifies the time interval for medium transmission priority RTP connections.

- **high tp int**
  Specifies the time interval for high transmission priority RTP connections.

- **network tp int**
  Specifies the time interval for network transmission priority RTP connections.

You can specify different transmission priority time limits for low, medium, high, and network transmission priority connections. For example, you might want a batch connection to wait longer intervals than an interactive connection before looking for another path which may be equal to or better than the current path.

If you do not specify an S, M, or H on a transmission priority interval, seconds are assumed.

**Results:**
- Specifying too short an interval might have a negative effect on network performance, especially if there are many RTP connections.
- If you specified the PSRETRY start option, the following commands trigger path switches for all RTP sessions, regardless of the PSWEIGHT start option value:
  - `MODIFY proname,TGP,TGPNAME=tg_profile,ID=adjacent_node,TGN=tgn`
    This command changes the TG characteristics for a TG.
  - `MODIFY proname,TOPO,FUNCTION=quiesce_normal,ORIG=origin_node,DEST=dest_node,TGN=tgn`
    This command changes a TG from QUIESCE to NORMAL or from NORMAL to QUIESCE.

This means that RTP sessions might be switched to the same path even if PSWEIGHT is set to LESS THAN. For information about PSWEIGHT, see “PSWEIGHT start option” on page 917. For more information, see the `MODIFY TGP` and `MODIFY TOPO` commands in z/OS Communications Server: SNA Operation.

**Immediate path switches**
- Specifies whether an immediate attempt to switch RTP sessions to another path is triggered when a link on this host is activated or changes status, regardless of the specified time intervals.

  **SCHED**
  Specifies that a link activation or status change does not trigger an
attempt to switch RTP sessions to another path. The path switch retry function follows the scheduled transmission priority time intervals.

**IMMED**
Specifies that a link activation or status change triggers an attempt to switch RTP sessions to another path.

**Guideline:** When using the function provided with PSRETRY, it is very important to ensure that the nodes in the network have a consistent and symmetrical view of the weights of the paths involved, as indicated with APPNCOS tables.

**Results:**
1. If the appropriate PSRETRY transmission priority interval value is nonzero, one path might be based on the PSRETRY time after an RTP pipe activates to store the path weight.
2. If the PSRETRY IMMED function initiates a path switch, PSRETRY does not initiate any other path switches for that transmission priority until the full PSRETRY time for that transmission priority expires.
3. If you specify IMMED, when a link on this host is activated or changes status for more than 15 times in 24 hours, the PSRETRY IMMED function does not attempt any further path switches because of status changes to the link until the next topology garbage collection time. The topology garbage collection occurs every 24 hours. You can use the `DISPLAY NET,TOPO,LIST=SUMMARY` command to display the date and time of the last topology garbage collection. See the `DISPLAY TOPO` command in [z/OS Communications Server: SNA Operation](https://www.ibm.com/support/knowledgecenter/en/SSLVMB_2.2.0/com.ibm.zos.v2r1.fsga200/sns/a02n4334.htm) for additional information.
4. PSRETRY behaves differently with links that originate on a network node and connect to an end node. If you specify PSRETRY time intervals or PSRETRY IMMED on an end node, a change in a TG that originates on the end node will trigger the path switch retry function to attempt a path switch. However, if you specify PSRETRY time intervals or PSRETRY IMMED on a network node, a change in a TG that originates on the network node will not trigger the path switch retry function. This anomaly occurs because the route selection process uses only links that originate on the end node to determine the path of an RTP pipe when an end node is one of the endpoints.

Specifying IMMED prevents a link that experiences repeated INOPs from causing excessive path switches. However, PSRETRY transmission priority intervals can still trigger RTP path switches even if a link on this host is activated or changes status for more than 15 times in 24 hours.

**Tip:**
1. You can set the value `PSRETRY=(0,0,0,SCHED)` or use the default values to disable the path switch retry function.
2. You can set the value `PSRETRY=(0,0,0,IMMED)` to disable regularly scheduled path switch retry attempts to reduce the performance impact, but to retain the path switch retry function for a link activation or status change on this host.

You can change the PSRETRY values with the MODIFY VTAMOPTS command while VTAM is running.

**PSSTRACE start option**
Specifies whether IRB trace entries, SRB trace entries, both, or neither should be generated when the VTAM internal trace (VIT) is turned on with OPTION=PSS or OPTION=ALL.

**PSSTRACE=IRB**
- Includes IRB entries in the VTAM internal trace.

**PSSTRACE=SRB**
- Includes SRB entries in the VTAM internal trace.

**PSSTRACE=BOTH**
- Includes IRB and SRB entries in the VTAM internal trace.

**PSSTRACE=NORB**
- Does not include IRB nor SRB entries in the VTAM internal trace.

You can change the value of PSSTRACE with the MODIFY VTAMOPTS command while VTAM is running.

### PSWEIGHT start option

Specifies the weight comparison between an old route and a newly calculated route before a path switch is attempted because of the PSRETRY function.

**Note:** When using the function provided with PSWEIGHT, it is very important to ensure that the nodes in the network have a consistent and symmetrical view of the weights of the paths involved, as indicated by way of APPNCOS tables.

**PSWEIGHT=EQUAL**
- Indicated that a path switch is wanted between equal weight routes.

**PSWEIGHT=LESSTHAN**
- Indicates that a path switch is wanted only if the newly calculated route is less than the current weight.

**PSWEIGHT=SAMEROUT**
- Indicates that PSRETRY will attempt to switch a rapid transport protocol (RTP) session even if the path switch results in the selection of the same route. This option will allow for a route to be switched to a more desirable VR within a VR-TG with multiple VRs.

**Result:** If you specified the PSRETRY start option, the following commands trigger path switches for all RTP sessions, regardless of the PSWEIGHT start option value:
- MODIFY procname,TGP,TGPNAME=tg_profile,ID=adjacent_node,TGN=tgn
This command changes the TG characteristics for a TG.

- MODIFY
  procname,TOPO,FUNCTI0N=quiesce_normal,ORIG=origin_node,DEST=dest_node,TGN=tgn
  This command changes a TG from QUIESCE to NORMAL or from NORMAL to QUIESCE.

This means that the RTP sessions might be switched to the same path even if PSWEIGHT is set to LESSTHAN. For information about PSWEIGHT, see "PSRETRY start option" on page 914. For more information, see the MODIFY TGP and MODIFY TOPO commands in z/OS Communications Server: SNA Operation.

QDI0STG start option

Notes:
1 QDI0STG defaults to MAX for 64-bit (z/Architecture) machines and MIN for non-64-bit machines.

Specifies how much storage VTAM keeps available for read processing for all OSA QDIO data devices. Units are defined in QDIO SBALs (QDIO read buffers). Each SBAL is 64k. For most users the default setting will be the most suitable option. The storage used for this read processing is allocated from CSM data space 4k pool, and is fixed storage.

The IBM recommended values can be configured by specifying MAX, AVG, or MIN, which are predefined constants (number of SBALs) that are most appropriate for this type of adapter. MAX allows for the best performance (for example, throughput), but requires more storage. Conversely, MIN could be used for devices with lighter workloads or where system storage might be constrained. Also instead of using one of the 3 predefined values, users can instead define a specific value for QDI0STG (the exact number of SBALs), such as QDI0STG=100. The supported range is 8 (minimum supported value) to 126 (maximum supported value). The predefined and possible user-defined values for QDI0STG are the following items:

QDI0STG=MAX
  Specifies the recommended maximum (64 SBALs for a total of 4 MB). This is the default for 64-bit z/Architecture machines.

QDI0STG=AVG
  Specifies the recommended average (32 SBALs for a total of 2 MB).

QDI0STG=MIN
  Specifies the recommended minimum (16 SBALs for a total of 1 MB). This is the default for non 64-bit z/Architecture machines.

QDI0STG=nnn
  Specifies the user defined specific number of SBALs, valid range of 8 - 126 (see Guideline 4).

Guidelines:
1. If you elect to define a specific value, carefully evaluate your system storage and performance needs for your QDIO devices. The following calculation is used to determine the amount of storage used: (NNN x 64K) x number of active inbound queues for each active QDIP data device. When QDIO inbound workload queueing (IWQ) is not enabled, only one inbound queue is active for each active device. When IWQ is enabled, one or more ancillary input queues might also be active. See QDIO inbound workload queueing in z/OS Communications Server: IP Configuration Guide to determine how many ancillary input queues are enabled for each QDIO data device.

2. You can use VTAM tuning stats to evaluate your needs and usage. Under a sample (typical) workload, the NOREADS counter should remain low (close to 0). If this count does not remain low you might need to consider a higher setting for QDIOSTG.

3. You can override the QDIOSTG value for a given OSA-Express QDIO device by using the READSTORAGE parameter on the IPAQENET LINK, the IPAQENET INTERFACE, or the IPAQENET6 INTERFACE statement in the TCP/IP profile.

4. Each active OSA-Express QDIO DATAPATH device consumes a large amount of fixed storage. Defining a large number (for example, 8 or more devices per z/OS image) of QDIO devices can cause z/OS Communications Server to consume a significant amount of fixed storage. This could lead to degradation of overall system performance. When configuring a large number of devices, it is important to use the controls provided to manage and tune the amount of fixed storage consumed by these devices. Review the following parameters with this in mind:
   - VTAM QDIOSTG start option
   - READSTORAGE specifications in the TCP/IP profile
   - FIXED MAX specification in the IVTPRM00 parmlib member for Communication Storage Manager (CSM).

RESUSAGE start option
RESUSAGE = usage_limit

Notes:
1. RESUSAGE is meaningful only if the NODETYPE=NN start option is also used.

   range: 0–2147483647

Specifies how many times a resource (node or TG) is used during route selection before that resource is considered overused. After a resource is overused, any routing trees using it will be reconstructed in an attempt to use alternate resources. RESUSAGE is valid only when NODETYPE=NN.

Note: RESUSAGE=0 indicates that the resource is never considered to be overused.

RESUSAGE does not apply to ENs and endpoint TGs.

You can change the value of RESUSAGE with the MODIFY VTAMOPTS command while VTAM is running.

**ROUTERES start option**

ROUTERES = resistance_value

Notes:
1. ROUTERES is meaningful only if the NODETYPE=NN start option is also used.

   range: 0–255

Specifies a route addition resistance value. This is a relative measure of the desirability for this node to perform intermediate session routing function. The value specified for this node is compared to the values of other network nodes during route calculation. The lower the value, the more desirable it is to have this node provide intermediate session routing.

You can change the value of ROUTERES with the MODIFY VTAMOPTS command while VTAM is running.

**RSIRFMSG start option**
Controls the display of messages IST1460I, IST1461I, IST2102I, IST2103I, and IST2104I. These messages are issued when a session initiation request fails and RSCV data exists.

Requirement: The RSIRFMSG start option is subordinate to the SIRFMSG and DSIRFMSG start options and SIRFMSG coded on a CDRSC or APPL. Therefore, you must code SIRFMSG or DSIRFMSG as ALLSSCP or OLUSSCP to receive the RSIRFMSG messages.

You can change the value of RSIRFMSG with the MODIFY VTAMOPTS command while VTAM is running.

RSIRFMSG=ALLSSCP
   Specifies that messages are issued in all SSCP's.

RSIRFMSG=OLUSSCP
   Specifies that messages are issued only in the origin logical unit (OLU) SSCP.

RSIRFMSG=NONE
   Specifies that no messages are issued in any SSCP.

SACONNS start option
Specifies if subarea connections will be used. SACONNS need only be specified when defining a pure APPN EN or NN that will use a HOSTSA number other than the default value (1). For details on defining various node configurations, see Table 70 on page 859.

**SACONNS=NO**

When NODETYPE is coded, indicates that this VTAM will not provide subarea function. (That is, this VTAM will operate as a pure APPN node.)

**SACONNS=YES**

Indicates that this VTAM will provide subarea function.

### SAVERSCV start option

<table>
<thead>
<tr>
<th>SAVERSCV</th>
<th>(NO, KEEP)</th>
<th>(YES, KEEP)</th>
<th>(NO, FREE)</th>
<th>(YES, FREE)</th>
<th>SAWDATA, KEEP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Notes:**

1. SAVERSCV is meaningful only if NODETYPE is also used.

Specifies whether or not Route Selection Control Vectors (RSCVs) for future sessions are to be saved for display purposes following the completion of session activation. SAVERSCV also specifies whether RSCV storage for existing sessions should be freed immediately. By default, these RSCVs are not saved. If YES or SAWDATA is specified, the D NET, SESSIONS, PATHINFO=YES, SID= command can then be used to display the saved RSCVs for the particular session specified.

SAVERSCV covers any session RSCVs reported by a DLUR node because of DLURSAW=YES having been specified as a start option, but SAVERSCV does not cover any RTP information (including HPR RSCVs) reported by a DLUR node.

You can change the value of SAVERSCV with the MODIFY VTAMOPTS command while VTAM is running.

**Note:** The secondary values, FREE and KEEP, have no effect at VTAM START time.
SAVERSCV=(FREE)
Specifies that RSCV storage for existing sessions is to be freed immediately. The primary value currently in effect (YES, NO, or SAWDATA) remains unchanged.

SAVERSCV=NO
Specifies that RSCV storage for future sessions are not saved following the completion of session activation. By default, RSCVs for existing sessions will remain.

SAVERSCV=(NO, FREE)
Specifies that RSCV storage for future sessions are to be freed following session activation, and that RSCV storage for existing sessions is to be freed immediately.

SAVERSCV=(NO, KEEP)
Specifies that RSCV storage for future sessions are to be freed following session activation, but that RSCV storage for existing sessions will remain.

SAVERSCV=YES
Specifies that RSCV storage for future sessions are to be saved following the completion of session activation.

SAVERSCV=SAWDATA
Specifies that only the RSCVs associated with any sessions matching the PLU/SLU combination in the SAW data filter table are saved. This option can be used whether or not NetView is running on the given VTAM. If a new SAW data filter table is activated and if SAVERSCV=SAWDATA, the SAW data filter table will control which RSCVs are reported to NLDM and will impact the saving and reporting of RSCVs associated with future sessions. The reporting of RSCVs for sessions that are already active will be governed by the previous SAW data filter table.

Note: Changing the SAVERSCV option from YES to NO, or from SAWDATA to NO, is intended to indicate that RSCV storage for future sessions is not to be saved.

SAWMAXDS start option
SAWMAXDS start option

range: 1–2147483647

A factor used in an equation to calculate the maximum buffer limit for SAW data space.

SAWMAXDS

SAWMAXDS = calculation_factor

SAWMXQPK start option

range: 0–2147483647

Used to calculate the number of session awareness (SAW) data space buffers to queue before packing SAW buffers.

You can change the value of SAWMXQPK with the MODIFY VTAMOPTS command while VTAM is running.

SDLCMDRS start option
Specifies whether VTAM records SDLC inoperative and statistical miscellaneous data records (MDRs) in LOGREC.

**Note:** If you do not record MDRs, and a problem occurs in an SDLC device, problem diagnosis can be difficult because there are no MDRs in LOGREC.

STATS,NOSTATS and YES,NO mean the same. STATS,NOSTATS is the preferred coding. YES,NO is supported on this start option for compatibility with prior releases.

You can change the value of SDLCMDRS with the MODIFY VTAMOPTS command while VTAM is running.

**YES**
- Specifies that VTAM records the SDLC statistical MDRs in LOGREC.

**STATS**
- Specifies that VTAM records the SDLC statistical MDRs in LOGREC.

**NO**
- Specifies that VTAM does not record the SDLC statistical MDRs in LOGREC.

**NOSTATS**
- Specifies that VTAM does not record the SDLC statistical MDRs in LOGREC.

**INOPS**
- Specifies that VTAM records the SDLC inoperative MDRs in LOGREC.

**NOINOPS**
- Specifies that VTAM does not record SDLC inoperative MDRs in LOGREC.

**SECLVLCP start option**
Notes:
1. The SECLVLCP start option is meaningful only if the NODETYPE and VERIFYCP start options are also used.

   dependency: LU 6.2 sessions

Specifies whether VTAM uses the basic or enhanced protocol for session-level security verification during activation of an LU 6.2 session involving control points.

Some VTAMs allow only basic protocol, while others allow basic or enhanced. VTAMs that allow both protocols fall into two classes:
- Those that allow choice of basic or enhanced protocol at any time
- Those that allow a restricted choice: after the enhanced protocol is used to a particular partner, only the enhanced protocol can be subsequently used to that partner.

**SECLVLCP=LEVEL1**

   Specifies that the basic protocol is used. If a partner CP does not support the basic protocol, VTAM rejects the session and issues a sense code.

   LEVEL1 is useful in a VTAM that is connected to an earlier version of VTAM that is using session-level security verification. LEVEL1 prevents the restricted-choice class of VTAMs from locking out subsequent sessions to the earlier version of VTAM.

**SECLVLCP=LEVEL2**

   Specifies that the enhanced protocol is used. If a partner CP does not support the enhanced protocol, VTAM rejects the session and issues a sense code.

   LEVEL2 is useful when all communicating VTAMs have session-level security, and you want to prevent sessions being set up with the less stringent LEVEL1 protocol.

**SECLVLCP=ADAPT**

   Specifies that VTAM will negotiate whether to use the basic or enhanced protocol for LU-LU session-level security verification, depending on the protocol supported by the partner CP. VTAM attempts to use the enhanced protocol but will use the basic protocol if a partner CP does not support the enhanced protocol.

   ADAPT is useful in a VTAM that communicates with those VTAMs using a particular partner, only the enhanced protocol can be subsequently used to that partner.

**SIRFMSG start option**
Controls the display of messages IST663I, IST664I, IST889I, and their message groups. These messages are issued when a session initiation request fails. The ASIRFMSG, ESIRFMSG, FSIRFMSG, and RSIRFMSG start options can be used in conjunction with SIRFMSG to generate additional messages as part of this message group.

- ASIRFMSG controls whether the IST890I and IST896I messages are displayed.
- ESIRFMSG controls whether the IST891I, IST892I, and IST893I messages are displayed.
- FSIRFMSG controls whether the IST1704I, IST1705I, IST894I, and IST895I messages are displayed.
- RSIRFMSG controls whether the IST1460I, IST1461I, IST2102I, IST2103I, and IST2104I messages are displayed.

You can change the value of SIRFMSG with the MODIFY VTAMOPTS command while VTAM is running.

**SIRFMSG=ALLSSCP**

Specifies that messages are issued in all SSCP.

**SIRFMSG=OLUSSCP**

Specifies that messages are issued only in the origin logical unit (OLU) SSCP.

**SIRFMSG=NONE**

Specifies that no messages are issued in any SSCP.

### SLOWVAL start option
Specifies that a network node which has been placed in a SLOWDOWN mode can now allow a maximum SLOWDOWN time to be set.

There are two allowed formats for coding start option SLOWVAL:
- SLOWVAL=warn_time
- SLOWVAL=(warn_time,end_time)

**warn_time**
Specifies the amount of time a resource can be in SLOWDOWN before a warning message will be sent to the console stating that the resource has been in slowdown for set amount of time. The device can be automatically removed from SLOWDOWN after the specified end_time, or specify SLOWVAL=0 for an informational warning message whenever a device is in SLOWDOWN for a long period of time or if the device comes out of SLOWDOWN mode.

**end_time**
Specifies the amount of time a resource can be in SLOWDOWN mode before it is automatically taken out of SLOWDOWN mode by VTAM. After a warning message has been issued, a new message will be displayed as soon as that resource is taken out of SLOWDOWN mode (whether VTAM-initiated or DEVICE-initiated). If end_time is specified, its value must be greater than or equal to the value specified for warn_time.

**SLUALMSG start option**
Specifies whether VTAM is to suppress messages for a session setup failure caused when the SLU is not enabled for a session.

You can change the value of SLUALMSG with the MODIFY VTAMOPTS command while VTAM is running.

**SLUALMSG=NOSUPP**
- Specifies that VTAM does not suppress the session setup failure messages.

**SLUALMSG=SUPPRESS**
- Specifies that VTAM suppresses the session setup failure messages.

### SMEAUTH start option

Controls session management exit (SME) authorization data without having to modify or code an SME. See z/OS Communications Server: SNA Customization for more information about the SME authorization data.

**SMEAUTH=DISCARD**
- Prevents SME authorization data from being saved.

**SMEAUTH=EXIT**
- Allows the SME to determine if the authorization data should be kept or freed when the session is activated.

**SMEAUTH=KEEP**
- Saves the authorization data for all sessions.

### SNAMGMT start option
SNAMGMT = NO

Specifies whether the ISTMGCEH subtask, which is the socket server for the SNA network management interface, is to be attached and the socket opened for use. See z/OS Communications Server: IP Programmer’s Guide and Reference for the programming specifications for the SNA network management interface.

You can change the value of SNAMGMT with the MODIFY VTAMOPTS command while VTAM is running.

SNAMGMT = YES

Specifies that the ISTMGCEH subtask is to be attached and the Network Management Interface socket opened for use. If you want to support application programs that exploit the Network Management Interface, specify SNAMGMT=YES for this start option.

If you specify SNAMGMT=YES, an AF_UNIX socket will be opened and will allow connections to be made by authorized client applications.

There is a security mechanism to restrict access to this function. To permit only particular users to use this function, define the RACF (or equivalent external security manager product) resource IST.NETMGMT.sysname.SNAMGMT in the SERVAUTH class. For users permitted to use the function, the MVS user ID is permitted to the defined resource. If the resource is not defined, then only superusers (users permitted to BPX.SUPERUSER resource in the FACILITY class) are permitted to it.

The VTAM address space user ID must be defined with write access to the /var directory.

The administrator must define an OMVS segment for VTAM if one is not already defined.

The directory /var/sock will be created by VTAM when enabling this function with MLS seclabel SYSMULTI.

SNAMGMT = NO

Specifies that the ISTMGCEH subtask is not to be attached and the Network Management Interface socket is not to be opened for use. If you do not want to support application programs that exploit the Network Management Interface, specify SNAMGMT=NO for this start option.

SNAPREQ start option
SNAPREQ = number_of_requests

range: 1–32767

Specifies, when the SMS (buffer use) trace is active, the threshold number of requests for VTAM buffers after which to take a snapshot dump.

**SNVC start option**

SNVC = subnet_visit_count

Notes:
1. SNVC is meaningful only if the BN=YES start option is also used.

range: 1–255

Defines a maximum number of networks this border node will search when looking for a resource. This start option is valid only when BN=YES for this node.

You can change the value of SNVC with the MODIFY VTAMOPTS command while VTAM is running.

Note:
1. If SNVC=1 is coded, the search scope is restricted to the originating subnetwork. A value of SNVC=2 is required to allow a request to traverse one subnetwork boundary.
2. The maximum subnet count can be modified downwards if a node along the path has a lower SNVC count. However, the received SNVC is never modified upward.

**SONLIM start option**
SONLIM = (60, 30)

Defines the maximum percentage of fixed I/O buffers available for the session outage notification (SON) request/response units.

VTAM calculates the \( \text{percent}_1 \) number of buffers and the \( \text{percent}_2 \) value, using the values you specify as the percentage of the I/O buffers available for SON. These numbers are calculated when VTAM is started, and are not affected by any increase in the number of I/O buffers because of dynamic buffer expansion.

The session outage notification does not pre-allocate buffers. No I/O buffers are allocated until SON processing begins.

There are three allowed formats for coding start option SONLIM:

- \( \text{SONLIM}=\text{percent}_1 \)
- \( \text{SONLIM}= (\text{percent}_1, \text{percent}_2) \)
- \( \text{SONLIM}= (\text{percent}_1) \)

\( \text{percent}_1 \)

Specifies the percentage of the fixed I/O buffer pool that can be allocated for SON request units. VTAM calculates this percentage by using the \( \text{percent}_1 \) value and the \( \text{baseno} \) value coded for IOBUF. This is the maximum number of buffers that can be allocated for notification request units. If VTAM cannot initiate all the SON request units (one buffer is required for each request unit) within the available storage, it suspends the notification process until the number of buffers allocated for SON request units drops below the calculated threshold. Then the process begins again and more SON request units are generated.

\( \text{percent}_2 \)

Specifies the percentage of fixed I/O buffers that are still allocated to session outage notification request units when the notification process begins again. After the number of buffers allocated to session outage notification drops below the \( \text{percent}_2 \) value, the process begins generating SON reports again.

\textbf{Note:} The value of \( \text{percent}_2 \) must be less than the value of \( \text{percent}_1 \).
Notes:

1 SORDER is meaningful only in an interchange node or a migration data host.

Controls the order in which the APPN and subarea networks are searched when a network search request is received from the subarea network. This search order may be overridden if the target LU’s location is already identified and SSCPORD=PRIORITY is specified. SORDER is valid only for network nodes (NODETYPE=NN) functioning as interchange nodes, and end nodes functioning as migration data hosts. For further information on the interaction of the SORDER start option with other start options, see the z/OS Communications Server: SNA Network Implementation Guide.

You can change the value of SORDER with the MODIFY VTAMOPTS command while VTAM is running.

For more granular control over the search order, a different SORDER can be specified for each individual adjacent SSCP table. For more information, see “Adjacent SSCP table” on page 705.

See “cdrmname” on page 707 for related information on network searches.

SORDER=ADJSSCP  
Specifies that the search should be conducted based on the order coded in the adjacent SSCP table. If you specify SORDER=ADJSSCP, the APPN network is not searched unless you also coded the operand ISTAPNCP as one of the entries within the adjacent SSCP table.

SORDER=APPN  
Specifies that searching the APPN network is preferred, regardless of whether (or where) the operand ISTAPNCP is coded in the adjacent SSCP table. However, the owning CDRM, the predefined owning CDRM, and any previously successful adjacent CDRMs (if SSCPORD=PRIORITY is specified), if known, are searched before the APPN network.

SORDER=APPNFRST  
Specifies that the APPN network is to be searched first, without regard to whether (or where) the operand ISTAPNCP is coded in the adjacent SSCP table, or whether the owning CDRM, the predefined owning CDRM, or any previously successful adjacent CDRMs are known.

Note: SORDER=APPNFRST and SSEARCH=APPNFRST are meaningful only on Network Nodes. If SSEARCH=APPNFRST is coded on an EN, it will prevent finding resources on the EN’s domain.

SORDER=SUBAREA  
Specifies that the subarea network is to be searched first, regardless of whether (or where) the operand ISTAPNCP is coded in the adjacent SSCP table.
SRCHRED start option

 Specifies whether this node can reduce searches for resources which are found to be unreachable. See z/OS Communications Server: SNA Network Implementation Guide for information about using the search reduction facility.

 You can change the SRCHRED value with the MODIFY VTAMOPTS command while VTAM is running.

 **SRCHRED=OFF**
 - Specifies that this node does not reduce searches.

 **SRCHRED=ON**
 - Specifies that this node does reduce searches.

 SRCOUNT start option

 Notes:

 1. SRCOUNT is meaningful only if the SRCHRED=ON start option is also used.

 range: 0–65535

 Specifies how many search requests can be limited before VTAM performs another resource discovery search. SRCOUNT is meaningful only when search reduction is active (SRCHRED=ON).

 If SRCOUNT=0 is coded, then the search count threshold is not used.

 You can change the value of SRCOUNT with the MODIFY VTAMOPTS command while VTAM is running.

 SRTIMER start option
Notes:
1. SRTIMER is meaningful only if the SRCHRED=ON start option is also used.

range: 0–65535 seconds

Specifies the number of seconds during which VTAM does not conduct searches for an unreachable resource. Once the time limit has expired, VTAM will conduct a search. SRTIMER is meaningful only when search reduction is active (SRCHRED=ON).

If SRTIMER=0 is coded, then the search time threshold is not used.

You can change the value of SRTIMER with the MODIFY VTAMOPTS command while VTAM is running.

**SSCPDYN start option**

Specifies whether VTAM dynamically adds entries to the adjacent SSCP table.

If you specify SSCPDYN=NO, you might have to define additional adjacent SSCP tables. When an application issues a CLSDST PASS macroinstruction during session setup, VTAM does not build an adjacent SSCP table in the application’s network. (When SSCPDYN=YES is coded in the application’s network, VTAM builds such a table automatically.)

This start option is closely related to SSCPORD. See [z/OS Communications Server SNA Network Implementation Guide](https://www.ibm.com/mg8212) for information about how to use these two start options.

**SSCPDYN=NO**

Specifies that VTAM does not add new entries unless it is the owner of the resource. The owner is determined by the CDRM operand of the CDRSC definition statement, or by session initiation.

**SSCPDYN=YES**

Specifies that VTAM adds a new entry to a cross-domain resource’s adjacent SSCP table whenever it receives a session initiation request from the resource through an SSCP that is not already in the table.
SSCPID start option

```plaintext
SSCPID—sscpid
range: 0–65535
```

Specifies that part of an SSCP identifier used when a physical unit or external CDRM establishes contact with VTAM. This start option is required.

It is recommended that the SSCPID in each network and each connecting network be unique. If the SSCPs within a network do not have cross-network sessions, the SSCPIDs are not required to be unique. If the SSCPs do have cross-network sessions, the SSCP for each end of the cross-network sessions must have a different SSCPID. If more than one SSCP in one network has the same SSCPID, they cannot all have a session with the same SSCP in another network.

Note:
1. VTAM determines the ACTCDRM contention winner by comparing the SSCPID of each SSCP. If ACTCDRM contention occurs on two SSCPs with the same SSCPID, the contention must be resolved by the network operators.
2. If you plan to incorporate a non-VTAM host, VTAM’s SSCPID and HOSTSA start options values should be coded so that both values are either higher or lower than the non-VTAM host.

SSCPNAME start option

```plaintext
SSCPNAME—name
```

Specifies the required 1–8 character name of the VTAM SSCP or CP. You should use the same name here that you use on the host CDRM in multiple-domain networks. This start option is required.

When you code the SSCPNAME start option, consider the items listed below. The first three items are requirements. The fourth item is a suggestion for improving usability and easing network management.

- The SSCPNAME must be the same as the name on the NAME operand of the corresponding GWNAU definition statement, if one is coded. (The GWNAU definition statement is used in the gateway NCP generation to predefine the gateway SSCP’s alias network address within the gateway NCP.)
  For more information on the GWNAU definition statement, see the NCP, SSP, and EP Resource Definition Reference.
- The SSCPNAMEs and CDRM names must be unique among SSCPs that will participate in cross-domain or cross-network sessions.
- When VTAM is defined as a type 2.1 node to another VTAM or NCP on a switched SDLC connection, specify the SSCPNAME value on the CPNAME operand of the PU statement. See z/OS Communications Server: SNA Network Implementation Guide for information about casual connection.
- The SSCPNAME should match the name on the host CDRM definition statement in a subarea environment, and must match the host CDRM definition statement in an APPN or mixed APPN-subarea environment to avoid errors. See “cdrmname” on page 155 for further details.
**SSCPORD start option**

Specifies whether VTAM searches an adjacent SSCP table in priority order (the default) or in the order in which the table is defined.

You can change the value of SSCPORD with the MODIFY VTAMOPTS command while VTAM is running.

For more granular control over the search order, a different SSCPORD can be specified for each individual adjacent SSCP table. See "Adjacent SSCP table" on page 703 for more information.

This start option is closely related to SSCPDYN. See **z/OS Communications Server: SNA Network Implementation Guide** for information about how to use these two start options.

**SSCPORD=DEFINED**

Specifies that VTAM searches the adjacent SSCP table in the same order that SSCPs are listed, if past session initiation attempts were successful, with the exception that the owner, if identified, is tried first.

**SSCPORD=PRIORITY**

Specifies that VTAM gives preference to SSCPs it uses. The search order is as follows:

1. The CDRM that owns the resource (if identified).
2. The SSCPs for which the most recent session-initiation attempt succeeded.
3. The SSCPs for which no session-initiation attempt has been made.
4. The SSCPs for which the last session-initiation attempt failed.

**SSDTMOUT start option**
SSDTMOUT = 30

SSDTMOUT = number_of_seconds

range: 1–65535

Specifies the number of seconds that can pass without a new LU-LU session beginning before VTAM automatically disconnects the switched PU.

You can change the value of SSDTMOUT with the MODIFY VTAMOPTS command while VTAM is running.

**SSEARCH start option**

- SSEARCH=YES
- SSEARCH=NO
- SSEARCH=CACHE
- SSEARCH=APPNFRST

**Notes:**

1. SSEARCH is meaningful only if the NODETYPE=NN start option is also used.

Controls whether the subarea network is searched when a network search request is received from the APPN network. For further information on the interaction of the SSEARCH start option with other start options, see the [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/SSEPGG_2.2.0/com.ibm.zos.v2r1.sna.doc/guides/guide/gsna0069.html).

You can change the value of SSEARCH with the MODIFY VTAMOPTS command while VTAM is running.

- **SSEARCH=CACHE**
  Specifies that the subarea network is not to be searched unless it is known that the target LU is located in the subarea network. This knowledge might exist as a result of system definition or as a result of an earlier successful search.

- **SSEARCH=APPNFRST**
  Specifies that if SSEARCH=APPNFRST is coded on an EN, it will prevent finding resources on the EN's domain. It is meaningful only on Network Nodes.

- **SSEARCH=NO**
  Specifies that the subarea network is not to be searched.

- **SSEARCH=YES**
  Specifies that the subarea network is to be searched.

**STRGR start option**
Enables you to rename the generic resources structure.

STRGR is valid only when the coupling facility function is supported. For VTAMs that are to share the same generic resource information, you must specify the same STRGR value for each VTAM. If subplexing is in effect for this VTAM node (that is, start option XCFGRPID=vv was specified), the actual structure name that VTAM attempts to connect to is suffixed by the XCF group ID. Thus, if XCFGRPID=11 was specified, the default structure name would be ISTGENERIC11 and if STRGR=ISTMYSTR was coded, VTAM would connect to the structure ISTMYSTR11. The fully suffixed generic resource structure name that will be accessed by this VTAM can be displayed by issuing the D NET,ID=VTAM command.

**STRGR=ISTGENERIC**

Specifies the IBM-supplied generic resources structure name.

**STRGR=NONE**

Specifies that VTAM will not check for the existence of a generic resources coupling facility structure. Coding STRGR=NONE disables the generic resources function for this VTAM, even if a structure has been defined.

**STRGR=structure_name**

Specifies a user-defined generic resources structure name. The `structure_name` value can have up to 16 characters. If subplexing is in effect, the `structure_name` value can have only up to 14 characters. The value must have a minimum of 3 characters, and these must be the characters IST. The value can contain numeric characters, alphabetic characters, national characters ($, @, #), or an underscore (_).

If the structure name specified on the STRGR option is not defined at VTAM initialization, then no connection attempt is made and the generic resource function is disabled.

**STRMNPS start option**
Specifies the name of the MNPS structure to be used for multinode persistent sessions.

If subplexing is in effect for this VTAM node (that is, start option XCFGRPID= was specified), the actual structure name that VTAM will attempt to connect to will be suffixed by the XCF group ID. Thus, if XCFGRPID=11 was specified, the default structure name would be ISTMNPS11 and if STRMNPS=ISTMYSTR was coded, VTAM would connect to the structure ISTMYSTR11. The fully suffixed MNPS structure name that will be accessed by this VTAM can be displayed by issuing the D NET,ID=VTAM command.

**STRMNPS=ISTMNPS**

Specifies the default name of the MVS coupling facility structure to be used for multinode persistent sessions.

**STRMNPS=NONE**

Specifies that VTAM will not check for the existence of a multinode persistent sessions coupling facility structure. Coding STRMNPS=NONE disables the multinode persistent sessions function for this VTAM, even if a structure has been defined.

**STRMNPS=structure_name**

Specifies a multinode persistent sessions structure name that is 1-16 characters in length. If subplexing is in effect, the structure_name value can have only up to 14 characters. The name can contain numeric characters, alphabetic characters, national characters ($, A, #) or an underscore (_). The first 3 characters of the name must begin with IST.

**Note:** If you are using multiple coupling facility structures for multinode persistent session support, the number of characters used to specify the name of the base structure (STRMNPS start option) including the 2-digit group ID suffix (if subplexing is in effect), must account for characters needed for unique names for all alternate structures. See z/OS Communications Server: SNA Network Implementation Guide for information about using alternate structures.

**SUPP start option**
Specifies the highest class of VTAM messages for which VTAM suppresses message output to the VTAM operator console, and suppresses transmission to a program operator, if one exists. Messages that are generated in response to an operator request (such as the DISPLAY command) and messages that require a response (prompting messages) cannot be suppressed.

If multiple console support (MCS) is included in the system, all suppressed messages, originally destined for an MVS console, are sent to the hardcopy log. These suppressed messages are also copied to the PPO log if PPOLOG=YES was specified.

If the destination for a suppressed message is a program operator, the message is lost.

You can change the value of SUPP with the MODIFY VTAMOPTS command or the MODIFY SUPP command while VTAM is running.

VTAM suppresses messages within a multiple line write-to-operator (MLWTO) group based on the suppression level of the first message in the group. VTAM ignores the suppression levels for subsequent messages within the group.

See z/OS Communications Server: SNA Messages for individual suppression levels.

**SUPP=NOSUP**

Specifies that all VTAM messages are printed at the console. (NOSUP means “no suppression”).

**SUPP=INFO**

Specifies that informational messages are suppressed. Informational messages are those that inform the operator that commands or procedures have been accepted for processing.

**SUPP=WARN**

Specifies that warning messages (as well as informational messages) are suppressed. Warning messages identify error conditions that do not cause commands to fail or be rejected. These messages inform the operator that there is a problem, such as an operand that is not valid or a minor node that cannot be activated, but that VTAM can continue to process other parts of the command or procedure.

**SUPP=NORM**

Specifies that normal completion messages (as well as informational and warning messages) are suppressed. Normal completion messages inform the operator that commands have completed processing successfully, a configuration has been activated successfully, or a procedure has been terminated.

**SUPP=SER**

Specifies that serious error messages (as well as informational, warning, and
normal completion messages) are suppressed. Serious error messages identify error conditions that cause commands or procedures to fail. These messages tell the operator that a command must be reentered or a procedure reinitiated.

Error messages that identify an even more serious situation, such as the abnormal termination of a user task or of VTAM itself, cannot be suppressed.

**SWNORDER start option**

```
SWNORDER - (CPNAME, FIRST)
```

Specifies the way VTAM locates a switched PU. If you do not code SWNORDER, VTAM searches for a PU by the CPNAME first. If VTAM does not find the PU by the CPNAME, it searches by the station identifier (IDBLK and IDNUM operands on the PU definition statement for the switched major nodes) since the second parameter defaults to FIRST.

If VTAM does not find the PU using the first SWNORDER parameter value you specify, and the second parameter of SWNORDER is specified as ONLY, the search for the PU terminates. For example, if you code **SWNORDER=(CPNAME,ONLY)** and VTAM cannot find a PU with a matching CPNAME, no attempt will be made to find the PU using the station identifier. If dynamic PUs are allowed, a dynamic PU will be created and PU connection processing will continue. If dynamic PUs are not allowed, the PU connection process will fail.

You can change the value of SWNORDER with the MODIFY VTAMOPTS command while VTAM is running. You can override this start option for specific resources by coding the SWNORDER operand on the GROUP or LINE statements.

```
SWNORDER=(value_1,value_2)
```

- **value_1=CPNAME**
  - Specifies that VTAM searches for a switched PU by the CPNAME first.

- **value_2=FIRST**
  - If a switched PU is not found using the value specified by **value_1**, the alternate value of **value_1** will be used to search for a switched PU.

- **value_1=STATNID**
  - Specifies that VTAM searches for a switched PU by the station identifier.

- **value_2=ONLY**
  - If a switched PU is not found using the value specified by **value_1**, a search using the alternate value of **value_1** will not be performed.

**Note:** SWNORDER controls only regular (non-DLUR) switched PUs. For DLUR-defined PUs, see the DLRORDER start option.
TCPNAME start option

TCPNAME specifies the job name of the TCP stack through which VTAM communicates with the IP network. The value specified for tcp_job-name is the 1-8 character job name of the required TCP stack.

The TCP/IP stack must be at a level to support Enterprise Extender. However, if there are multiple TCP/IP stacks, you may need to specify one, as there are products which predefine the IP address of partner nodes, and these nodes require z/OS CS to send through a given address, and, therefore, a given TCP/IP stack.

Note:
1. If neither the IPADDR nor the HOSTNAME operand are specified on any of the GROUP definition statements within the Enterprise Extender XCA major node, then either the HOSTNAME, TCPNAME or IPADDR start option must be specified in order to initiate an Enterprise Extender link. If VTAM has been activated without the specification of one of these options, then the MODIFY VTAMOPTS command can be used to set one of these options. In this case, if an Enterprise Extender link activation is attempted without HOSTNAME, TCPNAME or IPADDR specified, then the link activation will fail with the IST1890I message.

2. TCPNAME can be dynamically modified using MODIFY VTAMOPTS but the new value will not be used until all lines in the XCA major node used for Enterprise Extender are inactive. However, displays of VTAM start options will show the new value immediately. Any subsequent line activation from the Enterprise Extender XCA major node will make use of the new TCPNAME value.

3. The TCPNAME start option is required in a CINET environment if any of the following conditions exist:
   - The IPADDR parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
   - The HOSTNAME parameter is specified on any of the Enterprise Extender XCA major node GROUP definition statements.
   - The HOSTNAME keyword is specified as a VTAM start option.

   If OMVS is not initialized at Enterprise Extender line activation, a Common INET environment is assumed.

   In this case, if an Enterprise Extender link activation is attempted without TCPNAME specified, link activation will fail with message IST1897I.

4. TCPNAME is meaningful only if the NODETYPE start option is also used.

Restriction: All local Enterprise Extender static VIPA addresses must be associated with a single TCP/IP stack. Even in Common-INET environments, EE always establishes affinity with one and only one TCP/IP stack.

Rules for EE line activation:
- In a CINET environment, you must specify the TCP/IP stack name (using the TCPNAME start option) whenever host name resolution is to be performed during an EE line activation.
In a CINET environment, you must specify the TCP/IP stack name (using the TCPNAME start option) whenever the HOSTNAME or IPADDR value is specified on any GROUP statements in the EE XCA major node, even if no lines in the group have ever been activated.

In an INET or a CINET environment, you must specify the TCP/IP stack name (using the TCPNAME start option) when neither the HOSTNAME value nor the IPADDR value have been specified on any GROUP statements in the EE XCA major node, and neither the HOSTNAME start option nor the IPADDR start option have been specified.

If a HOSTNAME value or an IPADDR value is specified on any GROUP statements in the EE XCA major node, then a HOSTNAME value or an IPADDR value must be specified (or inherited from the HOSTNAME or IPADDR start option) on all GROUP statements for which EE lines will be activated.

Rule for learning a local static IPv4 VIPA address for EE from the TCP/IP stack:

VTAM dynamically obtains a local IPv4 VIPA address from the TCP/IP stack only when neither the HOSTNAME value nor the IPADDR value have been specified on any GROUP statements in the EE XCA major node, and neither the HOSTNAME start option nor the IPADDR start option have been specified. In this case, the TCP/IP stack name must be specified (using the TCPNAME start options). The TCP/IP stack is responsible for obtaining a local static IPv4 VIPA address for use by Enterprise Extender.

Rules for setting the HOSTNAME and IPADDR start options:

- When an IPv4 address for EE has been obtained dynamically from the TCP/IP stack, neither the HOSTNAME nor the IPADDR start option can be set (using the MODIFY VTAMOPTS command) until the EE XCA major node is deactivated.
- Resetting the HOSTNAME start option (using MODIFY VTAMOPTS,HOSTNAME=NONE) is allowed only when the EE XCA major node is inactive.
- Resetting the IPADDR start option by using MODIFY VTAMOPTS,IPADDR=0.0.0.0, or by using an IPv6 IP address of all zeros, usually written as IPADDR=:, is allowed only when the EE XCA major node is inactive.
- When the MODIFY VTAMOPTS command is used to set the HOSTNAME start option to any value other than NONE, the IPADDR start option is reset, because these start options are mutually exclusive. Message IST1895I will indicate the IPADDR start option has been reset.
- When the MODIFY VTAMOPTS command is used to set the IPADDR start option to any value other than 0.0.0.0 or an IPv6 IP address of all zeros, usually written as ::, the HOSTNAME start option is reset, because these start options are mutually exclusive. Message IST1895I will indicate the HOSTNAME start option has been reset.

**TDUDIAG start option**
Notes:
1 TDUDIAG is meaningful only if the NODETYPE=NN start option is also being used.

dependencies: applies only to APPN topology resources
range: 1 - 65535

Specifies when TDU diagnostic information is included with node or TG control vectors in a topology database update (TDU). This diagnostic information provides the CP name of the network node that updated the resource sequence number (RSN) for a topology resource, which helps determine which network nodes are causing a TDU war. A TDU war is an endless exchange of TDUs in contention over the same topology resource, which results in continuous performance degradation of the APPN network.

**TDUDIAG=resource_threshold**

Specifies a threshold number of times that this network updates the RSN for a node or TG topology resource within a 24-hour period. When this threshold is reached, TDU diagnostic information is appended with the topology control vectors that are included in a TDU for that resource every time the RSN is updated by this node.

**Tip:** The number of RSN updates for a topology resource involved in a TDU war increases rapidly and can reach a very high value before the problem is detected. A low TDUDIAG threshold is not necessary to capture the information that you need to diagnose the TDU war.

**TDUDIAG=ALWAYS**

Specifies that TDU diagnostic information is always appended with the topology control vectors that are included in a TDU for a topology resource when the RSN for that resource is updated by this network node.

**Result:** TDU diagnostic information that is received and sent in TDUs is saved with the node or TG resource in the topology database. Each topology resource can contain up to 50 of the most recent RSN update records, which can result in a significant storage increase if TDUDIAG=ALWAYS is specified. You can clear these RSN update records with the DISPLAY TOPO,LIST=TDUINFO or DISPLAY TOPO,LIST=TDUDIAG commands with the CLEAR operand specified. See the DISPLAY TOPO command in z/OS Communications Server: SNA Operation for additional information.

**TDUDIAG=NEVER**

Specifies that TDU diagnostic information is never appended with the topology control vectors included in a TDU.

**Result:** Not every network node in the network is required to append TDU diagnostic information in TDUs. However, because you cannot anticipate which network nodes are updating the RSNs of the resources that are in contention during a TDU war, you might not be able to diagnose the problem if one of the network nodes involved has TDUDIAG=NEVER specified or does not have support for TDU diagnostic information in TDUs.
**Guideline:** If you use the default value 1000 for the TDUDIAG operand, TDU diagnostic information is not added in a TDU for a resource unless a large number of RSN updates are occurring. A large number of updates might indicate a TDU war. Unless you have a compelling reason to change this value, you should use the default value for the TDUDIAG operand.

You can change the value of TDUDIAG by issuing the MODIFY VTAMOPTS command while VTAM is running.

### TNSTAT start option

**TNSTAT**
- Specifies that tuning statistics should be kept or tuning statistics parameters are to be altered.

**NOTNSTAT**
- Specifies that tuning statistics should not be kept.

The following operands are valid only if you code TNSTAT:

- **CNSL**
  - Specifies that the tuning statistics records are written to the console.

- **NOCNSL**
  - Specifies that tuning statistics records are not to be sent to the system console.

- **TIME=minutes**
  - Specifies the number of minutes that should elapse between records. One record is written every number of minutes for each device that is keeping statistics. A record is also written when a device is deactivated or when a MODIFY TNSTAT command or MODIFY NOTNSTAT command is issued that alters tuning statistics recording for any device or if the interval is altered.

### TOPOTIME start option
Notes:
1. TOPOTIME is meaningful only if the NODETYPE start option is also used.

dependencies: applies only to APPN topology resources
range: 00:01-24:00

Specifies the time of day that APPN topology and routing services (TRS) runs garbage collection of the topology database and deletes expired node and TG records. Garbage collection runs every 24 hours.

The topology.gc_time value must be specified in the format HH:MM and expressed in 24-hour notation. For example, 1:00 p.m. is 13:00. Valid values are in the range 00:01-24:00. The hours (HH) and minutes (MM) values (0-9) must be specified with a leading 0. The hours (HH) value can be in the range 00-24. The minutes (MM) value can be in the range 00-59.

Topology garbage collection does not run within 12 hours of the current VTAM initialization time. For example, if the current initialization time is 08:00 and you set TOPOTIME to 09:00, garbage collection will run at 09:00 the next day.

If TOPOTIME is not specified, topology garbage collection runs every 24 hours after the time that topology and routing services is activated at VTAM initialization.

TOPOTIME is valid only on an APPN node (NODETYPE start option is specified).

In addition to using the DISPLAY NET,VTAMOPTS,OPTION=TOPOTIME command to display the time of day that topology garbage collection will run, you can also display the last time and date of garbage collection with the DISPLAY NET,TOPO,LIST=SUMMARY command.

TRACE for buffer, I/O, NCP line, QDIOSYNC, SIT, SMS, or STATE (with ID) traces
NOTRACE, TYPE = BUF (¹)

TRACE, TYPE = BUF

BUF trace operands (²)

NOTRACE, TYPE = IO (¹)

TRACE, TYPE = IO

IO trace operands (²)

NOTRACE, TYPE = LINE (¹)

TRACE, TYPE = LINE

LINE trace operands (²)

NOTRACE, TYPE = QDIOSYNC (¹)

TRACE, TYPE = QDIOSYNC

QDIOSYNC trace operands (²)

NOTRACE, TYPE = SIT (¹)

TRACE, TYPE = SIT

SIT trace operands (²)

NOTRACE, TYPE = SMS (¹)

TRACE, TYPE = SMS

SMS trace operands (²)

NOTRACE, TYPE = STATE (¹)

TRACE, TYPE = STATE

STATE trace operands (²)

BUF trace operands:

-ID=node_name

-EVERY

-AMOUNT=PARTIAL

-AMOUNT=FULL

-PARTIAL
Chapter 4. VTAM start options  949
STATE trace operands:

```
<table>
<thead>
<tr>
<th>ID= node_name</th>
<th>IDTYPE= RESOURCE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IDTYPE= CP</td>
</tr>
<tr>
<td></td>
<td>IDTYPE= SSCP</td>
</tr>
<tr>
<td></td>
<td>IDTYPE= RESOURCE</td>
</tr>
<tr>
<td>SAVE= YES</td>
<td></td>
</tr>
<tr>
<td>SAVE= NO</td>
<td></td>
</tr>
</tbody>
</table>
```

Note:
1. Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.
2. Code TRACE and its qualifiers on one line. Code the TYPE qualifier immediately after TRACE.
3. COUNT applies only to the IBM 3720 and 3745 Communication Controllers.

Specifies whether VTAM starts or cancels a specific type of trace for a node or line, or monitors the usage of all the VTAM buffer pools.

If you code the TRACE option, code the TYPE qualifier immediately after TRACE.

You can specify a network-qualified name on the ID=node_name operand of the TRACE start option for buffer, I/O, and QDIOSYNC traces.

You can change the value of TRACE with the MODIFY TRACE command while VTAM is running.

AMOUNT
Specifies whether VTAM records part or all of the data that is transmitted in inbound and outbound message buffers.

AMOUNT=PARTIAL
Specifies that VTAM records the data in trace records with a maximum size of 256 bytes. Each trace record contains a trace record header and data. Data that does not fit in the trace record is lost. This operand can be abbreviated as AMT=P.

AMOUNT=FULL
Specifies that VTAM records all of the data transmitted in message buffers. Multiple trace records might be needed to record all of the data. This operand can be abbreviated as AMT=F.

COUNT=number_of_bytes or ALL
The number of bytes of data you want traced. The range is 0–254. If you do not code COUNT, or if you code COUNT=ALL, VTAM traces a maximum of 255 bytes of data in the PIU.

COUNT is valid only if you are running an SIT or LINE trace for a 3720, 3725, or 3745 communication controller.

EVERY
Specifies that VTAM starts or stops the requested trace for the resource named in the ID operand and all appropriate minor nodes of this resource.
You can abbreviate this operand as $E$.

This operand applies only to buffer and I/O traces. For example:

```
TRACE,TYPE=IO,ID=line name,E
```

This start option initiates an I/O trace for the line and each of its minor nodes.

This operand is not valid if ID=ISTPUS or ID=ISTIRN is also coded. If ID names a channel-attachment major node, code the EVERY option. You can name the link in a channel-attachment major node, but you cannot name the link station.

**IDTYPE**

Specifies the type of resource that the ID operand names. In cases where several types of resources share the same name, IDTYPE can be used to identify which resource the start option should act on.

If both a CP and an SSCP have the name specified on the ID operand, then both resources are traced.

IDTYPE is valid only for the buffer contents, I/O, and resource state traces.

**IDTYPE=CP**

Specifies that the name on the ID operand is a CP. Only the trace for a host CP or for the cross-domain resource that represents the adjacent CP is affected by this start option.

**IDTYPE=SSCP**

Specifies that the name on the ID operand is an SSCP. Only the trace for an SSCP is affected by this start option.

**IDTYPE RESOURCE**

 Specifies that the name on the ID operand is one of the following items. VTAM looks for the resource in the order they are listed here:

1. SSCP
2. Host CP
3. Adjacent CP
4. Another resource

**ID=line name**

Names the specific line for which a VTAM trace is started or stopped. $line\ name$ corresponds to the name on the LINE statement that represents the same line.

Each line traced must be explicitly coded in a TRACE start option.

**ID=node name**

Names the specific node for which a VTAM trace is started or stopped. $node\ name$ corresponds to the name assigned to the node and filed in the VTAM definition library. $node\ name$ can be a network-qualified name for the buffer and I/O traces. See the description of the MODIFY TRACE command in [z/OS Communications Server: SNA Operation](https://www.ibm.com/support/knowledgecenter/SSLTBW_22.10.0/com.ibm.zos.bksa.zos_doc/zos_comm_server.html) for a list of the types of nodes you can choose.

For CMIP application programs, $node\ name$ is the name of the internal or external application program. For the VTAM topology agent, $node\ name$ is ISTTOPAG. For notification services, $node\ name$ is ISTNOTIF.

For external CMIP application programs, $node\ name$ is the application name defined as the ACB name of the application program major node.
For a CDRM, you can specify a network-qualified name, but this does not remove the restriction that the non-network-qualified CDRM name must be unique across networks.

You can use ID=VTAM to trace all SSCP activity and, in a multiple-domain network, all cross-domain resource manager activity. You can use ID=ISTPUS or the HOSTPU value to trace all SSCP-PU activity. Each terminal or logical unit traced must be explicitly coded in a TRACE start option.

**ID=trle_name|**

When TYPE=QDIOSYNC is specified, the ID operand specifies the TRLE name of the OSA-Express2 or later adapter for which diagnostic data synchronization and filtering is to be started or stopped. Specify ID= elsewhere to start or stop QDIOSYNC for all TRLEs that define OSA-Express2 or later adapters. When the value ID=* is specified with SAVE=NO, ID=* indicates that the QDIOSYNC command is to be applied to all currently active TRLEs that define OSA-Express2 or later adapters. When the value ID=* is specified with SAVE=YES, ID=* indicates that the QDIOSYNC command is to be applied to all currently active TRLEs that define OSA-Express2 or later adapters and to those that are activated by this VTAM in the future.

When multiple QDIOSYNC NOTRACE or TRACE start options are specified, the last one that is applicable to a specific TRLE name takes precedence. For example, coding TRACE TYPE=QDIOSYNC,ID=TRLE1 requests synchronization for a single OSA-Express2 or later adapter but the request is cancelled if it is followed by NOTRACE TYPE=QDIOSYNC,ID=*.* Coding TRACE TYPE=QDIOSYNC,ID=* requests synchronization of all OSA-Express2 or later adapters and, if it is followed by NOTRACE TYPE=QDIOSYNC,ID=TRLE1, it results in the synchronization of all OSA-Express2 or later adapters except TRLE1.

**NOTRACE**

Specifies that VTAM cancels the trace requested by the TYPE operand. You should code this start option only when you need to override a predefined TRACE start option. Code a separate NOTRACE start option to stop each trace.

**OPTION**

For the value TYPE=QDIOSYNC, the OPTION operand specifies the scope and filter to be applied by the OSA-Express2 or later adapter. The option specifies the devices and the direction for which diagnostic data is to be gathered. You can abbreviate this operand as OPT.

**OPTION=ALLIN**

Indicates that the adapter should gather diagnostic data for all devices, including any that might be controlled by other operating systems. This option also indicates that the adapter should collect records pertinent to events and data flowing from the adapter to the host. Code the value OPTION=ALLIN only when advised by IBM service to do so.

**OPTION=ALLINOUT**

Indicates that the adapter should gather diagnostic data for all devices, including any that might be controlled by other operating systems. This option also indicates that the adapter should collect both inbound and outbound diagnostic data.

**OPTION=ALLOUT**

Indicates that the adapter should gather diagnostic data for all devices, including any that might be controlled by other operating systems. This option also indicates that the adapter should collect records pertinent to
events and data flowing from the host to the adapter. Code OPTION=ALLOUT only when advised by IBM service to do so.

**OPTION=IN**
Indicates that the adapter should gather diagnostic data for devices defined to this VTAM only. This option also indicates that the adapter should collect records pertinent to events and data flowing from the adapter to the host. Code OPTION=IN only when advised by IBM service to do so.

**OPTION=INOUT**
Indicates that the adapter should gather diagnostic data for devices defined to this VTAM only. This option also indicates that the adapter should collect both inbound and outbound diagnostic data. Code OPTION=INOUT only when advised by IBM service to do so.

**OPTION=OUT**
Indicates that the adapter should gather diagnostic data for devices defined to this VTAM only. This option also indicates that the adapter should collect records pertinent to events and data flowing from the host to the adapter. Code OPTION=OUT only when advised by IBM service to do so.

**SAVE**
Applies to TYPE=BUF, TYPE=IO, TYPE=QDIOSYNC, and TYPE=STATE. It specifies whether the trace request is saved for the resource named on the ID operand.

**SAVE=YES**
VTAM saves the trace request for the resource named on the ID operand. If the resource exists when this request is issued, the trace starts immediately. If the resource does not exist when this request is issued, VTAM saves the trace command and starts the trace when the resource is defined.

For TYPE=BUF, TYPE=IO, and TYPE=STATE, if you start a trace with IDTYPE=RESOURCE and a CP or an SSCP (but not both) exists when the command is issued, VTAM starts the trace for the existing resource and saves the trace commands for both resources. If both a CP and an SSCP exist when the command is issued, VTAM starts tracing for them immediately.

For TYPE=QDIOSYNC:
- If you specify ID=trlename, any saved command from a previous TRACE TYPE=QDIOSYNC start option for that TRLE is deleted.
- If you specify ID=*, all saved commands from previous TRACE TYPE=QDIOSYNC start options are deleted.

Use the MODIFY NOTRACE command to delete a saved trace command. VTAM will not delete a saved trace command until you issue a MODIFY NOTRACE command for it, even though the resource might be created and freed or activated and deactivated several times. Saved trace commands are lost when VTAM is halted and restarted.

**SAVE=NO**
Does not save the trace request. If the resource does not exist when you start the trace, the request fails.

**SYNCID**
Optional operand valid for TYPE=QDIOSYNC. The OSA-Express2 or later uses
this value as part of an identifier when it captures diagnostic data. If this operand is not specified, the $trlename$ value is used.

**Restriction:** If you specify a value for SYNCID, it must conform to the rules for names detailed in “Format of definition statements” on page 8.

**TRACE**

Specifies that VTAM starts a specific type of trace for a node or line, or monitors the use of all the VTAM buffer pools. Once started, the trace remains in effect until it is stopped, either by stopping VTAM or by entering the MODIFY NOTRACE command. More than one trace can run concurrently, but code a separate TRACE start option to start each trace.

**TRACEPT=trace_point_id**

Specifies the point in the microcode at which tracing should be activated. TRACEPT applies to TYPE=SIT and is valid only if you are tracing CSS resources on an IBM 3745 Communication Controller. If you omit this operand, tracing is done for all valid trace points. The TRACEPT operand allows you to limit the tracing to a single trace point if too much output is being produced.

VTAM accepts any integer in the range 1–255; however, only a few values are defined by the NCP. For information on which values are defined and what they mean, see NCP, SSP, and EP Trace Analysis Handbook.

**TYPE=BUF**

Specifies whether a VTAM buffer trace is started or stopped.

**TYPE=IO**

Specifies whether a VTAM I/O trace is started or stopped.

**TYPE=LINE**

Specifies that an NCP line trace is started or stopped for a communication controller. There can be up to 8 concurrent line traces and scanner interface traces (SIT) for each NCP, depending on how each NCP is generated.

**TYPE=SIT**

Specifies that a scanner interface trace is started or stopped for a 3720, 3725, or 3745 communication controller. There can be up to 8 concurrent line traces and SITs for each NCP, depending on how each NCP was generated.

**TYPE=QDIOSYNC**

Use TYPE=QDIOSYNC to synchronize and optionally apply a scope and filter to OSA-Express2 or later diagnostic data.

Arming the OSA-Express2 or later directs it to capture diagnostic data when there is an unexpected loss of host connectivity. Data is also captured when the following conditions occur:

1. The VTAM-supplied message processing facility (MPF) exit (IUTLLCMP) is driven.

2. Either the VTAM or TCP/IP functional recovery routine (FRR) is driven with ABEND06F. ABEND06F is the result of a PER trap SLIP command that specifies ACTION=RECOVERY.

   **Restriction:** The SLIP must be a PER trap SLIP in order to specify ACTION=RECOVERY.

See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for more information about using the QDIOSYNC facility.

**TYPE=SMS, ID=VTAMBUF**

Specifies that VTAM monitors the number of its requests to obtain buffers in the VTAM buffer pools. (After a specified number of requests occur, the trace...
creates a record to show how the pools are being used at that time.) If you code TYPE=SMS, also code ID=VTAMBUF.

**TYPE=STATE**

Specifies whether a resource state trace is started or stopped to monitor changes in the state of the resource named on the ID operand.

**TRACE for MODULE, STATE (with OPTION), or VTAM internal trace**

```
NOTRACE, TYPE=MODULE
TRACE, TYPE=MODULE

NOTRACE, TYPE=STATE
TRACE, TYPE=STATE

TRACE, TYPE=VTAM
```

**MODULE operands:**

```
-,OPTION=ALL
```

**STATE trace operands:**

```
-,OPTION=ALL
```

Chapter 4. VTAM start options 955
VIT operands:

1. MODE = INT, SIZE = 4M
2. MODE = EXT, BFRNUM = 2
3. MODE = INT, SIZE = size
4. OPTION = (API, CIO, MSG, NRM, PIU, PSS, SMS, SSCP)
5. SUBTRACE = ARBP, OPTION = HPR
6. SUBTRACE = DIO, OPTION = CIA
7. SUBTRACE = TGVC, OPTION = SSCP
8. OPTION = ALL
9. SUBTRACE = ALL
10. SUBTRACE = NONE

Notes:
1. Do not use NOTRACE when starting VTAM, except to override a TRACE start option coded in a predefined list.
2. You can code TRACE and its qualifiers through position 71, even if you are in...
the middle of the start option. Continue the remainder of the item in the next
record. Code the TYPE qualifier immediately after you code the TRACE start
option.

3 NOTRACE,TYPE=VTAM is accepted but ignored. Tracing is started with the
default trace table size and the default options.

4 The default options apply only to MODE=INT.

5 PSS and SMS can be turned off.

6 When SUBTRACE=ARBP is specified, if a single OPTION value is coded, it
must be HPR, ALL, or one of the group options (hprgrpopt) that include HPR
as an individual option equivalent. The applicable group options are
DLUOPTS, EEOPTS, HPDTOPTS, HPROPTS, QDIOOPTS, and XCOPTS.

7 If multiple trace options are coded in parentheses, either HPR or one of the
group options (hprgrpopt) that include HPR as an individual option equivalent
must be coded inside the parentheses when SUBTRACE=ARBP is coded.

8 When you specify SUBTRACE=DIO and you code a single OPTION value,
the OPTION value must be CIA, ALL, or one of the group options (ciagrpopt)
that include CIA as an individual option equivalent. The applicable group
options are EEOPTS, HPDTOPTS, HPROPTS, QDIOOPTS, TCPOPTS and
XCOPTS.

9 When SUBTRACE=DIO is coded and you code multiple trace options in
parentheses, you must code either CIA or one of the group options (ciagrpopt)
that include CIA as an individual option equivalent inside the parentheses.

10 When SUBTRACE=TGVC or SUBTRACE=TREE is coded, if a single OPTION
value is coded, it must be SSCP, ALL, or one of the group options (groupopt),
all of which include SSCP as an individual option equivalent. The group
options are APIOPTS, APPCOPTS, CPCPOPTS, CSMOPTS, DLUOPTS,
EEOPTS, HPDTOPTS, HPROPTS, LCSOPTS, QDIOOPTS, STDOPTS,
TCPOPTS, and XCOPTS.

11 If multiple trace options are coded in parentheses, either SSCP or one of the
group options (groupopt) must be coded inside the parentheses when
SUBTRACE=TGVC or SUBTRACE=TREE is coded.

Specifies whether VTAM starts or cancels a specific type of trace.

If you code the TRACE option, code the TYPE qualifier immediately after TRACE.

You can change the value of TRACE with the MODIFY TRACE command while
VTAM is running.

**BFRNUM=number**

Specifies the number of 8K external trace buffers the VTAM internal trace is to
allocate and use for generalized trace facility (GTF) processing. Integers 0 or
2–50 can be specified. The default is 2. Two times the number of processors in
the central processing unit (CPU) should ensure that enough buffers are
available.

When an integer of 2–50 is specified, VTAM accumulates approximately 8K of
external trace data prior to sending the data to GTF.

If zero is specified or there is no buffer available for the trace record, VTAM
sends each trace record to GTF as it is recorded. This incurs a significant
system overhead, and should be avoided.
MODE
Specifies that the VTAM internal trace is to record its data on an internal, wraparound table (MODE=INT) and optionally, on an external trace file (MODE=EXT).

MODE applies only to TYPE=VTAM.

You can record trace data internally and externally simultaneously. If you want, you can have different sets of trace options active for internal and external recording. VTAM always runs with MODE=INT and the default trace options, whether you request tracing or not.

You must run specific operating system utilities to trap, format, and view external trace output. See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for more information about use of these operating system utilities.

Do not specify MODE=EXT and SIZE on the same command.

MODE=INT
Specifies that the VTAM internal trace is to record its data on an internal, wraparound table.

MODE=EXT
Specifies that the VTAM internal trace is to record its data on an external trace file as well as on an internal, wraparound table. You can record external trace data using the generalized trace facility (GTF). GTF must be active when you initiate traces. You can format output using IPCS.

See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for a more detailed description of formatting and printing trace output.

NOTRACE
Specifies that VTAM cancels the trace requested by the TYPE operand. Code this start option only when it is necessary to override a predefined TRACE start option. Code a separate NOTRACE start option to stop each trace.

NOTRACE,TYPE=VTAM
Accepted but ignored. Tracing is started with the default trace table size and the default options.

OPTION=(option,option,...option)
Applies to TYPE=MODULE, TYPE=STATE and TYPE=VTAM.

You can abbreviate this operand as OPT.

You can code one or more of the options listed for each TYPE operand.

Note: Each trace option can affect the performance of your host and the VTAM network. Be sure to understand the effect this function can have on performance in your environment before specifying OPTION=ALL.

For TYPE=MODULE, OPTION specifies the modules for which tracing is to be started.

For TYPE=STATE, OPTION specifies the types of resources for which resource states are to be recorded. The data is recorded using the mode (internal or external) specified for the SSCP VIT option.

For TYPE=VTAM, OPTION is a listing of the VTAM functions that you want to trace.
When VTAM encounters an exception condition involving an API, NRM, PIU, SSCP, or SMS function, it generates a trace record whether or not you have activated that option.

Note: Although the default options are always active for internal recording, these options do not appear in DISPLAY TRACES output unless you have specified them on the MODIFY TRACE command or the TRACE,TYPE=VTAM start option.

**ALL**
Applies to TYPE=MODULE, TYPE=STATE and TYPE=VTAM.
For **TYPE=MODULE**, OPTION=ALL starts the tracing of all the modules shown on the OPTION operand for which TYPE=MODULE apply.
For **TYPE=STATE**, OPTION=ALL starts the tracing of resource states for all of the resource types shown on the OPTION operand for which TYPE=STATE apply.
For **TYPE=VTAM**, OPTION=ALL specifies that the VTAM internal trace (VIT) is to be started for all of the VTAM internal functions for which the VIT is available, including VCNS trace functions. It is equivalent to specifying all of the internal trace types. Exception trace entries continue to be recorded.

For **TYPE=MODULE**, you can also specify the following options:

**OPTION=COMMAND**
Starts tracing modules involved in command processing.

**OPTION=CONNECTIONS**
Starts tracing modules involved in setting up connections between nodes.

**OPTION=DEFINITION**
Starts tracing modules involved in resource definition processing.

**OPTION=INTERFACES**
Starts tracing modules involved in the interface with the host SSCP or the host CP.

**OPTION=MANAGEMENT**
Starts tracing modules involved in network management.

**OPTION=NOEXIT**
Specifies that module exits are not traced for modules associated with other OPTION values for TYPE=MODULE. Module exits are not traced for any modules until a subsequent MODIFY NOTRACE,TYPE=MODULE,OPTION=NOEXIT command is issued.
Starting a module trace for any OPTION with TYPE=MODULE starts the tracing of the module exits also, unless you specify OPTION=NOEXIT.

**OPTION=SESSION**
Starts tracing modules involved in session establishment.

For **TYPE=STATE**, you can also specify the following options:

**OPTION=ADJCP**
Starts tracing the states of all adjacent control points.

**OPTION=APPL**
Starts tracing the states of all application programs.
OPTION=CDRM
Starts tracing the states of all CDRMs.

OPTION=CDRSC
Starts tracing the states of all CDRSCs.

OPTION=GROUP
Starts tracing the states of all line groups.

OPTION=LINE
Starts tracing the states of all lines.

OPTION=LU
Starts tracing the states of all logical units.

OPTION=NCP
Starts tracing the states of all NCPs.

OPTION=PU
Starts tracing the states of all physical units.

For TYPE=VTAM, you can also specify the following options:

API
Helps you determine whether an application program is causing a problem. API entries are written for RPL macros, RPL exit routines, user exit routines, and user posts.

APIOPTS
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose potential non-LU 6.2 application program problems. Specifying APIOPTS is equivalent to specifying all of the following VIT options: API, MSG, NRM, PIU, PSS, SMS, and SSCP.

APPC
Helps you determine whether an LU 6.2 application is causing a problem. LU 6.2 entries are written for the following items:
- APPCCMD macro invocations
- User posts and exit scheduling by LU 6.2 code
- Calls to a security manager for security processing
- Message unit transmissions between LU 6.2 components

APPCOPTS
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose potential LU 6.2 application program problems. Specifying APPCOPTS is equivalent to specifying all of the following VIT options: API, APPC, MSG, NRM, PIU, PSS, SMS, and SSCP.

CFS
Helps you assess and manage your coupling facility structure.

CIA
Helps isolate problems indirectly related to channel IO by tracing additional data near the time of channel IO activity.

CIO
Helps isolate problems related directly to channel I/O (CIO). CIO entries are written for attentions, error recovery, interruptions, HALT I/O SVC, and START I/O SVC.
**CMIP**
Indicates whether a CMIP problem is in VTAM or in a CMIP application program.

This option traces:
- Calls from CMIP application programs to the management information base (MIB) application programming interface
- Calls to the read-queue exit of the CMIP application program
- Topology updates from VTAM resources

**CPCPOPTS**
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose potential CP-CP session problems. Specifying CPCPOPTS is equivalent to specifying all of the following VIT options: API, APPC, MSG, NRM, PIU, PSS, SMS, and SSCP.

**CSM**
Traces the parameter list information that flows across the communications storage manager (CSM) interface for the following events:
- Requests to obtain a buffer
- Requests to free a buffer
- Requests to assign additional tokens to the buffer
- Requests to change the ownership of a buffer
- Requests to change the pageable state of a buffer
- Requests to copy data to or from a CSM buffer or a user data area
- Buffer pool expansion
- Buffer pool contraction

*Note:* For installations using high performance data transfer (HPDT), the use of the CSM trace should be restricted to times when diagnostic information is needed. Using this option on a regular basis minimizes the performance benefits provided by HPDT services.

**CSMOPTS**
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose potential communications storage manager (CSM) problems. Specifying CSMOPTS is equivalent to specifying all of the following VIT options: API, APPC, CIO, CSM, MSG, NRM, PIU, PSS, SMS, SSCP, and XBUF.

**DLUROPTS**
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose dependent LU requester (DLUR) problems. Specifying DLUROPTS is equivalent to specifying all of the following VIT options: API, APPC, HPR, MSG, NRM, PIU, PSS, SMS, and SSCP.

**EEOPTS**
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose Enterprise Extender (EE) problems. Specifying EEOPTS is equivalent to specifying all of the following VIT options: CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, SSCP, and TCP.

**ESC**
Helps you track, in detail, the flow of requests for a given process.
HPDTOPTS
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose high performance data transfer (HPDT) problems. Specifying HPDTOPTS is equivalent to specifying all of the following VIT options: CIA, CIO, HPR, MSG, PIU, PSS, SMS, and SSCP.

HPR
Helps you isolate problems related to High-Performance Routing (HPR). Events include invocation of the HPRCTL macroinstruction, invocation of the HPR timer control macroinstruction, any LU-LU session data sent across a rapid transport protocol (RTP) connection, a network layer packet (NLP) sent or received, a dispatch of the RTP context manager (RCM) PAB, a dispatch of a RTP PAB, an error detected by RTP, and a route setup sent or received.

HPROPTS
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose High-Performance Routing (HPR) problems. Specifying HPROPTS is equivalent to specifying all of the following VIT options: API, APPC, CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, and SSCP.

LCS
Helps you isolate problems occurring during data transfers from an IBM 3172 Interconnect Controller to VTAM. The LCS option enables tracing of data that VTAM receives from an IBM 3172 Interconnect Controller at four levels: LCSX (cross-channel), LCSP (port or adapter), LCSL (service access point of the adapter) and LCSL (line).

LCSOPTS
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose LAN channel station (LCS) problems. Specifying LCSOPTS is equivalent to specifying all of the following VIT options: CIA, LCS, MSG, NRM, PIU, PSS, SMS, and SSCP.

LOCK
Helps you determine when VTAM modules get and release locks.

MSG
Helps you correlate other VIT internal entries with the console messages even if the console sheet is lost. MSG entries are written for all messages to the VTAM operator.

NONE
Specifies that no traces are active.

NRM
Helps you follow the services of the network resource management component. These include the assignment of, references to, and the deletion of certain VTAM resources such as node names, network addresses, and control blocks. NRM entries are written for all I/O and buffer contents traces.

PIU
Helps you isolate problems to hardware, to the NCP, or to VTAM. PIU entries are written for all I/O to and from VTAM, which can be an advantage over I/O and buffer content traces.

PSS
Helps you track the flow of requests through VTAM. PSS entries are
written for the VTAM macros that invoke and control PSS, and for scheduling and dispatching VTAM routines.

QDIOOPTS
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose queued direct I/O (QDIO) problems. Specifying QDIOOPTS is equivalent to specifying all of the following VIT options: CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, and SSCP.

SMS
Helps you isolate problems caused by storage shortages. When used with the SSCP or PSS trace options, it can also help you isolate internal VTAM problems. SMS entries are written when SMS macros are used to request or release fixed-length or variable-length buffers. SMS entries are also written when VTAM expands or attempts to expand a buffer pool.

SSCP
Helps you isolate problems to a specific VTAM component or module. SSCP entries are written for the request or response units (RUs) sent between VTAM components.

STDOPTS
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose problems related to high CPU, session services, storage, Open/Close ACB, and DLCs such as MPC and CTC. Specifying STDOPTS is equivalent to specifying all of the following VIT options: API, CIO, MSG, NRM, PIU, PSS, SMS, and SSCP.

TCP
Helps you trace Enterprise Extender events.

TCPOPTS
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose problems related to TCP/IP. Specifying TCPOPTS is equivalent to specifying all of the following VIT options: CIA, CIO, MSG, NRM, PIU, PSS, SMS, SSCP, and TCP.

VCNS
Helps you determine whether a VCNS application is causing a problem. VCNS entries are written for VCNSCMD macro invocations, user posts, and exit scheduling by VCNS code, and work element transmissions between VCNS components.

XBUF
Records information contained within the extended buffer list, particularly in regard to information on CSM usage by VTAM. This option traces the contents from the application supplied extended buffer list as well as the internal buffer list that VTAM uses to carry the extended buffer list information.

Note: For installations using high performance data transfer (HPDT), the use of the XBUF trace should be restricted to times when diagnostic information is needed. Using this option on a regular basis minimizes the performance benefits provided by HPDT services.

XCF
Helps you trace XCF communication.

XCFOPTS
A grouping of multiple VIT options that includes all of the individual VIT options usually required to diagnose cross-system coupling facility (XCF)
problems. Specifying XCFOPTS is equivalent to specifying all of the following VIT options: CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, SSCP, and XCF.

**SIZE=**size

Applies only when you specify TYPE=VTAM,MODE=INT. The size operand specifies the number of megabytes to be allocated for the internal trace table. Valid values are in the range 4M - 2048M. The VTAM internal trace table is allocated in 64-bit common (HVCOMMON) storage.

**Restriction:** If you specify a SIZE value that is larger than the default value, z/OS performs paging on portions of the VIT table. Before you specify a large SIZE value, ensure that you have sufficient real or auxiliary storage to contain the entire VIT. Failure to ensure sufficient storage might result in an auxiliary storage shortage. If an SVC dump is taken that includes common storage, the size of the dump data set also increases. You must also take the increase in the size of the dump data set into consideration.

**SUBTRACE**

Specifies that SUBTRACE can be used to turn on a subset of trace entries under a trace option. Of the SUBTRACE types defined, subtrace TREE and TGVC are defined under the SSCP trace option, subtrace ARBP is defined under the HPR trace option, and subtrace DIO is defined under the CIA trace option.

**Recommendation:** All of the SUBTRACE options are defaulted to off. They can generate a large number of records in the VTAM trace and can incur a significant overhead, but may be necessary in some cases for diagnostic purposes. It is not recommended to activate them at VTAM start time. If used, the SUBTRACE options should be turned off when the necessary trace output has been obtained.

**SUBTRACE=ARBP**

Specifies the ARB (Adaptive Rate Based) algorithm performance trace records that are to be generated. The OPTION operand is required when SUBTRACE is specified and HPR must be one of the trace options specified when SUBTRACE=ARBP is coded. After subtrace ARBP is activated, the following trace records will be generated for the ARB algorithm processing: ARBR (Generated when ARB Responsive Mode algorithm is used) and ARBB (Generated when ARB Base Mode algorithm is used).

**SUBTRACE=DIO**

Specifies that Direct IO related trace records are to be generated. The OPTION operand is required when SUBTRACE is specified and CIA must be one of the trace options specified when SUBTRACE=DIO is coded. After subtrace DIO is activated, the following trace records may be generated for QDIO and Hipersockets processing: QAPL, QDIP and QSRB.

**SUBTRACE=TGVC**

Specifies that APPN transmission group vector (TGVC) trace records are to be generated. The OPTION operand is a required keyword when SUBTRACE is specified and SSCP must be one of the trace options specified when SUBTRACE=TGVC is coded. After subtrace TGVC is activated, the following trace records will be generated for various TG Vector requests: TGVC, TGV2. If large amounts of data are being traced, additional TGVC records (plus subsequent TGV2 records) may occur.

**SUBTRACE=TREE**

Specifies the APPN routing tree trace records that are to be generated. The
OPTION operand is required when SUBTRACE is specified and SSCP must be one of the trace options specified when SUBTRACE=TREE is coded. After subtrace TREE is activated, the following trace records will be generated for routing trees used by APPN route computation: TRRT, TRR2, TRR3, TRR4, TRR5, HLST, and HLS2.

**TYPE=MODULE**

Specifies whether a module trace is started or stopped for the types of modules listed on the OPTION operand.

**TYPE=STATE**

Specifies whether a resource state trace is started or stopped to monitor changes in the state of the type of resources listed on the OPTION operand.

**TYPE=VTAM**

Specifies the VTAM internal trace (VIT).

Use IBM Health Checker for z/OS to check whether various VTAM internal trace parameters have been set to values that reflect best practices. For more details about IBM Health Checker for z/OS, see z/OS Communications Server: IP Diagnosis Guide.

**TRANSLAT start option**

Controls alias name translation. It specifies which names will be translated, but not how the names will be translated. If TRANSLAT is specified, only the bits corresponding to the values specified will be turned on. All other bits will be turned off. For example, if you specify

TRANSLAT=(OLUALIAS,DLUSSCP,LOGMODES)

the hexadecimal value will be X'68' (equivalent to B'10101000'). If you specify TRANSLAT=(), all bits will be turned off.

**TRANSLAT=OLUALIAS or 0 (bit 0)**

Specifies that VTAM is to determine the originating logical unit (OLU) alias names.
TRANSLAT=DLUREAL or 1 (bit 1)
Specifies that VTAM is to determine destination logical unit (DLU) real names.

TRANSLAT=DLUSSCP or 2 (bit 2)
Specifies that VTAM is to determine the DLU’s owning SSCP name.

TRANSLAT=COSNAME or 3 (bit 3)
Specifies that VTAM is to determine Class of Service names.

TRANSLAT=LOGMODES or 4 (bit 4)
Specifies that VTAM is to determine logon mode names.

TRANSLAT=LUANAME or 5 (bit 5)
Specifies that VTAM is to determine associated LU alias names.

TRANSLAT=USERVAR or 6 (bit 6)
Specifies that VTAM is to enable the USERVAR function.

TRANSLAT=DLUALIAS or 7 (bit 7)
Specifies that VTAM is to determine DLU alias names only when needed.

**UNRCHTIM start option**

![Diagram](image)

Notes:

1. UNRCHTIM is meaningful only if the NODETYPE start option is also used.

   range: unreachable_time: 0, or 10-65535
   partner_limit: 5-65535

Specifies the options associated with Enterprise Extender connection network reachability awareness.

There are three allowed formats for coding start option UNRCHTIM:

- UNRCHTIM=unreachable_time
- UNRCHTIM=(unreachable_time ,partner_limit)
- UNRCHTIM=(partner_limit)

**unreachable_time**
Specifies the default number of seconds that a partner node for a session path through an Enterprise Extender connection network is considered unreachable after certain Enterprise Extender connection network failures. During this period, the path through the Enterprise Extender connection network to this partner node is not considered for new sessions or HPR path switches.

**partner_limit**
Specifies the limit for the number of unreachability records that can be associated with an Enterprise Extender connection network before that connection network is considered unusable for session route selection. No paths through the Enterprise Extender connection network will be considered for new sessions or HPR path switches until the number of unreachability
records falls below 80% of the partner_limit value. The number of unreachability records associated with an Enterprise Extender connection network virtual routing node (VRN) can be displayed with a DISPLAY TOPO, LIST=UNRCHTIM, VRN=virtual_node command.

**Result:** Unreachability records are not maintained on an end node. They are maintained only by the end node’s network node server. Therefore, the partner_limit value is ignored if specified on an end node.

A partner node is determined to be unreachable when a dial failure occurs across an Enterprise Extender connection network or when an INOP of an existing connection through an Enterprise Extender connection network is detected. UNRCHTIM is meaningful only if the NODETYPE start option is also used.

UNRCHTIM is meaningful only for Enterprise Extender connection networks. You can override the UNRCHTIM unreachable_time start option value for individual Enterprise Extender connection networks by coding UNRCHTIM=unreachable_time on the PORT or GROUP statement that defines each connection network in the Enterprise Extender External Communications Adapter (XCA) major node. The UNRCHTIM partner_limit value cannot be coded on the PORT or GROUP statements. See z/OS Communications Server: SNA Network Implementation Guide for additional information about performance considerations with UNRCHTIM.

**Note:**
1. If the UNRCHTIM start option is not specified, the default unreachable_time value is 0, which indicates that paths through Enterprise Extender connection networks will always be considered for routing.
2. A minimum (nonzero) value of 10 seconds is set for the UNRCHTIM unreachable_time value to reduce the risk of performance concerns because of excess TDU flows.
3. On an end node, instead of displaying the value configured, NO is displayed if UNRCHTIM=0 is configured and YES is displayed if a nonzero value is configured.

You can change the UNRCHTIM start option values with the MODIFY VTAMOPTS command while VTAM is running. The unreachable_time value will be changed for the PORT statement and all GROUP statements that define connection networks in an XCA major node that do not override the start option value. The unreachable time for existing paths through connection networks to partner nodes that were determined to be unreachable before the start option value was modified will not be changed. When the existing unreachable time for a path expires, the new unreachable_time value will be used if the partner node through that path is determined to be still unreachable. The new partner_limit value will be used for all routes selected for new sessions and HPR path switches after the partner_limit value is changed.

**UPDDELAY start option**
Notes:
1. UPDDELAY is meaningful only if the OSIMGMT=YES start option is also used.

range: 5–65535 seconds (or equivalent value in minutes or hours)

Specifies the maximum time the VTAM topology agent waits between checking the list of resources in transient states (such as PENDING ACTIVE and PENDING INACTIVE) to determine whether the resources have been in transient states long enough to be sent to the topology manager. If a resource reaches resting state before the interval set by UPDDELAY has expired, the update for that resource is still sent.

By setting a longer interval on the UPDDELAY start option, you can reduce the CPU cycles required to send updates to the topology manager. By setting a shorter interval, you can send updates to the topology manager more frequently. Frequent sending of updates increases the opportunity to discover resources that are hung in transient states.

You can change the value of UPDDELAY with the MODIFY VTAMOPTS command while VTAM is running. If you change the value, the new value takes effect after the time specified on the original value has expired. For example, if the original value for UPDDELAY is 1 hour and you specify 5 seconds for the UPDDELAY start option on the MODIFY VTAMOPTS command, the new value of 5 seconds takes effect after the hour has expired.

The value for UPDDELAY can be expressed as:

\( n \text{ or } nS \)

Specifies the maximum number of seconds in the range 5–65535.

\( nM \)

Specifies the maximum number of minutes in the range 1–1092.

\( nH \)

Specifies the maximum number of hours in the range 1–18.

**USSTAB start option**
Specifies the name of a USS table to be used for VTAM operator messages and for VTAM operator commands that are processed through USS. `table_name` is the 8-character name of a USS table. Choose one of the following USS tables:

- ISTINCNO, which contains the IBM-supplied VTAM messages and commands. ISTINCNO is the default for IBM-supplied VTAM messages. This table can be supplemented with a user-defined message table. When a user-defined USS command table is specified on the USSTAB start option, the user-defined table is the only table searched to process a VTAM command.

- ISTCFCMM contains a master copy of the original messages in ISTINCNO. Choose this table name if you want to continue using these original messages in program operator applications after ISTINCNO is modified. Commands processed through USS continue to use ISTINCNO.

- A user-defined table that contains user-modified messages and commands. Messages not defined in the user-defined table continue to use ISTINCNO; commands not defined in the user-defined table will not be found. The user-defined table should not be named either ISTINCNO or ISTINCDT; otherwise, you might not be able to use all of the defined USS commands and messages.

If the attempt to load the specified table fails, VTAM uses ISTINCNO by default. Message IST116I is issued if the specified table is not found; message IST448I is issued if storage is insufficient for loading the table.

**VARYWLD start option**
Specifies whether wildcards are permitted in certain VARY commands.

You can change the value of VARYWLD with the MODIFY VTAMOPTS command while VTAM is running.

**VARYWLD=FULLWILD**

Specifies that wildcards are permitted in VARY commands from all network operators. Wildcards are permitted in VARY commands from program operator applications whose APPL definition statement indicates VARYWLD=YES.

**VARYWLD=NOWILD**

Specifies that wildcards are not permitted in any VARY commands.

**VARYWLD=OPERONLY**

Specifies that wildcards are permitted in VARY commands from the network operator, but not from program operator applications.

**VARYWLD=POAONLY**

Specifies that wildcards are permitted in VARY commands from program operator applications whose APPL definition statement indicates VARYWLD=YES. Wildcards are not permitted from the network operator.

**VERIFYCP start option**
Notes:

1. The VERIFYCP start option is meaningful only if the NODETYPE start option is also used.

Specifies whether VTAM performs session-level verification during activation of LU 6.2 sessions involving control points. The related SECLVLCP start option values specify the type of verification.

If you do not code SECLVLCP, VTAM attempts to use the enhanced protocol if VERIFYCP=OPTIONAL or VERIFYCP=REQUIRED is specified, but will accept basic protocol.

If you code VERIFYCP=OPTIONAL or VERIFYCP=REQUIRED, a security management product, such as RACF 1.9 or later, must be installed and active.

**Note:** When performing LU verification for the VTAM CP, the RACF profile must be created using four-part names. See z/OS Communications Server: SNA Network Implementation Guide for an example of how to create the RACF profile.

**VERIFYCP=NONE**

Specifies that there is no verification of the partner CP’s identity during session activation.

**VERIFYCP=OPTIONAL**

Specifies that identity verification is done for certain partner CPs during session activation. Identity verification is done when there is a password defined for the CP partners in the installed security manager product. See z/OS Communications Server: SNA Network Implementation Guide for information about how VTAM determines when to perform identity verification.

**VERIFYCP=REQUIRED**

Specifies that VTAM verifies the identity of all partner CPs during activation of sessions between LU 6.2 applications. Every partner CP must have a password defined. Any partner CPs that do not have a password defined cannot establish LU 6.2 sessions with this application program.

**VFYRED start option**
VFYRED = YES

Indicates that LU 6.2 session initiation requests do not need to be delivered to the target LU.

Note: If the BIND fails, VTAM reissues the session initiation request indicating that the resource must be verified.

VFYRED = NO

Indicates that the session initiation request must be delivered to the target LU. Normal session setup logic will be performed.

VFYREDTI start option
Notes:
1 VFYREDTI is meaningful only if the NODETYPE=NN start option is also used.

dependencies: LU 6.2 application programs using the APPCCMD API
range: 1–2147483647 seconds (or equivalent minutes or hours)

Specifies the maximum amount of time a resource's location is not verified during session setup.

You can change the value of VFYREDTI with the MODIFY VTAMOPTS command while VTAM is running.

**VFYREDTI=OFF**
Indicates that there is no node-wide verification reduction timer value. Verification reduction will be attempted for the time indicated by the target resource's cached information. If the target resource has no cache time value VTAM will assume a time value of 1 day.

**VFYREDTI=0**
Indicates that all requests must be verified.

**VFYREDTI=reduction_timer**
Specifies the amount of time resource verification reduction is to be performed before the resource is verified. The lesser of this value and the target resource's cache time value (or the 1 day default) is used.

*reduction_timer* can be specified in seconds (S), minutes (M), or hours (H), where n is an integer:

- **n or nS**
  Specifies the number of seconds resource verification reduction is to be performed before the resource is verified. Seconds is the default interval.

- **nM**
  Specifies the number of minutes resource verification reduction is to be performed before the resource is verified.

- **nH**
  Specifies the number of hours resource verification reduction is to be performed before the resource is verified.

---

**VOSDEACT start option**
The VOSDEACT keyword can be coded in the VTAM start list, or turned on/off by way of the MODIFY VTAMOPTS command, to cause VTAM to bring a virtual route (VR) down if the VR goes out-of-sequence.

**VOSDEACT= YES**  
Specifies that VR deactivation will be initiated when a VR goes out-of-sequence.

**VOSDEACT= NO**  
Specifies that VR deactivation will not be initiated when a VR goes out-of-sequence. VOSDEACT=NO continues the previous mode of virtual route processing.

### VRTG start option

**VRTG= NO**  
Indicates that VR-based transmission group connection is not requested when SSCP-SSCP sessions are established.

**VRTG= YES**  
Indicates that VR-based transmission group connection is requested when SSCP-SSCP sessions are established.

Notes:

1. VRTG is meaningful only if the NODETYPE and HOSTSA start options are also used.

Indicates whether VR-based transmission group connections are to be requested when the SSCP-SSCP session is established for this node.

This option is valid only for interchange nodes and migration data hosts when start options NODETYPE and HOSTSA are specified.

**Note:** This value can be overridden by coding the VRTG operand in the CDRM definition statement.

You can change the value of VRTG with the MODIFY VTAMOPTS command while VTAM is running.

**VRTG= YES**  
Indicates that VR-based transmission group connection is requested when SSCP-SSCP sessions are established.

**VRTG= NO**  
Indicates that virtual-route-based transmission group connection is not requested when SSCP-SSCP sessions are established.

### VRTGCPCP start option
Notes:
1. VRTGCPCP is meaningful only if the NODETYPE and HOSTSA start options are also used.

dependencies: VRTG=YES coded

Indicates whether CP-CP sessions are supported over the virtual-route-based transmission group.

This option is valid only for interchange nodes and migration data hosts when start options NODETYPE and HOSTSA are specified.

Note: This value can be overridden by coding the VRTGCPCP operand in the CDRM definition statement.

You can change the value of VRTGCPCP with the MODIFY VTAMOPTS command while VTAM is running.

**VRTGCPCP=**YES
  Indicates that CP-CP sessions are supported over the VR-based transmission group.

**VRTGCPCP=**NO
  Indicates that CP-CP sessions are not supported over the VR-based transmission group.

VTAMEAS start option
VTAMEAS = number_of_concurrent_sessions

range: 0-2147483647

Specifies the number of concurrent sessions VTAM can have with other LUs.

**XCFGRPID start option**

XCFGRPID = group_id

dependencies: z/OS; applies only to VTAMs defined in a sysplex environment

Specifies the subplex group ID suffix that is to be used for this VTAM. The group_id value is a 1-digit or 2-digit numeric value. The value specified must be in the range 2 - 31. This value is used to modify the sysplex group names that VTAM joins in the sysplex.

If XCFGRPID=vv is specified, VTAM will join the sysplex groups ISTXCFvv and ISTCFSvv. In addition, the structure names for generic resources and MNPS (which can be specified in the STRGR and STRMNPS start options) will be suffixed with the XCFGRPID value. This value is also used in combination with the TCP XCFGRPID value (specified on the GLOBALCONFIG statement in the TCP/IP profile) to modify the name of the sysplex group that TCP/IP joins. If GLOBALCONFIG XCFGRPID tt is specified in the TCP/IP profile, then the TCP/IP stack joins sysplex group EZBTvott. In addition, if SWSA or Sysplexports Coupling Facility structures are used, the structure names are EZBDVIPAvott and EZBEPORTvott, respectively.

If no VTAM XCFGRPID option is specified, VTAM joins groups ISTXCF and ISTCFS01, and the generic resource and MNPS structures are not suffixed. If a TCP/IP XCF group ID is specified, then the TCP/IP stack joins groups EXBTCPtt, and the structure names for SWSA and sysplexports are EZBDVIPA01tt and EZBEPORT01tt.

**Guideline:** For MNPS structures, ensure that the suffixed structure name in one node cannot be mistaken as an alternate MNPS structure of an MNPS base structure in another node. For example, if one VTAM node in the sysplex is started with start options specifying XCFGRPID=11 and STRMNPS=ISTMYMNPS and another VTAM node is started with STRMNPS=ISTMYMNPS and no XCFGRPID start option is specified, then the structure name generated for the MNPS structure in the first VTAM node (ISTMYMNPS11) will appear to be an alternate structure for the MNPS base structure (ISTMYMNPS) in the second VTAM node.

**XCFINIT start option**
Specifies whether XCF connections or XCF definitions for connections to other VTAMs in a sysplex are to be established at VTAM initialization.

**XCFINIT=NO**

Specifies that this VTAM node will not join an ISTXCF sysplex group at VTAM initialization and will not participate in establishing XCF connectivity (SNA or TCP/IP) with other VTAM nodes in the sysplex.

**XCFINIT=YES**

Specifies that this VTAM node will join an ISTXCF sysplex group at VTAM initialization. The name of the group will be ISTXCFvv, where vv is the 2-character XCF group ID representing the numeric value specified on the XCFGRPID start option. If no XCFGRPID start option was specified, the group name will be ISTXCF. VTAM will create definitions for XCF connectivity to other VTAM nodes in the sysplex as this node becomes aware of their presence in the sysplex group. The node will activate SNA connectivity to these other nodes immediately. TCP/IP connectivity between these nodes will be enabled. XCFINIT=YES is valid only for APPN nodes (that is, nodes that specify the NODETYPE start option). It is the default for APPN nodes.

**Rule:** HPR=RTP is required to activate SNA XCF connectivity between VTAM nodes. If XCFINIT=YES is specified and HPR is not RTP, the value for XCFINIT will be forced to DEFINE.

**XCFINIT=DEFINE**

Specifies that this VTAM node will join the sysplex group at VTAM initialization. The name of the group will be ISTXCFvv, where vv is the 2-character XCF group ID representing the numeric value specified on the XCFGRPID start option. If no XCFGRPID start option was specified, the group name will be ISTXCF. VTAM will create definitions for XCF connectivity to other VTAM nodes in the sysplex as this node becomes aware of their presence in the sysplex group. It will not activate SNA connectivity to these other nodes immediately. For an APPN node, SNA connectivity can be activated at a later time by issuing a VARY ACTIVATE command for the XCF SNA APPN PU. For a pure subarea node, SNA connectivity is not allowed through XCF. TCP/IP connectivity between these nodes will be enabled. XCFINIT=DEFINE is the default for pure subarea nodes.

**Guideline:** Specify this option when you want XCF connectivity for TCP/IP communications within the sysplex but the use of XCF connectivity for APPN communications is not required at VTAM initialization.
XNETALS start option

![Diagram showing the relationship between XNETALS and NETID on PU statement with YES and NO options]

depends on:

NETID on PU statement

Determines whether the network ID of an attaching node or VTAM's network ID is used for an adjacent nonnative node. Used to control how a nonnative node can attach to VTAM.

**Note:** The XNETALS option has changed in the way VTAM treats PUs that send in nonnative NETIDs in their XID. VTAM used to let the PU connect and treated it as if it sent in the same NETID that VTAM started with but that is no longer true.

**Note:** Coding XNETALS=NO and GWCAP=NO will prevent subarea searches of the node in which the options are coded when the DLU NETID is not the NETID of the host. If YES is coded for either option, subarea searches will be performed for the host in which the options were coded.

The following matrix shows the resulting XNETALS values when XNETALS and CONNTYPE are coded for various levels:

**Table 71. XNETALS matrix**

<table>
<thead>
<tr>
<th>Start list values</th>
<th>Specified on GROUP, LINE, or PU (lowest level wins)</th>
<th>CONNTYPE</th>
<th>XNETALS and result for the PU</th>
</tr>
</thead>
<tbody>
<tr>
<td>YES</td>
<td>YES</td>
<td>APPN</td>
<td>YES</td>
</tr>
<tr>
<td>YES</td>
<td>YES</td>
<td>LEN</td>
<td>YES</td>
</tr>
<tr>
<td>YES</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES</td>
<td>NO</td>
<td>APPN</td>
<td>NO</td>
</tr>
<tr>
<td>YES</td>
<td>NO</td>
<td>LEN</td>
<td>NO</td>
</tr>
<tr>
<td>YES</td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES</td>
<td>APPN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES</td>
<td>LEN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td>YES</td>
<td>APPN</td>
<td>YES</td>
</tr>
<tr>
<td>NO</td>
<td>YES</td>
<td>LEN</td>
<td>YES</td>
</tr>
<tr>
<td>NO</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td>NO</td>
<td>APPN</td>
<td>NO</td>
</tr>
<tr>
<td>NO</td>
<td>NO</td>
<td>LEN</td>
<td>NO</td>
</tr>
<tr>
<td>NO</td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td>APPN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td>LEN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>not specified</td>
<td>YES</td>
<td>APPN</td>
<td>YES</td>
</tr>
<tr>
<td>not specified</td>
<td>YES</td>
<td>LEN</td>
<td>YES</td>
</tr>
<tr>
<td>not specified</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>not specified</td>
<td>NO</td>
<td>APPN</td>
<td>NO</td>
</tr>
<tr>
<td>not specified</td>
<td>NO</td>
<td>LEN</td>
<td>NO</td>
</tr>
<tr>
<td>not specified</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

978  

*® OS V2R1.0 Communications Server: SNA Resource Definition Reference*
Table 71. XNETALS matrix (continued)

<table>
<thead>
<tr>
<th>Start list values</th>
<th>Specified on GROUP, LINE, or PU (lowest level wins)</th>
<th>CONNTYPE</th>
<th>XNETALS and result for the PU</th>
</tr>
</thead>
<tbody>
<tr>
<td>not specified</td>
<td>not specified</td>
<td>LEN</td>
<td>NO</td>
</tr>
<tr>
<td>not specified</td>
<td>not specified</td>
<td>APPN</td>
<td>YES</td>
</tr>
<tr>
<td>not specified</td>
<td>not specified</td>
<td>LEN</td>
<td>NO</td>
</tr>
<tr>
<td>not specified</td>
<td>not specified</td>
<td></td>
<td>NO (Default)</td>
</tr>
</tbody>
</table>

**XNETALS=NO**

Indicates that VTAM’s network ID will be used to represent the connecting node’s network ID.

During connection:
- If the NETID operand is specified on the PU statement then, it must specify VTAM’s network ID.
- If the NETID operand is not coded on the PU statement then, VTAM’s network ID is assumed and the adjacent node’s network ID is ignored.

During LU-LU session establishment:
- If the network ID for the originating LU is specified (during connection time) and is not identical to VTAM’s network ID, the session will not be established.
- If the originating LU is not using a network qualified name, VTAM’s network ID is assumed and the session will be established.

**Note:** If the PU is a dependent LU server or requester, nonnative network connection is not allowed if XNETALS=NO is specified and the NETID operand is not coded.

**XNETALS=YES**

Indicates that the network ID of the adjacent PU will be used for connection and session establishment.

During connection:
- If the NETID operand on the PU statement is coded, it will be used as the PU’s network ID. In this case, the network ID specified on the PU statement must match the network ID specified by the attaching node during connection processing.
- If the NETID operand is not specified on the PU statement, then the network ID specified by the attaching node during connection processing will be assumed.

During LU-LU session establishment:
- If the network ID for the originating LU is specified and does not match the PU’s network ID, the session will not be established.
- If the originating LU name is not a network qualified name, the session will be established. VTAM will qualify the LU’s name with the attaching node’s network ID.
Chapter 5. User-defined tables and data filter

This topic describes how to create or modify the following tables and filter:

- "Associated LU table"
- "Class of service" on page 985
- "CMIP services directory definition file" on page 1008
- "Interpret table" on page 1017
- "Logon mode table" on page 1026
- "Message-flooding prevention table" on page 1046
- "Model name table" on page 1054
- "Subarea mapping table" on page 1057
- "Session awareness data filter" on page 1061
- "SAW sense filter" on page 1066
- "Unformatted system services tables" on page 1069

Follow the assembler language conventions described in "Format of macroinstructions" on page 10 when coding the macroinstructions discussed in this topic.

The MODIFY TABLE operator command lets you dynamically replace the associated LU, CoS, interpret, logon mode, model name, and USS tables, the CMIP services directory definition file, and the session awareness (SAW) data filter. See z/OS Communications Server: SNA Operation for information on the MODIFY TABLE command.

When link editing a table for which IBM has shipped a default, it is recommended that you use the linkage editor default parameters.

To assemble the tables, your JCL will need to point to the SYS1.SISTMAC1 macroinstruction library.

Associated LU table

An associated LU table contains associated LU names that can be passed to VTAM application programs in their LOGON exits. VTAM application programs use the associated LU names to create dynamic definitions for their session-partner resources. These names specify primary and alternate printers that are logically related to the SLU. This information is not available in model definitions. For information on model definitions, see "Model name table" on page 1054.

Operands on an SLU’s resource definition can associate that SLU with the proper associated LU data. The ASLTAB operand specifies the associated LU table to be used, and the ASLENT operand specifies the proper entry within the table. These operands are described for each LU-type resource in “About this topic”. 
Table 72. Associated LU table macroinstructions and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ASLTAB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ASLENT</td>
<td>PRINTER1</td>
<td>984</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRINTER2</td>
<td>985</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ASLPLU</td>
<td>PLU</td>
<td>984</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRINTER1</td>
<td>984</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRINTER2</td>
<td>984</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R  Required operand.

Associated LU table coding notes

This information contains coding notes for the associated LU table.

**ASLTAB macroinstruction**

The ASLTAB macroinstruction indicates the beginning of the associated LU table.

**ASLENT macroinstruction**

The ASLENT macroinstruction indicates the start of an associated LU table entry and optionally builds a default set of associated LU data. The end of the entry is indicated either by another ASLENT macroinstruction or by an end of file.

**ASLPLU macroinstruction**

The ASLPLU macroinstruction sets up associated LU data for a specific PLU within the table entry started by the previous ASLENT macroinstruction.

**Installing an associated LU table**

Store the set of macroinstructions as a member of SYS1.VTAMLST and use the member name as the name of the table.

No assembly or link-edit is required to install this table. The table will be dynamically built upon the first activation of any resource that has a defined association to the table. When the last resource using it is deactivated, the table is deleted and its storage is freed.

IBM does not supply a default associated LU table.

**Associated LU table full syntax**

The full syntax for the associated LU table follows. The syntax for each operand is repeated in "Associated LU table operand descriptions" on page 983.

**ASLTAB**
Associated LU table operand descriptions

The following information describes the operands you can code for the associated LU table.

name
On ASLTAB and ASLPLU:
Specifies optional names for the ASLTAB and ASLPLU macroinstructions.

On ASLENT:
Specifies the required 1–8 character name for the ASLENT macroinstruction.

<table>
<thead>
<tr>
<th>PLU</th>
</tr>
</thead>
<tbody>
<tr>
<td>=plu_name</td>
</tr>
</tbody>
</table>

Specifies the 1–8 character name of the PLU as it is known in the SLU’s network.

If you omit PRINTER1 and PRINTER2, no corresponding associated LU name is sent to the PLU or the subsystem during session initiation. Omit the PRINTER1 and PRINTER2 names if you have specified default data by using the ASLENT macroinstruction, but make sure that you prevent any associated LU data from being sent to a specific PLU.

<table>
<thead>
<tr>
<th>PRINTER1</th>
</tr>
</thead>
</table>
Specifies the 1–8 character name of the primary printer associated with the terminal as identified in the SLU’s network.

**PRINTER2**

Specifies the 1–8 character name of the alternate printer associated with the terminal as identified in the SLU’s network.

If you code PRINTER1 or PRINTER2, the printer must be in the same network as the terminal or must have a unique name that requires no translation across network boundaries. Violation of this restriction might cause failure of print-screen operations during cross-network sessions.

If you code PRINTER1 or PRINTER2 on the ASLENT macroinstruction, VTAM uses this value for any PLUs that are not specified in a following ASLPLU macroinstruction.

If both PRINTER1 and PRINTER2 are omitted on the ASLENT macroinstruction, no default data is created. Consequently, any PLU not specified in one of the ASLPLU macroinstructions within this entry will not normally receive any associated LU data.

### Class of service

You can define class of service (COS) for subarea and APPN, though you do not need to create APPN classes of service unless your network has special requirements.

You define subarea class-of-service tables with macroinstructions, and APPN CoS with definition statements and operands. Details for defining classes of service follow.

### Subarea class-of-service table

To define subarea classes of service, create a class-of-service table with entries containing lists of routes grouped together on the basis of characteristics such as security, transmission priority, and bandwidth. The characteristics of a particular list determine a class of service. For example, you could place the fastest routes in one list for use in interactive sessions and slower routes in another list for use by batch jobs. If a session involves the transmission of sensitive data, you would place only secure routes in the list used for that session. By specifying the name of an
entry from the CoS table in the logon mode table associated with a session, you can select the list of routes you want to be used for the session.

VTAM does not provide a default subarea CoS table. However, you can define a default CoS in the logon mode table by specifying ISTCOSDF on the LOGMODE operand of the MODEENT macroinstruction. You can modify the session characteristics in ISTCOSDF, though if you do, it is recommended that you put ISTCOSDF in a separate table rather than modifying it in the IBM-supplied default logon mode table, ISTINCLM.

VTAM also offers a default list of virtual routes that is used if you do not create a CoS table. This default list is used if you do not name a Class of Service in the logon mode table for a session. It is also used if you specify ISTVTCOS as the Class of Service in the logon mode table and there is no explicit Virtual Route List defined for ISTVTCOS. You can replace this default list by creating a CoS table with an unnamed (blank) CoS entry containing the new list. This new default list is then used if you do not name a Class of Service in the logon mode table for a session, or if you specify ISTVTCOS as the Class of Service in the logon mode table and there is no explicit Virtual Route List defined for ISTVTCOS.

In addition to coding an unnamed CoS entry in your CoS table, you might also want to code an entry for use in SSCP sessions (SSCP-SSCP, SSCP-PU, and SSCP-LU). You can specify the routes used for SSCP sessions by including an entry named ISTVTCOS in the CoS table.

### Table 73. Subarea CoS macroinstructions and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro-instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>COSTAB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>COS</td>
<td>VR</td>
<td>989</td>
<td>R</td>
<td></td>
<td>see description</td>
</tr>
<tr>
<td></td>
<td>SUBSTUT</td>
<td></td>
<td>989</td>
<td></td>
<td>NO</td>
<td></td>
</tr>
</tbody>
</table>

**COSEND**

**Legend:**

| R      | Required operand. |

### Subarea class-of-service table coding notes

This information contains coding notes for the subarea class-of-service table.

**COSTAB macroinstruction**

The COSTAB macroinstruction begins the CoS table and must precede all CoS definition entries.

**COS macroinstruction**

The COS macroinstruction defines a Class of Service entry.

**COSEND macroinstruction**

The COSEND macroinstruction marks the end of the Class of Service table.

**Note:** You can also code an assembler end statement. However, if you code an assembler end statement, it must follow the COSEND macroinstruction.
Attention: Using assembler language statements in generating a CoS table may cause an invalid CoS table to be built.

Installing a subarea CoS table
In a single-network installation, the assembled output of this process must be named ISTSDCOS. A VTAM acting as a gateway SSCP, designated to resolve CoS names for a gateway NCP, uses the CoS table named on the NETWORK or BUILD definition statement in the gateway NCP’s generation deck.

This table should be link-edited as a nonexecutable module into a library associated with the SYS1.VTAMLIB DD statement.

VTAM loads the resulting load module during initialization and uses it later to establish sessions.

Subarea class-of-service table full syntax
The full syntax for the subarea Class of Service table follows. The syntax for each operand is repeated in “Subarea class-of-service table operand descriptions” on page 988.

COSTAB

COS
Subarea class-of-service table operand descriptions

The following information describes the operands you can code for the subarea Class of Service table.

name
On COSTAB:
Specifies a required 1–8 character name for the COSTAB macroinstruction. In a single-network installation, specify ISTSDCOS as the COSTAB name.

On COS:
Specifies an optional 1–8 character name for the COS macroinstruction. If name is not specified, the unnamed default Class of Service is assumed. For SSCP sessions, specify ISTVT COS as the CoS name.

SUBSTUT

Indicates whether this entry will be substituted when VTAM does not recognize the CoS name that is specified. SUBSTUT can be used for only one entry in the CoS table. If more than one entry is coded, only the first one is recognized.

Note: You can monitor substitutions and reject establishment of sessions which you do not want to use the substitute CoS values by coding a session management exit routine.

See [z/OS Communications Server: SNA Customization](https://www.ibm.com/support/knowledgecenter/SSLTBK_2.2.1/com.ibm.zos.bksrvs.doc/fmoflash.cpm.html) for more information about the session management exit routine, and [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/SSLTBK_2.2.1/com.ibm.zos.bknlrg.doc/fmoflash.cpm.html) for more information about substituting Class of Service values.

VR
VR = (virtual_route,trans_priority)

VR = ((virtual_route,trans_priority))

macroinstructions: COS
dependencies: see rules following

Specifies one or more ordered pairs of numbers that must meet the following rules:

- Each pair must be enclosed within parentheses and must consist of a virtual route number \((\text{virtual}\_\text{route})\) and a transmission priority indicator number \((\text{trans}\_\text{priority})\).
- If more than one pair is specified, the entire list must be enclosed within parentheses and each pair must be separated by a comma.
- A virtual route number must be a decimal integer in the range 0–7.
- A transmission priority indicator number must be a decimal integer in the range 0–2, where:
  - 0 Specifies low-priority session traffic
  - 1 Specifies medium-priority session traffic
  - 2 Specifies high-priority session traffic
- You can specify up to 24 ordered pairs.
- Virtual route selection attempts are performed in the order in which the pairs are specified. The sequence of pairs, therefore, should reflect the route selection order that you want. See \(z/OS\) Communications Server: SNA Customization for information on the virtual route selection exit routine.

**APPN Class of Service definitions**

Code an APPN Class of Service with the following definition statements:

- APPNCOS
- LINEROW
- NODEROW

Put the APPN CoS definitions in one or more VTAMLST files that can be activated at VTAM initialization or with the VARY ACT command.

**Table 74. APPN CoS definition statements and operands**

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>APPNCOS</td>
<td>PRIORITY</td>
<td>1000</td>
<td>R</td>
<td></td>
<td>MEDIUM 8</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>NUMBER</td>
<td>997</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LINEROW</td>
<td>NUMBER</td>
<td>997</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CAPACITY</td>
<td>992</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE</td>
<td>994</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTTIME</td>
<td>995</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY</td>
<td>999</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SECURITY</td>
<td>1002</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Uparm1</td>
<td>1002</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Uparm2</td>
<td>1003</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Uparm3</td>
<td>1003</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WEIGHT</td>
<td>1004</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 74. APPN CoS definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>NODE-ROW</td>
<td>NUMBER</td>
<td>998</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NODE-ROW</td>
<td>CONGEST</td>
<td>993</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NODE-ROW</td>
<td>ROUTERES</td>
<td>1001</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NODE-ROW</td>
<td>WEIGHT</td>
<td>1005</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R     Required operand.

APPN Class of Service coding notes
This information contains coding notes for APPN Class of Service.

APPNCOS definition statement
The APPNCOS definition statement begins the APPN CoS definitions.

LINEROW definition statement
The LINEROW definition statement contains the operands which specify line characteristics.

NODEROW definition statement
The NODEROW definition statement contains the operands which specify node characteristics.

Implementing APPN Classes of Service
Details for implementing APPN Classes of Service are given in the z/OS Communications Server: SNA Network Implementation Guide.

See "IBM-supplied APPN Classes of Service" on page 1007 for descriptions of the seven IBM-supplied APPN Classes of Service, and "APPN Class of Service (CoS) definitions (COSAPPN)" on page 1313 for samples of them.

APPN Class of Service full syntax
The full syntax for the APPN CoS follows. The syntax for each operand is repeated in "APPN Class of Service operand descriptions" on page 992.

APPNCOS
APPN Class of Service operand descriptions

The following information describes the operands you can code for the APPN Class of Service.

CAPACITY
CAPACITY

statements: LINEROW
dependencies: APPN only
range: 1K–1000M (for both min and max)

Specifies the range of the number of bits per second that the link can transmit data. The same range of values applies to min and max.

Table 75. Default values for LINEROW CAPACITY operand

<table>
<thead>
<tr>
<th>Eight-row definitions</th>
<th>Twelve-row definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>The minimum default depends on the line row being defined:</td>
<td>The minimum default depends on the line row being defined:</td>
</tr>
<tr>
<td><strong>Row</strong></td>
<td><strong>Minimum default</strong></td>
</tr>
<tr>
<td>1</td>
<td>4M</td>
</tr>
<tr>
<td>2</td>
<td>56000</td>
</tr>
<tr>
<td>3</td>
<td>19200</td>
</tr>
<tr>
<td>4</td>
<td>9600</td>
</tr>
<tr>
<td>5</td>
<td>19200</td>
</tr>
<tr>
<td>6</td>
<td>9600</td>
</tr>
<tr>
<td>7</td>
<td>4800</td>
</tr>
<tr>
<td>8</td>
<td>MINIMUM</td>
</tr>
</tbody>
</table>

The maximum default is MAXIMUM for all line rows.

CONGEST
CONGEST = min,max

statements: NODEROW
dependencies: APPN only
range: LOW–HIGH

Indicates whether congestion is allowed for each node row.

CONGEST allows you to indicate the weighting preference for uncongested nodes, or to disallow the use of congested nodes.

A range of LOW,LOW indicates only uncongested nodes are allowed for the node row. A range of LOW,HIGH indicates uncongested and congested nodes are allowed for the node row.

min=LOW or HIGH and max=LOW or HIGH. The minimum default is LOW for all node rows. The maximum default depends on the node being defined:

<table>
<thead>
<tr>
<th>Row</th>
<th>Maximum default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LOW</td>
</tr>
<tr>
<td>2</td>
<td>LOW</td>
</tr>
<tr>
<td>3</td>
<td>LOW</td>
</tr>
<tr>
<td>4</td>
<td>LOW</td>
</tr>
<tr>
<td>5</td>
<td>LOW</td>
</tr>
<tr>
<td>6</td>
<td>LOW</td>
</tr>
<tr>
<td>7</td>
<td>HIGH</td>
</tr>
<tr>
<td>8</td>
<td>HIGH</td>
</tr>
</tbody>
</table>

COSTBYTE
Specifies a range for cost-per-byte. The cost per byte is a relative value indicating the cost of the line per byte sent. Low values are less expensive than high values.

Table 76: Default values for LINEROW COSTBYTE operand

<table>
<thead>
<tr>
<th>Row</th>
<th>Eight-row definitions</th>
<th>Twelve-row definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>128</td>
<td>128</td>
</tr>
<tr>
<td>7</td>
<td>196</td>
<td>128</td>
</tr>
<tr>
<td>8</td>
<td>255</td>
<td>128</td>
</tr>
<tr>
<td>9</td>
<td>128</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>128</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>255</td>
</tr>
<tr>
<td>12</td>
<td>255</td>
<td></td>
</tr>
</tbody>
</table>

COSTTIME
Specifies a range for cost-per-connect-time. The cost per connect time is a relative value indicating the cost of the line per unit of connect time. Low values are less expensive than high values.

**Table 77. Default values for LINEROW COSTTIME operand**

<table>
<thead>
<tr>
<th>Eight-row definitions</th>
<th>Twelve-row definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>The minimum default is 0 for all line rows. The maximum default depends on the line row being defined:</td>
<td>The minimum default is 0 for all line rows. The maximum default depends on the line row being defined:</td>
</tr>
<tr>
<td><strong>Row</strong></td>
<td><strong>Maximum default</strong></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>128</td>
</tr>
<tr>
<td>7</td>
<td>196</td>
</tr>
<tr>
<td>8</td>
<td>255</td>
</tr>
<tr>
<td>9</td>
<td>128</td>
</tr>
<tr>
<td>10</td>
<td>64</td>
</tr>
<tr>
<td>11</td>
<td>255</td>
</tr>
<tr>
<td>12</td>
<td>255</td>
</tr>
</tbody>
</table>

**name**
Specifies a required 1–8 character name for the APPNCOS definition statement. The name must be unique within a node.

**Note:** If you reuse the name of an APPN Class of Service definition, the operand values of the later definition replace the operand values of the earlier definition.

**NUMBER (APPNCOS)**

Specifies which set of default values is to be used for the APPN CoS table. If you specify 8, the default values for 8-row definitions are in effect. If you specify 12, the default values for 12-row definitions are in effect.

See "IBM-supplied APPN Classes of Service" on page 1007 for information on how to determine which set of default values is appropriate for the CoS table you are defining.

**NUMBER (LINEROW)**
Linerow (Linerow)

Specifies a line row to be defined.

The default characteristic values of a given line row can differ depending on the number of the row being defined. So, an operand might have a different value for line row 7 than for line row 8. See the line row operand descriptions for defaults for each row number.

Line rows are searched in ascending order to determine whether a given TG is acceptable for a particular CoS. For example, when evaluating TG1 for use in route selection, its characteristics are compared against line row 1 for COSX. If TG1's characteristic values are within the acceptable range of values for the characteristics of line row 1, TG1 is assigned the weight specified for line row 1. If TG1's characteristics are not within the acceptable range, the next defined line row for COSX is compared against TG1. If no COSX line rows are found that agree with TG1's characteristic values, TG1 is assigned an infinite weight and is not used to calculate a route for COSX.

A line row is defined when its number is specified, even though no other line row operands are coded.

**NUMBER (NODEROW)**
**NODEROW**

Specifies a node row to be defined.

The default characteristic values of a given node row can differ depending on the number of the row being defined. So, an operand might have a different value for node row 7 than for node row 8. See the node row operand descriptions for defaults for each row number.

Node rows are searched sequentially based on this numbering to determine whether a given node is acceptable for a particular CoS.

For example, when NODE A becomes active, its characteristics are compared against node row 1 for COSX. If NODE A’s characteristic values are within the acceptable range of values for the characteristics of node row 1, then NODE A is assigned the weight specified for node row 1. If NODE A’s characteristics are not within the acceptable range, the next defined node row for COSX is compared against NODE A. If no COSX node rows are found that meet NODE A’s characteristic values, NODE A is assigned an infinite weight and is not used for route calculation for COSX.

A node row is defined when its number is specified, even though no other node row operand are specified.

**PDELAY**
PDELAY = min,max

**statements:** LINEROW
**dependencies:** APPN only
**range:** see values

Specifies a range for propagation delay. Propagation delay represents the time needed for a signal to travel from one end of the link to the other.

*min* and *max* have the same range of the following values:

- **MINIMUM**
  - Indicates minimum delay.
- **NEGLIGIB**
  - Indicates a local area network delay (less than 0.48 milliseconds).
- **TERRESTR**
  - Indicates telephone network delay (between 0.48 and 49.152 milliseconds).
- **PACKET**
  - Indicates a packet-switched network delay (between 49.152 and 245.76 milliseconds).
- **LONG**
  - Indicates a satellite delay (greater than 245.76 milliseconds).
- **MAXIMUM**
  - Indicates a maximum delay.

**Table 78. Default values for LINEROW PDELAY operand**

<table>
<thead>
<tr>
<th>Row</th>
<th>Eight-row definitions</th>
<th>Twelve-row definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>MAXIMUM</td>
<td>MAXIMUM</td>
</tr>
<tr>
<td>7</td>
<td>MAXIMUM</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>PACKET</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>PACKET</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>TERRESTR</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>TERRESTR</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>TERRESTR</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>NEGLIGIB</td>
<td></td>
</tr>
</tbody>
</table>

The minimum default is MINIMUM for all line rows. The maximum default depends on the line row being defined: MAXIMUM for all line rows.

**PRIORITY**
Specifies one of the following transmission priorities for an APPN CoS:

**PRIORITY=LOW**
- Recommended for batch traffic

**PRIORITY=MEDIUM**
- Recommended for normal traffic

**PRIORITY=HIGH**
- Recommended for interactive traffic

**PRIORITY=NETWORK**
- Used for APPN network services traffic, and is valid only for CPSVCMG and SNASVCMG.

**ROUTERES**

Specifies the range of the relative amount of route addition resistance (RAR) for each node row. The route addition resistance is used to identify network nodes that are more preferable for intermediate routing compared to other network nodes. A low RAR value indicates a more desirable node for intermediate traffic.

The minimum default is 0 for all node rows. The maximum default depends on the node row being defined:

<table>
<thead>
<tr>
<th>Row</th>
<th>Maximum default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>31</td>
</tr>
<tr>
<td>2</td>
<td>63</td>
</tr>
<tr>
<td>3</td>
<td>95</td>
</tr>
<tr>
<td>4</td>
<td>127</td>
</tr>
<tr>
<td>5</td>
<td>159</td>
</tr>
<tr>
<td>6</td>
<td>191</td>
</tr>
<tr>
<td>7</td>
<td>223</td>
</tr>
<tr>
<td>8</td>
<td>255</td>
</tr>
</tbody>
</table>
SECURITY

\[ \text{SECURITY} \rightarrow \text{min}, \text{max} \]

statements: LINEROW
dependencies: APPN only
range: see values

Specifies a range from the minimum security level to the maximum security level.

\textit{min} and \textit{max} can each be one of the following values, which are listed from least to most secure:

\textbf{UNSECURE}
- Indicates not secured.

\textbf{PUBLIC}
- Indicates public switched network.

\textbf{UNDERGRO}
- Indicates unguarded underground cable.

\textbf{SECURE}
- Indicates unguarded secure conduit.

\textbf{GUARDED}
- Indicates guarded conduit, physical only.

\textbf{ENCRYPT}
- Indicates link encryption.

\textbf{SHIELDED}
- Indicates guarded conduit, physical and radiation shielded.

\textbf{MAXIMUM}
- Indicates maximum security (equivalent to \textit{SHIELDED}).

The minimum default is UNSECURE for all line rows. The maximum default is MAXIMUM for all line rows.

UPARM1
UPARM1 = \textit{min,max} \\
statements: LINEROW \\
dependencies: APPN only \\
range: 0–255

Specifies a range for a user-defined value. The user determines the meaning of this value.

See \textit{z/OS Communications Server: SNA Network Implementation Guide} for an example of using the UPARM operands.

UPARM2 \\

\textit{UPARM2 = \textit{min,max}} \\
statements: LINEROW \\
dependencies: APPN only \\
range: 0–255

Specifies a range for a user-defined value. The user determines the meaning of this value.

UPARM3
Specifies a range for a user-defined value. The user determines the meaning of this value.

**WEIGHT (LINEROW)**

Specifies the relative weight for each line row. Low weights indicate more desirable choices.

When the TG (transmission group) goes to a VRN (virtual real node), the weight of the TG is halved.

**Note:** An ascending order of weights on a row-by-row basis will not be enforced. For example, line row 1 can be assigned a weight that is greater than line row 2. However, this is not recommended.

**Table 79. Default values for LINEROW WEIGHT operand**

<table>
<thead>
<tr>
<th>Eight-row definitions</th>
<th>Twelve-row definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>The default depends on the line row being defined:</td>
<td>The default depends on the line row being defined:</td>
</tr>
<tr>
<td>Row</td>
<td>Default</td>
</tr>
<tr>
<td>-----</td>
<td>--------</td>
</tr>
<tr>
<td>1</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>60</td>
</tr>
<tr>
<td>3</td>
<td>90</td>
</tr>
<tr>
<td>4</td>
<td>120</td>
</tr>
<tr>
<td>5</td>
<td>150</td>
</tr>
<tr>
<td>6</td>
<td>180</td>
</tr>
<tr>
<td>7</td>
<td>210</td>
</tr>
<tr>
<td>8</td>
<td>240</td>
</tr>
<tr>
<td>9</td>
<td>180</td>
</tr>
<tr>
<td>10</td>
<td>200</td>
</tr>
<tr>
<td>11</td>
<td>220</td>
</tr>
<tr>
<td>12</td>
<td>240</td>
</tr>
</tbody>
</table>
WEIGHT (NODEROW)

Statements: NODEROW
Dependencies: APPN only
Range: 0–255

Specifies the relative weight for each node row. Low weights indicate more desirable choices.

Note: An ascending order of weights on a row-by-row basis is not enforced. For example, node row 1 can be assigned a weight that is greater than node row 2. However, this is not recommended.

The default depends on the node row being defined:

<table>
<thead>
<tr>
<th>Row</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
</tr>
<tr>
<td>5</td>
<td>80</td>
</tr>
<tr>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>7</td>
<td>120</td>
</tr>
<tr>
<td>8</td>
<td>150</td>
</tr>
</tbody>
</table>

Default values for APPN Class of Service definitions

You can use default values for APPN CoS operands. The examples in this information illustrate the use of defaults for APPN Class of Service definitions along with descriptions of how default values are substituted.

Note: Defaults apply only to partially coded rows. If a row is not coded, no defaults are assumed.

Example 1: WEIGHT operand not coded

In the following example, the WEIGHT operand was not specified. Because the line row number was specified as 5 and the default values for 8-row definitions are in effect (NUMBER=8 on the APPNCOS definition statement), the value 150 is substituted as the default value for WEIGHT. Had the default values for 12-row definitions been in effect (NUMBER=12 on the APPNCOS definition statement), the value 100 would have been substituted as the default value for weight. Note also that line rows 1, 2, 3, 4, 6, 7, and 8 were not coded. Therefore, only one line row is defined for COSDEF1.

***********************************************************************
* COS entry                                                      *
***********************************************************************
COSDEF1 APPNCOS PRIORITY=HIGH
transmission priority
Example 2: Minimum value for COSTTIME operand not coded
In the following example, the minimum value for COSTTIME was not specified. Because the line row number was specified as 2, the value 0 is substituted as the minimum default value for COSTTIME. In this case, the default value for 8-row definitions is being used because NUMBER is not coded on the APPNCOS definition statement and it therefore defaults to 8. However, even if NUMBER=12 was coded on the APPNCOS definition statement, the value 0 would still be substituted as the minimum default value for COSTTIME because for line row 2, the minimum default value is 0 for both 8-row and 12-row definitions. Also, UPARM1 was not coded. The values 0 and 255 are substituted for the UPARM1 minimum and maximum values respectively.

Example 3: One line row defined for COSDEF3
In the following example, one line row has been defined for COSDEF3. Because the only operand specified for the line row was its number, default values are substituted for the remaining line row characteristics. These values are obtained from the appropriate set of defaults for line row 8, which in this case, is the set of defaults for 8-row definitions (because NUMBER is not coded on the APPNCOS definition statement and it therefore defaults to 8). Had NUMBER=12 been coded on the APPNCOS definition statement, the set of defaults for 12-row definitions would have been substituted for the remaining line row characteristics for line row 8. The same situation applies for the node row coded in this example, except that the default values are obtained from the set of defaults for node row 1.
IBM-supplied APPN Classes of Service

Descriptions of the seven IBM-supplied APPN Classes of Service follow. These classes define seven generic types of session traffic. See "APPN Class of Service (CoS) definitions (COSAPPN)" on page 1313 for samples.


The IBM-supplied Classes of Service are shipped with the name COSAPPN in SYS1.ASAMPLIB and should be copied into SYS1.VTAMLST at VTAM installation.

Note: The COSAPPN member is automatically activated.


#BATCH
An APPN CoS for LU-LU sessions that specifies a general batch-oriented CoS that uses low transmission priority, and for which high bandwidth and low cost are considered more important than short delay.

#BATCHSC
An APPN CoS for LU-LU sessions that specifies a general batch-oriented CoS that uses low transmission priority, and for which high bandwidth and low cost are considered more important than short delay. A minimum security level is required.

#CONNECT
An APPN CoS for LU-LU sessions that provides connectivity at medium transmission priority.

#CONNECT should not be changed.

CPSVCMG
An APPN CoS for CP-CP sessions that is used for network flows. It provides connectivity at network transmission priority.
CPSVCMG should not be changed.

#INTER
An APPN CoS for LU-LU sessions that specifies a general, interactive Class of Service that uses high transmission priority, and for which short delay is considered more important than high bandwidth and lost cost.

#INTERSC
An APPN CoS for a LU-LU session that specifies a general, interactive CoS that uses high transmission priority, and for which short delay is considered more important than high bandwidth and lost cost. A minimal security level is required.

SNASVCMG
An APPN CoS for LU-LU CNOS sessions that provides connectivity at network transmission priority.
SNASVCMG should not be changed.
CMIP services directory definition file

If you specified the OSIMGMT=NO start option, you do not need to read this information.

If you specified the OSIMGMT=YES start option, a directory definition file is required. This information describes how to code the directory definition file to implement Common Management Information Protocol (CMIP) services.

The CMIP services directory definition file performs name mapping and address mapping, and controls the associations that CMIP services on this host has with other CMIP services and with itself. An association is a connection between CMIP services on this host and CMIP services on another node or between CMIP services on this host and itself.

The IBM-supplied directory definition file in member ACYDDF of the SYS1.SISTCMIP data set allows unlimited associations with other instances of CMIP services and between CMIP services and itself. If you do not update the IBM-supplied file, any CMIP application program in any network can have access to CMIP services on this host.

To learn about the purpose of the CMIP services directory definition file, see the z/OS Communications Server: SNA Network Implementation Guide.

This information covers the following topics:

- Example of a simple directory definition file
- Explanation of the example directory definition file
- Specifying different types of associations
- Coding the directory definition file
- How CMIP services uses the directory definition file
- Troubleshooting the directory definition file

Example of a simple directory definition file

The following example shows a simple directory definition file for NETA. An explanation follows the example.

# Example directory definition file.

class aetitle

class aetitle
  name '1.3.18.0.2.4.6=NETA'
  associationKey '*'

class aetitle
  name '1.3.18.0.2.4.6=NETB'
  associationKey 'a0b1c2d3e4f50011'

class aetitle
  name '*'
  associationKey '-'

Explanation of the example directory definition file

The directory definition file contains comments and directory entries:

- Comments are not processed. They begin with the # symbol in column 1. Blank lines are treated as comments.
• Directory entries determine which associations are allowed, and for those associations, what type of security is to be used.

Each directory entry contains attributes and their values. An attribute is similar to a VTAM operand, in that the attribute and the value specified on it define a characteristic of an object. In the directory definition file, the object is CMIP services.

The value is a single expression, which can contain blanks if the value is enclosed in quotation marks. Everything after the value in a directory entry record is treated as a comment record and is not processed.

An entry ends where the next entry begins (with a different name or class attribute) or at the end of the directory definition file.

The first attribute in the example directory definition file is the class attribute, with the value \textit{aetitle}. See “Format of directory definition file entry records” on page 1010 for an explanation of class. This entry specifies that the names under the class attribute are assumed to be CMIP services names or names of CMIP application programs that have registered their own AE titles. For a description of AE titles, see \textit{z/OS Communications Server: CMIP Services and Topology Agent Guide}.

The class and its attribute are followed by the second attribute, name with a value. The required name attribute is the first attribute for an object that belongs in the class specified by the preceding class attribute. CMIP services searches for the name attribute in the directory definition file when it determines whether a particular instance of CMIP services can have an association with CMIP services on this host.

The third attribute is the associationKey attribute with a value. Specifying associations is discussed in the following information.

\textbf{Specifying different types of associations}

In the directory definition file, you can specify the following options for establishing associations:

• No security on associations
• Data encryption standard (DES)-based security on associations
• Application-program-to-application-program security on associations
• No associations

\textbf{No security on associations}

The following entry
\begin{verbatim}
class aetitle
   name '1.3.18.0.2.4.6=NETA'
   associationKey '*'
\end{verbatim}

specifies that all instances of CMIP services on network NETA can exchange requests and actions with the CMIP services on this host. All associations are allowed.

\textbf{DES-Based security on associations}

The following entry
\begin{verbatim}
class aetitle
   name '1.3.18.0.2.4.6=NETB'
   associationKey 'a0b1c2d3e4f50011'
\end{verbatim}
specifies that some instances of CMIP services on NETB can exchange requests and actions with CMIP services on this host. Associations are allowed only if the correct DES encryption key is used. Note that the directory definition file on any instance of CMIP services on NETB is required to have the same DES encryption key for the association to be allowed.

In the following example, the combination of the * value for the name attribute and the associationKey attribute establishes an encryption key that must be used by all instances of CMIP services that are not already specified with a name attribute on another entry in the directory definition file.

class aetitle
  name '*'
  associationKey '0123456789abcdef'

This entry specifies that all associations with CMIP services on this host must use the same encryption key, 0123456789abcdef, to have an association with CMIP services on this host.

**Application-program-to-application-program security on associations**
The following entry

class aetitle
  name '1.3.18.0.2.4.6=NETA'
  associationKey '.'

specifies that the CMIP application programs control access to themselves, rather than CMIP services controlling access to the application programs.

If you want to use security, but do not want DES-based security, the CMIP application programs can define application-program-to-application-program security. The VTAM topology agent and the NetView program do not implement application-program-to-application-program security. For information about defining this type of security, see [z/OS Communications Server: CMIP Services and Topology Agent Guide](https://www.ibm.com/support/knowledgecenter/SANSN_2.1.0/com.ibm.zos.v2r1.sncmd.web.sna/guides/appprogram_program_security.html).

**No associations**
The following entry

class aetitle
  name '*'
  associationKey '-'

specifies that no associations are allowed for instances of CMIP services on other networks, except NETA and NETB, because they are specifically listed with a name attribute and an associationKey attribute. The asterisk specified as the value of the name attribute means that the directory entry applies to any name not specifically listed in another directory entry in the file.

**Coding the directory definition file**
This information explains the format and rules used to code entries in the directory definition file.

**Format of directory definition file entry records**
Directory entry records can have one or more blanks (for indentation) before the keyword. There can be multiple blanks between the keyword and the value. The value is a single string and can contain blanks only when it is enclosed in quotes. Everything after the value in a record is treated as a comment and is ignored.
A minimum entry consists of a name line and one or more attributes associated with that name. Each entry is ended by the next entry (class or name) or the end of the file. More specifically, an entry consists of the following items:

- The class of a list of entries. The class applies to all succeeding entries until another class is specified. The first nonblank item on the line is the keyword "class", followed by the class attribute, or type (separated from the "class" keyword by at least one blank). The only types of classes are:
  - managed object
  - aetitle

The first entry in the file must start with a class record, or a "no class specified" message is issued, and initialization or reloading of the table fails. Subsequent entries can omit the class record if these entries belong to the last class specified.

- The name attribute. This is the string against which the search is made, within the class. The first nonblank item on the line is the keyword "name", followed by the value of the name (a string in single quotes), separated from the keyword by at least one blank. Multiple entries with the same name value may exist in the directory definition file, providing all attributes under entries with that name value are different.

- After the name attribute, an attribute associated with an entry. The first nonblank item on the line is the keyword for a particular attribute, followed by the value (a string in single quotes) of the attribute, separated from the keyword by at least one blank. The current attributes used by CMIP services for the class, managed object, are:
  - name
  - aetitle

The current attributes used by CMIP services for the class, aetitle, are:
  - name
  - address
  - associationKey
  - timeSync

If an attribute is encountered that has already been defined for a particular name value (for example, a duplicate attribute entry, such as two different associationKey values), the second one produces a "duplicate attribute" message, and initialization or table reloading fails.

Individual attributes are optional on any particular entry.

Each record in the file can extend over many lines, up to a maximum of 2080 character spaces. This includes any leading spaces before an attribute at the beginning of the line. The maximum logical record length (LRECL) for the directory definition file is 2080. The record format (RECFM) for the file must be FIXED.

Code entries as one or more records in the following format:

```
Column 1                  End of line

[...attribute[ value]...            ...&

... Indicates blanks. One or more blanks can precede the first attribute.

attribute
  Represents a characteristic of the managed object. Directory entry records
```
can have one or more blanks before the attribute and can have multiple
blanks between the attribute and the value. Separate the attribute from the
value with one or more blanks.

value The value can have intervening blanks, if it is enclosed in quotation marks.
& If the entry requires several records, code the continuation character (&) in
the last position on the line.
#
Comment lines begin with the # symbol in column 1. Comments are not
processed. Blank lines are treated as comments.

The following information shows a sample entry in a directory definition file. The
first line is a scale to show column positions. Every tenth column position is
indicated by a number.

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
class managed object
name '1.3.18.0.2.4.6=USIBMNR;2.9.3.2.7.4=(name NR555501)'
aetitle '1.3.18.0.2.4.6=USIBMNR;2.9.3.2.7.4=(name "NR555551");1.3.18.0.2.4.12=OSISMASE'

Observe the following rules if more than one record is required for an entry.
• Code the line up to the point where you want to break the line. This point can
be anywhere on the line.
• At the end of the line, code the continuation character &; The & symbol does not
form part of the line that is processed.
• Continue the entry in the first column of the next line. A comment line cannot
follow a continuation line.
• Continue coding lines with continuation characters until the entry is finished.
• Entries can be coded on as many lines as needed.

Attributes coded in the directory definition file
For VTAM CMIP services, the following attributes are valid:
• On the class, managed object
  – name
  – aetitle
• On the class, aetitle
  – name
  – address
  – associationKey
  – timeSync

How to code the name attribute:
The first nonblank item on the line is the name attribute followed by the value of
the name attribute, which is enclosed in single quotation marks.

At least one blank must separate the name attribute from the value for the name
attribute. The value of the name attribute is an AE-title.

The format of the value for the name attribute can vary. The actual value to be
used is determined by the manager or agent application program.

Multiple entries with the same value for the name attribute can exist in the
directory definition file when all attributes under entries with that name value are
different.
The following examples show how to specify the value for the name attribute. The value stands for an instance of CMIP services.

- The name of CMIP services on a particular network-qualified CP or SSCP name.
  An ampersand causes the line to continue.
  ```
  name '1.3.18.0.2.4.6=NETA;2.9.3.2.7.4=(name SSCP1A); &
  1.3.18.0.2.4.12=OSISMASE'
  ```

  The name consists of object identifiers and values. Object identifier 1.3.18.0.2.4.6 means the SNA network node name:
  - ISO has assigned 1.3.18 to IBM.
  - IBM has assigned 0.2.4.6 to mean a SNA network node name.

  The other object identifiers are similarly assigned:
  - Object identifier 2.9.3.2.7.4 means a SNA node name, either a CP or an SSCP name.
  - Object identifier 1.3.18.0.2.4.12 means the application entity title (AE-title).
    Generally, the AE-title refers to CMIP services, which is always OSISMASE for VTAM.

- A network-qualified CP or SSCP name by itself:
  ```
  name '1.3.18.0.2.4.6=NETA;2.9.3.2.7.4=(name SSCP1A)'
  ```

  A trailing wildcard character (*) can be used in the CP or SSCP name to indicate that the rule matches any network-qualified CP or SSCP name with the specified netid and a CP or SSCP name that matches up through the wildcard character and has zero or more characters after it. If the wild card is specified, it must be the last character in the CP or SSCP name.

  For example, the following name attribute rules match NETA.SSCP2B:
  1. ```
     name '1.3.18.0.2.4.6=NETA;2.9.3.2.7.4=(name S*)'
  ```
  2. ```
     name '1.3.18.0.2.4.6=NETA;2.9.3.2.7.4=(name SSSC*)'
  ```
  3. ```
     name '1.3.18.0.2.4.6=NETA;2.9.3.2.7.4=(name SSSCP2*)'
  ```
  4. ```
     name '1.3.18.0.2.4.6=NETA;2.9.3.2.7.4=(name SSSCP2B*)'
  ```
  5. ```
     name '1.3.18.0.2.4.6=NETA;2.9.3.2.7.4=(name SSSCP2B)'
  ```

  If an exact match is specified, such as 3 in the previous example, it will be selected, even if there are one or more wildcard rules which match, such as 1, 2, 3, and 4 in the previous example.

  If more than one exact match is specified, it is indeterminate which exact match will be selected by CMIP Services.

  If no exact match is specified and multiple wildcard matches are specified, it is indeterminate which wildcard match will be selected by CMIP Services.

- A network name by itself:
  ```
  name '1.3.18.0.2.4.6=NETA'
  ```

- Global match, which matches any name:
  ```
  name '*'
  ```

**How to code the address attribute:**

Values for address records are specified according to the protocol used to communicate over the association.
You can specify a network ID and node in the address attribute, rather than allowing CMIP services to extract them from the AE-title.

**How to code the aetitle attribute:**
The aetitle attribute is composed of three parts:

* The network ID
* The node or network addressable unit
* The System Management Application Entity (SMAE) name

The three parts are separated by semicolons. Except within the node part, there are no embedded blanks.

```plaintext
class managed object
  name '1.3.18.0.2.4.6=USIBMNR;2.9.3.2.7.4=(name NR55550I)'
  aetitle '1.3.18.0.2.4.6=USIBMNR;2.9.3.2.7.4=(name "NR55555I");1.3.18.0.2.4.12=OSISMASE'
```

* The object identifier 1.3.18.0.2.4.6 is assigned the value equivalent to the network ID for the first part.

* The second part consists of the node ID (CP Name) associated with object identifier 2.9.3.2.7.4, in the form (name "nodeID"). Note that the above example shows the nodeID both by itself and surrounded by double-quotes ("').
  Algorithmic mapping performed by directory services always generates an AE-title with the node ID enclosed in double-quotes. Therefore, they should also be specified in any directory definition file “aetitle” records. However, directory services allows a match even when the directory definition file record does not have the node ID enclosed in quotes.

* The third part is the SMAE name, “OSISMASE” in the above example. It is the value for object identifier 1.3.18.0.2.4.12.

In general, when interconnecting management networks, the network administrator or other persons updating the directory definition file must determine how the aetitle is to be specified for each remote partner for which an entry must be made.

**How to code the associationKey attribute:**
The associationKey attribute specifies one of the following options for the association:

* No security
* DES-based security
* Application-program-to-application-program security
* No association

The value can be a hexadecimal representation of the actual DES encryption key. Hexadecimal representation is used to avoid problems between systems using EBCDIC and systems using ASCII for character representations.

Unless the key is one of the three special character values described in the following list, the key must be 16 hexadecimal digits long. This is essentially the same as having a 16-character key consisting of digits 0-9 and letters A-F. Either uppercase or lowercase letters can be used, though VTAM translates and processes letters in uppercase.

The four associationKey values specify how access is to be controlled. To indicate that:

* All instances of CMIP services have access to CMIP Services on this host, specify:
associationKey '*'

- CMIP services controls access to itself by using DES encryption, specify the value of the hexadecimal key used for DES encryption:
  associationKey 'a0b1c2d3e4f50011'

- CMIP applications programs (both manager and agent application programs) control access to themselves, rather than CMIP services controlling access to the application programs, specify:
  associationKey '. '

**Note:** The VTAM topology agent and the NetView program do not implement application-program-to-application-program security.

- No associations are allowed, specify:
  associationKey '- '

**How to code the timesync attribute:**
The timesync attribute specifies the maximum time difference (specified in seconds) that is allowed between the current time of day (specified in date, hours, minutes, and seconds) on this host and the current time of day on the node of the partner CMIP services.

The value is an absolute value. For example, if you specify 4 seconds as the value for the timesync parameter, the system time of day can be either 4 seconds ahead of or 4 seconds behind the value received.

The default is 300 seconds (5 minutes). The maximum value allowed is 86400 (1 day).

If a timesync value of 0 is specified, time-of-day checking is disabled.

**How CMIP services uses the directory definition file**
The directory definition file contains names of other instances of CMIP services and of itself. When searching the directory definition file, CMIP services uses the most specific name it can find in the directory definition file. CMIP services searches the directory definition file for the attribute value for a particular name. Here is an example of a directory definition file, followed by some sample searches. (The line numbers are shown for reference only; they cannot appear in a valid directory definition file.)

```
1  class aetitle
2   name '*'
3   timeSync 800
4
5  class aetitle
6   name '1.3.18.0.2.4.6=SNANET1'
7   associationKey 'd6c8f0d3e4F7c5c2'
8   timeSync 17500
9
10 class aetitle
11  name '1.3.18.0.2.4.6=SNANET1;2.9.3.2.7.4=(name SECURED)'  
12  address USIBMNR.RALVMS
13  associationKey '*'
14
15 class aetitle
16  name '1.3.18.0.2.4.6=SNANET1;2.9.3.2.7.4=(name SECURED);&
17 1.3.18.0.2.4.12=BadSMASE'
18  associationKey '- ' 
```
CMIP services searches the directory definition file for a particular attribute for a given full name. Here are some sample searches:

- Search for the associationKey attribute for name
  
  `1.3.18.0.2.4.6=SNANET1;2.9.3.2.7.4=(name SECURED);14 1.3.18.0.2.4.12=BadSMASE`

  In this example, line 18 is used (name matches exactly and associationKey exists for that specific name).

- Search for the associationKey attribute for name
  
  `1.3.18.0.2.4.6=SNANET1;2.9.3.2.7.4=(name SECURED);14 1.3.18.0.2.4.12=OSISMASE`

  In this example, line 13 is used. The exact name is not in the directory definition file, so CMIP services searched for name `1.3.18.0.2.4.6=SNANET1;2.9.3.2.7.4=(name SECURED)` in the directory definition file and found it, and it had an associationKey.

- Search for the associationKey attribute for name
  
  `1.3.18.0.2.4.6=SNANET1;2.9.3.2.7.4=(name UNSECURED);14 1.3.18.0.2.4.12=OSISMASE`

  In this example, line 7 is used. The exact name, and the first 2/3 name are not found, but name `1.3.18.0.2.4.6=SNANET1` is found, and it has an associationKey.

- Search for the associationKey attribute for name
  
  `1.3.18.0.2.4.6=OTHERNET;2.9.3.2.7.4=(name GEORGE);14 1.3.18.0.2.4.12=OSISMASE`

  In this example, name is not found, which is equivalent to associationKey "*" (all associations allowed).

- Search for the timeSync attribute for name
  
  `1.3.18.0.2.4.6=SNANET1;2.9.3.2.7.4=(name SECURED);14 1.3.18.0.2.4.12=OSISMASE`

  In this example, line 8 is used. Note that line 11 matches this name (the first 2/3 of this name, actually), but there is no timeSync for that name. So CMIP services keeps searching, and finds a match for the first 1/3 of the name and it does have a timeSync value.

**Troubleshooting the directory definition file**

If you receive the message IST1444I filename NOT LOADED - DUPLICATE ATTRIBUTE AT RECORD record, CMIP services has encountered an attribute that has already been defined for an object. The directory definition file fails to load. For example, if the associationKey attribute has already been defined with a value and a different associationKey value is encountered for the same value on the name attribute, message IST1444I is issued.

If you receive the message IST1444I filename NOT LOADED - SYNTAX ERROR AT RECORD record, CMIP services has encountered an attribute that is not valid. The directory definition file fails to load.

If you receive the message IST1444I filename NOT LOADED - NO CLASS SPECIFIED, the class attribute is missing. The directory definition file fails to load.

For other examples, see the description of message IST1444I in [z/OS Communications Server: SNA Messages](https://publib.boulder.ibm.com/infocenter/comdev/v2r1/topic/com.ibm.zos.v2r1.doc/comm/v2r1/commhlp/commhlp.html).
Interpret table

When VTAM receives a logon or logoff request, it uses the interpret table to determine which application program is to be notified.

The standard logon procedure should meet the needs of most installations. (See “LOGON command” on page 1103 for the standard logon procedure.) But, you can write your own interpret table for special circumstances (for example, if the logon sequence you want to use does not follow the syntax for USS commands).

IBM does not supply a default logon-interpret routine, but if you need one, you can code it.

Note: An independent logical unit does not support “uninterpreted” names; therefore, the interpret table does not apply to names received from an independent logical unit.

Table 80. Interpret table macroinstructions and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro-instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>INTAB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>LOGCHAR</td>
<td>APPLID</td>
<td>1022</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>REMOVE</td>
<td>1023</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SEQNCE</td>
<td>1024</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>ENDINTAB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R Required operand.

Interpret table coding notes

This information contains coding notes for the interpret table.

INTAB macroinstruction

The INTAB macroinstruction defines an interpret table that lists the VTAM application programs with which one or more logical units can establish a session. One INTAB macroinstruction defines the name of the interpret table and a group of logon message definitions.

LOGCHAR macroinstruction

Each LOGCHAR (logon-characters) macroinstruction defines a single logon message and the name of an application program, a logon-interpret routine, or a USERVAR. You can include more than one LOGCHAR macroinstruction in an interpret table.

VTAM compares the logon message (character by character) with successive entries in the specified interpret table. If the leading characters in the logon message correspond to all the characters in an entry in the interpret table, VTAM accepts the logon message as valid (even though the logon message can be longer than the corresponding entry in the interpret table). If the first character or characters of several logon messages are identical, you should arrange the LOGCHAR
macroinstructions so the logon sequences for the logon messages are from the most restrictive (greatest number of characters) to the least restrictive (fewest number of characters). For example:

SEQ1 LOGCHAR APPLID=(APPLICID,AP2),SEQNCE='LOG2'
SEQ2 LOGCHAR APPLID=(APPLICID,AP1),SEQNCE='LOG'

Otherwise, in the preceding example, if sequence LOG had preceded LOG2 in the interpret table, both logon messages LOG and LOG2 would be valid logons to application program AP1. If you use two or more LOGCHAR macroinstructions, they must be arranged so that their SEQNCE fields are in reverse collating order.

**Note:** Although you can define a USERVAR in the interpret table, you can use the interpret table entry in conjunction with a USS table to process an incoming character-coded LOGON. For details, see [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com).  

**ENDINTAB macroinstruction**

The ENDINTAB macroinstruction defines the end of an interpret table. Code one ENDINTAB macroinstruction after one or more LOGCHAR macroinstructions to define the end of an interpret table. You can also follow the ENDINTAB macroinstruction with an assembler language END statement or with CSECTs containing one or more user-written logon-interpret routines.

If you code an assembler language END statement, it must be in the format:

```
END name
```

where `name` is the label of the INTAB macroinstruction and specifies the main entry point.

Follow the ENDINTAB macroinstruction with an assembler language END statement unless the interpret table is to be followed by CSECTs containing one or more user-written APPLID routines, as described in “Coding logon-interpret routines.”

**Coding logon-interpret routines**

You can code logon-interpret routines to validate logons and determine the name of the application program that is to receive the logons. The entry point name must match the `routine name` specified in the `APPLID=(ROUTINE,routine name)` operand in the LOGCHAR macroinstruction. All logon-interpret routines specified in an interpret table must be assembled and link-edited with that interpret table.

The logon-interpret routine interface allows the routine to supply a network-qualified application name for interpreted logons.

If you want the logon-interpret routine to supply a network-qualified application name, you need to change the interpret routine parameter list. If you do not want the routine to supply a network-qualified name, you do not need to change the routine parameter list. See [z/OS Communications Server: SNA Customization](https://www.ibm.com) for details about coding the interpret routine parameter list.

**Requirements for logon-interpret routines**

**Entry from:** VTAM
Entry point:

routine name

Contents of registers at entry:

Register 0:
Length of logon message

Register 1:
Address of first byte of logon message. For LOGON requests, VTAM searches the interpret table again, after USS translation, looking only for the specified APPLID. After USS translation, register 1 contains the address of the first byte of the APPLID.

Register 2:
Address of an 8-byte logical unit name

Register 4:
Address of parameter list for the network identifier and resource name.

Register 13:
Address of a 72-byte save area provided by VTAM

Register 14:
Return address

Register 15:
Address of entry point of this routine

Operation: The logon-interpret routine is run synchronously in pageable storage under the control of VTAM and not under the control of an application program. For the application program to receive the logon, this routine must validate the logon, obtain the name of the application program to receive control, and provide this name to VTAM. Otherwise, the routine specifies that the logon is not valid or that the name of the application program was not found.

The logon-interpret routine must also:

- Save and restore the contents of registers 2–14 when receiving and passing control
- Use re-enterable code (the routine must not store anything within itself or modify itself during execution)
- Perform no I/O operations; an I/O request causes the routine to terminate abnormally.

Because the logon-interpret routine operates at VTAM’s main task dispatching priority, there is a possibility of lockout if a wait requires another task action. The routine gets control in supervisor state with a VTAM storage key, so errors within the routine could cause damage to VTAM or to system control blocks and modules.

Note:

1. You can modify the logon message that is passed to the interpret routine. However, remember these two points:
   - VTAM does not look at the changed storage; it is passed as user data to the application.
   - You should modify with caution, as modification outside the message storage boundaries could result in VTAM outages.
2. All data is addressable only in 24-bit mode.
Contents of registers at exit: Registers 0 and 1 contain the name of the application program (in EBCDIC characters) with which the logical unit is to establish a session:

Register 0:
    First 4 characters of name (left-aligned).

Register 1:
    Last 4 characters of name (left-aligned).

Registers 2–14:
    Restored to condition at entry.

Register 15:
    Return code:
    00    Application program was found and the name is placed in registers 0 and 1.

    Nonzero    Application program was not found and the name is not placed in registers 0 and 1.

If the name of the application program contains fewer than 8 characters, use blanks to provide a name with 8 characters.

Installing and changing interpret tables

Follow these steps to install each interpret table and any user-written routines:
1. Assemble the interpret table and the user-written routines referred to by the LOGCHAR macroinstruction.
   Put the user-written routines in a private call library.
2. Link-edit the interpret table with its associated routines, preferably assigning a module name that matches the name of the interpret table as specified with the INTAB macroinstruction.
   Link-edit the resulting module as a non-executable module into a library identified by the SYS1.VTAMLIB DD statement.
3. Code the assigned name on the LOGTAB operand of the appropriate definition statement to associate the logical unit or logical units with this interpret table (see z/OS Communications Server: SNA Network Implementation Guide for information about specifying the LOGTAB operand in definition statements).

To change an installed interpret table by adding a LOGCHAR macroinstruction, or to replace an installed interpret table, follow these steps:
1. Add or replace the LOGCHAR macroinstruction in the source program.
2. Assemble and link-edit the new interpret table, thus replacing the old module.

The control sections of the interpret table module consist of the interpret table itself and a CSECT for each user-written routine identified by the ROUTINE operand of the LOGCHAR macroinstruction. Individual CSECTs can be extracted when the interpret table is link-edited. Because reprocessing a load module deletes the END statement, use the linkage editor ENTRY control statement to specify the entry point of the new load module. For linkage editor requirements, see z/OS MVS Program Management: User’s Guide and Reference.

To change the name of an installed interpret table, do the following steps:
1. Change the name specified on the INTAB macroinstruction and follow the steps for changing an installed interpret table. When link-editing the interpret table, the module name and entry point must be made to match the new name of the interpret table.

2. Change the LOGTAB operand in all affected definition statements.

3. File the corrected NCP source program and the corrected set of definition statements in the appropriate VTAM library, replacing the old source programs. (The procedure is described in the link to the z/OS Communications Server: SNA Network Implementation Guide.) It is not necessary to do a partial NCP generation because the LOGTAB operand is used only by VTAM.

4. Use the MODIFY TABLE command to associate the new table with the corresponding logical unit, and load the table.

**Interpret table full syntax**

The full syntax for the interpret table follows. The syntax for each operand is repeated in the “Interpret table operand descriptions” on page 1022.

### INTAB

```
INTAB
```

### LOGCHAR

```
LOGCHAR
```

### ENDINTAB

```
ENDINTAB
```
Interpret table operand descriptions

The following information describes the operands you can code for the interpret table.

APPLID

```plaintext
APPLID= (APPLICID, application_name)
(APPLICID, application_name)
(APPLICID, application_name)
(APPLICID, application_name)
```

macroinstructions: LOGCHAR
dependencies: none

Specifies the name of an application program, a logon-interpret routine, or a
USERVAR.

APPLID= (APPLICID, application_name)

Specifies the name of the application program. `application_name` can be any of
the following items:
- ACBNAME of an application program in this host
- `applname` of an application program in this host
- `applname` of an application program in another host
- USERVAR representing an application program

`application_name` can be a network-qualified name. A network-qualified name
takes the form of `netid.application_name`. If `application_name` is network-qualified,
then the network identifier is considered real and is not allowed to change.
The resource name of the network-qualified name is considered generic and
can undergo USERVAR translation.

Note: If ACBNAME and the network name on the APPL definition statement
for the application program are different, then you cannot use a
network-qualified ACBNAME.

APPLID= (ROUTINE, routine_name)

Specifies the routine name of the associated logon-interpret routine. All
logon-interpret routines specified in an interpret table must be assembled and
link-edited with that interpret table. See "z/OS Communications Server: SNA
Customization" for details about coding logon-interpret routines.

APPLID= (USERVAR, uservar_name)

The same as specifying APPLICID.
Specifies optional names for the INTAB, LOGCHAR, and ENDINTAB macroinstructions. If specified, name must be unique and should be used as the operand for the assembler language END statement.

When the INTAB, LOGCHAR, and ENDINTAB macroinstructions are assembled, this name is used to identify the entry point to the interpret table CSECT. You can also use this name as a member name for this interpret table when you use the linkage editor to put the interpret table into the appropriate library. This assignment prevents different names from being used for the same interpret table (one for the entry point to the interpret table CSECT, and another as the name on the LOGTAB operand that other NCP and VTAM definition statements use to refer to this interpret table).

The interpret table is used by terminals that are defined by NCP and VTAM definition statements. The name assigned to the interpret table in the VTAM definition library must therefore be coded in the LOGTAB operand of those statements.

**REMOVE**
Specifies whether VTAM is to remove the first nonblank set of characters from the user logon sequence before it passes the CINIT request unit (RU) to the application program.

**REMOVE=Y**
- Specifies that VTAM is to remove the first nonblank set of characters from the user logon sequence data being processed. The remaining data is left-aligned and padded with blanks on the right.

**REMOVE=N**
- Specifies that VTAM is not to remove any data from the user logon sequence.

For example, if the following information is sent and REMOVE=Y is specified, VTAM removes “IMS10” before it passes the information to the application program in the user data field of the CINIT RU.

IMS10 NAME PASSWORD ===> NAME PASSWORD
**LOGCHAR**

Specifies the required part of a logical unit's logon message.

If 'characters' is a logon message, optional information, which is not specified in the LOGCHAR macroinstruction, can be used by the logon-interpret routine (if the ROUTINE operand is specified), or by an application program's LOGON exit routine.

To specify an apostrophe (') or an ampersand (&) within the logon message, code a double apostrophe ("') or a double ampersand (&&) within the character string. If the terminal user enters the logon message in lowercase and the message is not translated to uppercase (for example, by USS translate table), the value for 'characters' must be coded in lowercase.

Do not specify leading and trailing device-control characters within a character string that is to be interpreted, because the USS facility deletes these characters. Device control characters coded within a logon message will be deleted; therefore, a blank should not be coded for each occurrence of these characters. However, if a character within the logon message will be translated to a blank by the interpret table, code a blank to represent that character.

For non-SNA terminals, the length of the entire logon message (required information plus optional information) is limited to the number of characters that can be accommodated on one line of the terminal screen, up to 255 characters. For SNA terminals, the length is limited to 255 characters (VTAM deletes new-line [NL] characters before the INTRPRET macroinstruction is completed).

For non-SNA terminals, if you do not code SEQNCE in one LOGCHAR macroinstruction and a logon message does not match the character string of the SEQNCE operand in a preceding LOGCHAR macroinstruction in the interpret table, VTAM accepts this logon message and requests logon for this terminal to the application program specified in the LOGCHAR macroinstruction (the one in which SEQNCE is not coded). Therefore, do not place a LOGCHAR macroinstruction at the beginning of the interpret table (immediately after the INTAB macroinstruction) without coding the SEQNCE operand. Otherwise, the remaining logon messages in the interpret table are not compared with the logon message entered by the terminal user.

If you code LOGCHAR for SNA devices, you need to code SEQNCE. Otherwise, VTAM rejects an attempt to logon to the application specified in the APPLID operand when you press the enter key. VTAM issues a sense code stating the enter key RU is not valid because it is null length. Pressing the space key (an acceptable RU), then the enter key allows the logon.

**Note:** If you use two or more LOGCHAR macroinstructions, arrange them so that their SEQNCE fields are in reverse collating order.
Logon mode table

A logon mode table contains one or more sets of session parameters representing rules (session protocols) that describe how a session is to be conducted. You define each table by specifying the following macroinstructions:

- A MODETAB macroinstruction
- One or more MODEENT macroinstructions
- A MODEEND macroinstruction

VTAM has an IBM-supplied logon mode table named ISTINCLM that provides generally accepted session protocols for a basic list of IBM device types. See “Default logon mode table (ISTINCLM)” on page 1117 for the macroinstructions and operands that make up the default logon mode table.

For your information, the code released to you as the VTAM licensed program includes the source language for this table.

The source language is located in SYS1.ASAMLIB.

You can modify or replace the IBM-supplied logon mode table, provided that the modified or replacement table has the same name as the IBM-supplied table and that the IBM-supplied table is deleted. However, you should create a supplemental table instead of deleting the IBM-supplied table, because you might need the IBM-supplied table for problem determination.

You can associate a logon mode table with a logical unit by specifying the table’s name in the MODETAB operand of the logical unit’s definition statement. If you do not designate a logon mode table for a logical unit, VTAM uses the IBM-supplied logon mode table, ISTINCLM. Furthermore, because you cannot code a resource-specific logon mode table for dynamic CDRSCs, VTAM uses the IBM-supplied logon mode table, ISTINCLM.

Regardless of the source of the logon or what session parameters are associated with the logon, the application program decides which session parameters are to be used for the session.

Table 81. Logon mode table macroinstructions and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>MODETAB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>MODEENT</td>
<td>APPNCOS</td>
<td>1030</td>
<td></td>
<td></td>
<td>APPN only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CKEY</td>
<td>1031</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COMPRES</td>
<td>1032</td>
<td></td>
<td>SYSTEM</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COMPROT</td>
<td>1033</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COS</td>
<td>1033</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DCODE</td>
<td>1033</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ENCR</td>
<td>1034</td>
<td></td>
<td>0</td>
<td>data encryption</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ENCRTYP</td>
<td>1035</td>
<td></td>
<td>0</td>
<td>data encryption</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FMPROF</td>
<td>1036</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LANG</td>
<td>1036</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOGMODE</td>
<td>1037</td>
<td></td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
Table 81. Logon mode table macroinstructions and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC</td>
<td>1038</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>data encryption</td>
</tr>
<tr>
<td>MACLNTH</td>
<td>1039</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>data encryption</td>
</tr>
<tr>
<td>MACTYPE</td>
<td>1040</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>data encryption</td>
</tr>
<tr>
<td>PRIPROT</td>
<td>1041</td>
<td></td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>PSERVIC</td>
<td>1041</td>
<td></td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>PSNDPAC</td>
<td>1041</td>
<td></td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RUSIZES</td>
<td>1042</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SECPROT</td>
<td>1043</td>
<td></td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SRCVPAC</td>
<td>1044</td>
<td></td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SSNDPAC</td>
<td>1044</td>
<td></td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>TSPROF</td>
<td>1045</td>
<td></td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>TYPE</td>
<td>1045</td>
<td></td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Logon mode table coding notes
This information contains coding notes for the logon mode table.

**MODETAB macroinstruction**

The MODETAB macroinstruction indicates the beginning of a logon mode table.

**MODEENT macroinstruction**

A MODEENT macroinstruction associates a logon mode name with a set of parameters representing session protocols.

The component description for the device represented by the logical unit should describe the features you want to use and the session protocols required to use those features. See z/OS Communications Server: SNA Programming for information about setting session parameter fields. If a bit is identified as being reserved, set it to 0.

After you have determined what bits must be set, convert the eight bits in each byte of the session parameter field into the two equivalent hexadecimal digits. The value for each operand is the hexadecimal equivalent of the bits in the byte with which the operand is associated. For example, the operand PRIPROT is associated with byte 3 of the session parameter field. If you decide, for example, that the appropriate bit settings for the byte are 1111 0001, code PRIPROT=X’F1’.

The values for the FMPROF and TSPROF operands, can be specified as unframed decimal digits (for example, 4) or as framed hexadecimal digits (for example, X’04’). The other operands require hexadecimal values.

The MODEENT macroinstruction follows the MODETAB or another MODEENT macroinstruction.
**MODEEND macroinstruction**

The MODEEND macroinstruction indicates the end of the logon mode table. Code it after the last MODEENT macroinstruction.

**For independent logical units**

If the session partner of the logical unit for which this mode table applies is an independent logical unit, code only these operands:

- PSNDPAC
- RUSIZES
- SRCVPAC
- SSNDPAC

**Installing a logon mode table**

Each logon mode table must be assembled and link-edited into a library associated with the VTAMLIB DD statement.

With the MODIFY TABLE command, the user can refresh or reload the logon mode table without deactivating the major nodes that should use the new table. See [z/OS Communications Server: SNA Operation](https://www.ibm.com/support/knowledgecenter/en/SANSNA_2.1.0/com.ibm.zos.v2r1.sna.doc/welcome.html) for information about using the MODIFY TABLE command.

**Logon mode table full syntax**

The full syntax for the logon mode table follows. The syntax for each operand is repeated in "Logon mode table operand descriptions" on page 1030.

**MODETAB**
Notes:
1  For the ENCRTYP operand, a value of 0 means the DES encryption. The DES encryption is the current level of cryptography, and is used if the session uses encryption.

Logon mode table operand descriptions
The following information describes the operands you can code for the logon mode table.

APPNCOS
APPNCOS = appn_cos_name

**macroinstructions:** MODEENT
**dependencies:** APPN only

Specifies the name of the APPN Class of Service to be used for the APPN portion of the session that uses this logon mode.

`name` can be 1–8 characters.

Class of Service names are determined as follows:

- The subarea portion of the session uses the Class of Service name coded on the COS operand.
- The APPN portion of the session uses the Class of Service name coded on the APPNCOS operand.
- Using both the COS and APPNCOS operands enables you to use two different Class of Service names for the subarea and APPN portions of the same session.
- If the COS operand is coded but the APPNCOS operand is not coded, the CoS name is used for both the subarea and APPN portions of the session. Therefore, if you want to use the same CoS name to represent identical classes of service in both the subarea and APPN networks, coding the COS operand is sufficient. You do not need to code the APPNCOS operand.
- If the APPNCOS operand is coded but the COS operand is not coded, the default Class of Service name (8 blanks) is used for the subarea portion of the session and the APPNCOS Class of Service name is used for the APPN portion.
- If neither the COS operand nor the APPNCOS operand is coded, then the default Class of Service names (8 blanks for the subarea portion of the session and #CONNECT for the APPN portion) are used.

**CKEY**
**macroinstructions:**

**MODEENT**

Specifies which cryptographic key name should be used for this logical unit. If you do not specify CKEY, any previous specification is unchanged.

**ALTERNATE**

Specifies that cryptographic session keys are generated using the alternate level of the cryptographic key name with the suffix ALT.

You can abbreviate ALTERNATE as ALT.

**PRIMARY**

Specifies that cryptographic session keys are generated using the primary level of the cryptographic key name (that is, the LU name on the LU definition statement or the value of the CKEYNAME).

You can abbreviate PRIMARY as PRIM.

---

**COMPRES**

**macroinstructions:**

**MODEENT**

**dependencies:** none

Specifies how the negotiation of data compression usage should be handled. This operand allows data-compression override specifications to be associated with a logon mode name. The SLU sends this information to the PLU to indicate what level of data compression it wants before the PLU sends the BIND.

**COMPRES=PROHIB**

Specifies that override data should explicitly prevent the use of data compression for sessions using this logon mode.

**COMPRES=REQD**

Specifies that override data should explicitly request data compression, regardless of link characteristics along the path, for sessions using this logon mode, if both session partners can compress data.

REQD causes compression to be specified only when you specify one of the following items:

- The CMPVTAM start option is not set to 0
- One of the following items is not set to 0 on the APPL definition statement:
  - The CMPAPPPLI operand
  - the CMPAPPLO operand
  - Both the CMPAPPPLI operand and the CMPAPPLO operand
If your application is the SLU and you want to use compression, code REQD.

**COMPRES=SYSTEM**

Specifies that no override is needed. For sessions using this logon mode, normal BIND negotiation logic determines whether data compression is used. The default system compression for VTAM application programs when VTAM acts as the SLU is no compression.

**COMPROT**

```
COMPROT --> 0
COMPROT --> common_lu_protocol
```

*macroinstructions:* MODEENT
*dependencies:* none

Represents the common LU protocols (bytes 5 and 6 in the session parameter field) for this logon mode. Any hexadecimal number in the range 0–FFFF can be specified.

**COS**

```
COS --> subarea_cos_name
```

*macroinstructions:* MODEENT
*dependencies:* subarea only

Specifies the name of the Class of Service to be used for a session that uses this logon mode. It should be a 1–8 character name. If COS is omitted, 8 blanks will be stored into the COS field in the mode entry. This blank name is the default COS.

**Note:** Because CP-CP session paths can include subarea VRs, it is strongly recommended that you specify an appropriate COS=subarea_cos_name in your logon mode tables (including the CPSVCMG and CPSVRMGR logon mode table entries in the IBM-supplied logon mode table, ISTINCLM). Otherwise, a blank COS name is used to determine the subarea VR and transmission priority that will be used for the VR portion of the CP-CP session path.

**DCODE**
DCODE = device_code

Specifies the LU device code, and indicates to application programs whether a teletypewriter exchange service (TWX) or a World Trade teletypewriter (WTTY) device is a keyboard and printer or a keyboard and display terminal.

Values for DCODE can be one of the following items:
X'00'  The terminal is a keyboard and printer.
X'80'  The terminal is a keyboard and display.

If a value that is not valid is specified for DCODE, or if the parameter is not specified at all, the device is assumed to be a keyboard and printer.

ENCR

Specifies what type of cryptography the logical unit expects.

Note: ENCR can be specified only for the VTAM data encryption facility.

Any hexadecimal number in the range 0–F (or an equivalent decimal value) can be specified. One or two digit hexadecimal strings can also be used. A 4-bit binary string (for example, ENCR=B'0001') can also be specified.

The ENCR value is converted to a 4-bit string, which is treated as two 2-bit fields. The meanings of the bit settings are as follows:

xx..  Private cryptography field
00..  No private cryptography
01..  Private cryptography used
..xx  VTAM cryptography field
..00  No session-level cryptography
..01  Selective cryptography; the primary logical unit can encipher messages and the secondary logical unit must support cryptography.
..10  Reserved
..11  Mandatory cryptography; all messages on this session will be enciphered and deciphered.

This value is stored in the first four bits of byte 26 in the session parameter field.
Note: The VTAM data encryption facility can use the following cryptography facilities:

- Integrated Cryptographic Service Facility (ICSF/MVS)
- IBM Programmed Cryptographic Facility program product (PCF)
- IBM Cryptographic Unit Support program product (CUSP)
- Translation Security System product (TSS)

**ENCRTYP**

(1)

ENCRTYP = 0

ENCRTYP = TDES24

Notes:

1. For the ENCRTYP operand, a value of 0 means the DES encryption. The DES encryption is the current level of cryptography, and is used if the session uses encryption.

Macroinstructions: MODEENT

Dependencies: VTAM data encryption facility only

Specifies that the session using this log mode entry must use Triple DES encryption. If the session uses encryption, the only valid value that can be coded is Triple DES with 24-byte keys (TDES24). If this operand is not coded or is specified as 0, the DES encryption, which is the current level of cryptography, is used. Triple DES is viable only with the Integrated Cryptographic Service Facility (ICSF/MVS).

FMPROF
**FMPROF**

Specifies the function management profile (byte 1 in the session parameter field) for this logon mode.

Any hexadecimal number in the range 0–FF (or its equivalent decimal value) can be specified. Values 2, 3, 4, 7, and 18 have defined meanings that are described in [z/OS Communications Server: SNA Programming](https://pubs.opengroup.org/onlinepubs/9699919/ applications/zoscommserver/snapgm.pdf).

**LANG**

Defines the language information that will be passed to the VTAM application program. See [z/OS Communications Server: SNA Programming](https://pubs.opengroup.org/onlinepubs/9699919/ applications/zoscommserver/snapgm.pdf) for details about how the application program receives this information.

Any decimal number in the range 0–255 can be specified. Equivalently, any hexadecimal number in the range 0–FF can be specified by using the standard assembler notation `number`.

TSO/VTAM uses the bits 0–7 when processing the value coded for the LANG operand:

- Bit 0
  - TSO/VTAM uses this bit to indicate that devices with extended data stream capability should be queried for language information (double-byte character set capability). See [z/OS Communications Server: SNA Network Implementation Guide](https://pubs.opengroup.org/onlinepubs/9699919/networking/zoscommserver/sna/guides/sna_gmdz044040416a010010400000.html) for additional information about how TSO/VTAM uses the entire value coded for LANG.
  - The value of this bit always comes from the LANG operand of the MODEENT (it cannot be overridden using the LANG or LANGTAB operands on a USS command entered by a terminal user).

- Bit 1–7
  - Specifies the preferred language to be passed to the application program. The application program can use this information for language specific processing.

TSO/VTAM uses this language in several ways and it might affect the following items:

- The language used for TSO/VTAM end-user messages
The filtering done on single-byte character set (SBCS) data streams
The language information returned on the GTTERM macroinstruction

For additional information about how TSO/VTAM uses this language, see z/OS Communications Server: SNA Network Implementation Guide.

For a list of valid values to use for this field, see z/OS Communications Server: SNA Programming. The value of this field can be overridden if the terminal operator enters a LOGON command with the LANG or LANGTAB operand. See "LOGON command" on page 1103 for more information. See 3274 Control Unit Description and Programmer’s Guide for the set of valid character codes for katakana. The 3274 Control Unit must be at microcode level 65 or higher to use the LANG operand.

### Description

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0...</td>
<td>The character set is identified from the code specified in bits 1–7. The device is assumed incapable of double-byte character set (DBCS) processing.</td>
</tr>
<tr>
<td>1...</td>
<td>A query command is sent to the device to determine the character set and DBCS capability. If the character set cannot be determined from the input received from the query, the character set is identified from the code specified in bits 1–7.</td>
</tr>
<tr>
<td>.000</td>
<td>The character set is U.S. English or any single-byte character set (SBCS) in which the valid character codes are compatible with the English codes.</td>
</tr>
<tr>
<td>.001</td>
<td>The character set is katakana, or any SBCS in which the valid character codes are compatible with katakana.</td>
</tr>
<tr>
<td>.111</td>
<td>VTAM performs no character set translation. Any SBCS in which the valid character codes are not compatible with English or katakana (for example, Thai) should specify this value.</td>
</tr>
</tbody>
</table>

**Note:** The only way to establish a kanji character-translation set (or alternate double-byte character set (DBCS)) is to query the device by specifying LANG=X’80’ in the MODEENT macroinstruction. The correct character translation set for the device will be established from the device’s response to the query command.

**LOGMODE**
LOGMODE = __logon_mode_name__

**macroinstructions:**

**MODEENT**

**dependencies:**

none

Specifies the logon mode name to be used as a key for the session parameters in this table entry.

When coding LOGMODE, consider the following information:

- If the LOGMODE parameter is omitted (defaulted) or coded with no name or coded with 1 - 8 blanks, it must be the first entry in the table to be accepted and used. If the LOGMODE parameter is omitted (defaulted) or coded with no name or coded with 1 - 8 blanks on a MODEENT macro that is not the first in the table, the following conditions occur:
  - The table assembles.
  - An assembler message warns you that a blank entry was found and that it is not the first entry in the table.
  - Blank entries are ignored, except when the blank entry is the first entry in the table. Because the first entry is also the default entry, it becomes the default entry for this table but with a blank name.

- If duplicate non-blank LOGMODE names exist in the same table, an assembler error informs you of a “previously defined symbol” and the table assembly fails. Correct the error (remove the duplicate logmode entry or use a different logmode name), and reassemble the table.

- Do not specify LOGMODE names using the following symbols, which are used by the MODETAB, MODEENT and MODEEND macros to generate tables:

  ISTMnnnn ISTSnnnn ISTEnnnn ISTXnnnn

  where nnnn is an integer from 1 to the number of LOGMODE tables (that is, the number of MODETAB macros) contained in the LOGMODE table being assembled.

**MAC**
Specifies the message authentication requirements for this logical unit.

**MAC=REQD**

Specifies that all messages on this session must be authenticated. When MAC=REQD, a session cannot be activated unless the logical unit or host associated with the application program supports message authentication.

**MAC=COND**

Specifies that this application program supports message authentication, which is required if the session partner is MAC capable.

**Note:** If a cryptographic product is not available at session start, a session without message authentication support and cryptographic enhancements is established.

**MAC=None**

Specifies that this logical unit does not engage in message authentication support and cryptographic enhancements supported sessions.
MACLNTH specifies the minimum length, in bytes, of the message authentication code that is to be generated.

MACLNTH is based on MACTYPE, and can be one of the following values:

* MACTYPE
  * MACLNTH
    * CRC: 2 or 4
    * DES: 4, 6, or 8

MACTYPE specifies the method VTAM will use to create or verify a message authentication code.

* MACTYPE=CRC
  Specifies that VTAM use an internal cyclic redundancy check (CRC) algorithm to generate a message authentication code.

* MACTYPE=DES
  Specifies that VTAM use the data encryption standard (DES) services provided by the available Common Cryptographic Architecture (CCA) product to generate a message authentication code.
Specifies optional names for the MODETAB, MODEENT, and MODEEND macroinstructions.

**PRIPROT**

```
PRIPROT = 0
PRIPROT = plu_protocol
```

Specifies the primary LU protocol (byte 3 in the session parameter field) for this logon mode. Any hexadecimal number in the range 0–FF can be specified.

**PSERVIC**

```
PSERVIC = 0
PSERVIC = present_services_prof
```

Specifies the LU presentation services profile and LU presentation services usage field (bytes 13–24, respectively, in the session parameter field) for this logon mode. Specify a 24-digit hexadecimal number, using the bit settings described in z/OS Communications Server: SNA Programming.
PSNDPAC = prim_send_pace_count

macroinstructions: MODEENT
range: 0–63 or X'00'–X'3F'

Specifies the primary send pacing count and whether one- or two-stage pacing is to be used. You can specify decimal or hexadecimal numbers. For decimal numbers, the range is 0–63. For hexadecimal numbers, the pacing count cannot be greater than X'3F'. However, any number in the range X'00'–X'FF' that does not use the reserved bit in position 1 can be specified. The low-order 6 bits of the hexadecimal number correspond to the primary pacing count. The high-order bit is used to specify whether one- or two-stage pacing is to be used. Note that the high-order bit can be set only by coding a hexadecimal value for this operand. See z/OS Communications Server: SNA Programming for more information. Also see z/OS Communications Server: SNA Network Implementation Guide.

RUSIZES

macroinstructions: MODEENT
range: see formula

Specifies a portion of the transmission services usage field (bytes 9 and 10 in the session parameter field) for this logon mode. It specifies the maximum length in bytes of the RUs that the primary logical unit and secondary logical unit can send to each other, as calculated by the formula $m \times 2^n$.

Specify RUSIZES as four hexadecimal digits. The leftmost two digits apply to the secondary logical unit, and the rightmost two digits apply to the primary logical unit.

The format is the same for both sets of digits. The first digit is the mantissa ($m$), and it must be in the range X'8'–X'F'. The second digit is the exponent ($n$), and it must be in the range X'0'–X'F'.

For example, RUSIZES=X'96A8' specifies that the secondary logical unit can send a maximum length of $9 \times 2^6$ (or 576) bytes and that the primary logical unit can send a maximum of $10 \times 2^8$ (or 2560) bytes.

Table 82 on page 1043 shows the exponents, mantissas, and the values for RUSIZES.

Note: For non-APPC sessions, if both the mantissa and exponent are set to 0 or if RUSIZES is not specified, then the default size is used. The default is obtained as follows:
• If the BIND is an extended BIND, the default is X'FFFF'.
• If the BIND is not an extended BIND, the default is the value of the MAXLURU start option.
For APPC sessions, if the RU size value is zero or less than 256, the value is overridden and set to 256.

Table 82. Exponent and mantissa values for each RU size

<table>
<thead>
<tr>
<th>Exp (n)</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>1</td>
<td>16</td>
<td>18</td>
<td>20</td>
<td>22</td>
<td>24</td>
<td>26</td>
<td>28</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>32</td>
<td>36</td>
<td>40</td>
<td>44</td>
<td>48</td>
<td>52</td>
<td>56</td>
<td>60</td>
</tr>
<tr>
<td>3</td>
<td>64</td>
<td>72</td>
<td>80</td>
<td>88</td>
<td>96</td>
<td>104</td>
<td>112</td>
<td>120</td>
</tr>
<tr>
<td>4</td>
<td>128</td>
<td>144</td>
<td>160</td>
<td>176</td>
<td>192</td>
<td>208</td>
<td>224</td>
<td>240</td>
</tr>
<tr>
<td>5</td>
<td>256</td>
<td>288</td>
<td>320</td>
<td>352</td>
<td>384</td>
<td>416</td>
<td>448</td>
<td>480</td>
</tr>
<tr>
<td>6</td>
<td>512</td>
<td>576</td>
<td>640</td>
<td>704</td>
<td>768</td>
<td>832</td>
<td>896</td>
<td>960</td>
</tr>
<tr>
<td>7</td>
<td>1024</td>
<td>1152</td>
<td>1280</td>
<td>1408</td>
<td>1536</td>
<td>1664</td>
<td>1792</td>
<td>1920</td>
</tr>
<tr>
<td>8</td>
<td>2048</td>
<td>2304</td>
<td>2560</td>
<td>2816</td>
<td>3072</td>
<td>3328</td>
<td>3584</td>
<td>3840</td>
</tr>
<tr>
<td>9</td>
<td>4096</td>
<td>4608</td>
<td>5120</td>
<td>5632</td>
<td>6144</td>
<td>6656</td>
<td>7168</td>
<td>7680</td>
</tr>
<tr>
<td>A</td>
<td>8192</td>
<td>9216</td>
<td>10240</td>
<td>11264</td>
<td>12288</td>
<td>13312</td>
<td>14336</td>
<td>15360</td>
</tr>
<tr>
<td>B</td>
<td>16384</td>
<td>18432</td>
<td>20480</td>
<td>22528</td>
<td>24576</td>
<td>26624</td>
<td>28672</td>
<td>30720</td>
</tr>
<tr>
<td>C</td>
<td>32768</td>
<td>36864</td>
<td>40960</td>
<td>45056</td>
<td>49152</td>
<td>53248</td>
<td>57344</td>
<td>61440</td>
</tr>
<tr>
<td>D</td>
<td>65536</td>
<td>73728</td>
<td>81920</td>
<td>90112</td>
<td>98304</td>
<td>106496</td>
<td>114688</td>
<td>122880</td>
</tr>
<tr>
<td>E</td>
<td>131072</td>
<td>147456</td>
<td>163840</td>
<td>180224</td>
<td>196608</td>
<td>212992</td>
<td>229376</td>
<td>245760</td>
</tr>
<tr>
<td>F</td>
<td>262144</td>
<td>294912</td>
<td>327680</td>
<td>360448</td>
<td>393216</td>
<td>425984</td>
<td>458752</td>
<td>491520</td>
</tr>
</tbody>
</table>

SECPROT
SECPROT = slu_protocol

Specifies the secondary LU protocol (byte 4 in the session parameter field) for this logon mode. Any hexadecimal number in the range 0–FF can be specified.

**SRCVPAC**

SRCVPAC = sec_receive_pace_count

Specifies the secondary receive pacing count and whether fixed or adaptive pacing is to be used. You can specify decimal or hexadecimal numbers. For decimal numbers, the range is 0–63. For hexadecimal numbers, the pacing count cannot be greater than X'3F'. However, any number in the range X'00'–X'FF' that does not use the reserved bit in position 1 can be specified. The low-order 6 bits of the hexadecimal number correspond to the secondary receive pacing count. The high-order bit is used to specify whether fixed or adaptive pacing is to be used. Note that the high-order bit can only be set by coding a hexadecimal value for this operand. See [z/OS Communications Server: SNA Programming](https://www.ibm.com/support/pages/zos-v2r10) for more information. Also see the [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/docs/en/zos/2.4.0?topic=communication-manuals).

**SSNDPAC**
Specifies the secondary send pacing count and whether one- or two-stage pacing is to be used. You can specify decimal or hexadecimal numbers. For decimal numbers, the range is 0–63. For hexadecimal numbers, the pacing count cannot be greater than X'3F'. However, any number in the range X'00'–X'FF' that does not use the reserved bit in position 1 can be specified. The low-order 6 bits of the hexadecimal number correspond to the secondary pacing count. The high-order bit is used to specify whether one- or two-stage pacing is to be used. Note that the high-order bit can be set only by coding a hexadecimal value for this operand. See z/OS Communications Server: SNA Programming for more information. Also see the z/OS Communications Server: SNA Network Implementation Guide.

TSPROF

Specifies the transmission services profile (byte 2 in the session parameter field) for this logon mode. Any hexadecimal number in the range 0–FF (or its equivalent decimal value) can be specified. Values 2, 3, 4, and 7 have defined meanings that are described in z/OS Communications Server: SNA Programming.
Specifies the type of BIND command represented by this logon mode entry (bits 4–7 of byte 0 in the session parameter field). While only the values of 0 and 1 are defined, any hexadecimal value in the range 0–F can be specified. If 0 is specified, the logon mode entry represents session parameters that could be sent in a negotiable BIND. If 1 is specified, a non-negotiable BIND should be sent.

**Note:** LU 6.2 sessions require a negotiable bind.

Specifying TYPE=0 (indicating a negotiable BIND entry) does not cause VTAM to necessarily send a negotiable BIND to the secondary logical unit. The logon mode entry can indicate only that the secondary logical unit can support a negotiable BIND. The application program issuing the OPNDST macroinstruction must specify PROC=NEGBIND (in the node initialization block (NIB)) before VTAM will send a negotiable BIND command. For more information on negotiable BIND, see [z/OS Communications Server: SNA Programming](https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.2.1/zos222/COMVDV0/zos222_100002b.htm).

### Message-flooding prevention table

The message-flooding prevention facility identifies and suppresses duplicate messages that are issued in rapid succession. This reduces the possibility of duplicate messages flooding the operator console and concealing critical information. VTAM bases its suppression on the time interval since the message was last issued and on the similarity of variable text in the original and subsequent message.

For each candidate message, the message-flooding prevention table contains the criteria that must be met before VTAM suppresses duplicate messages and whether suppressed messages are sent to the hardcopy log. The suppression criteria include the amount of time between the original and subsequent messages, and an indication of which variable text fields are to be compared. If the message is reissued within the specified time interval and the specified variable text fields contain the same information, VTAM suppresses the message.

**Table 83. Message-flooding prevention table macroinstructions and operands**

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>FLDTAB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>FLDENT</td>
<td>MESSAGE</td>
<td>1052</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HARDCOPY</td>
<td>1050</td>
<td></td>
<td>YES</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LIST</td>
<td>1050</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TIME</td>
<td>1053</td>
<td></td>
<td>30</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>VARTEXT</td>
<td>1053</td>
<td></td>
<td>YES</td>
<td></td>
</tr>
</tbody>
</table>

Name FLDEND
Table 83. Message-flooding prevention table macroinstructions and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
</table>

**Legend:**

R Required operand.

---

**Message-flooding prevention table coding notes**

This information contains coding notes for the message-flooding prevention table.

**FLDTAB macroinstruction**

The FLDTAB macroinstruction begins the message-flooding prevention table.

**FLDENT macroinstruction**

One or more FLDENT macroinstructions specify which messages are to be suppressed.

**FLDEND macroinstruction**

The FLDEND macroinstruction ends the table.

**Installing a message-flooding prevention table**

Store the set of macroinstructions as a member of SYS1.VTAMLST. Use the member name as the name of the table.

No assembly or link-edit is required to install this table. The table is dynamically built when:

- VTAM is initialized, if the FLDTAB start option has been specified
- An operator issues a MODIFY TABLE command specifying the table name

The MODIFY TABLE command lets you replace or modify the message-flooding prevention table without affecting VTAM operations. See [z/OS Communications Server: SNA Operation](https://www.ibm.com/support/knowledgecenter/SSSLBS_8.5.1/com.ibm.zos.v2r11.snn.sna.doc/) for more information about this command.

To change the IBM-supplied message-flooding prevention table:

1. Copy the IBM-supplied table (ISTMSFLD).
2. Rename the copy.
3. Modify the copy.

VTAM uses the internal version of the IBM-supplied message-flooding prevention table if either of the following conditions is true:

- No table has been specified (FLDTAB start option not specified)
- The specified table (on the FLDTAB start option) has failed

You can display the name of the message-flooding prevention table by specifying ISTNOP, the name of the application program that represents the network operator, on the DISPLAY ID command. See [z/OS Communications Server: SNA Operation](https://www.ibm.com/support/knowledgecenter/SSSLBS_8.5.1/com.ibm.zos.v2r11.snn.sna.doc/).
Considerations for message-flooding prevention

The following information shows the rules for suppressing messages through the message-flooding prevention table:

- The message is recorded in the VTAM internal-trace table.
- The message is constructed, but is not transmitted to the operator. The message might be routed to other areas. For example, if you coded PPOLOG=YES in your start options, the message will be routed to the PPO (primary program operator) log.
- If the first line of a message group is suppressed, all messages in the group are also suppressed.
- A message within a message group cannot be suppressed by specifying that message on the FLDEMT macroinstruction.
- USSMSG00 – USSMSG14 are not suppressed.

The message-flooding prevention process can cause some undesirable circumstances to occur. For example:
1. A command might be entered and result in a message to the operator. If the message is a member of the flooding-prevention table and the command is reentered within the designated time-span, the message might not appear the second time as expected.
2. If the header of a message group is suppressed, all messages in the group are also suppressed. The information in these messages might differ from the last issuance, but they are not reissued.

Example of message suppression

The following example demonstrates when a message will be suppressed to prevent flooding of the operator console. This suppression occurs without regard to the suppression level assigned to the message in the USS table and without regard to the message suppression level assigned to VTAM using the MODIFY SUPP command or the SUPP start option.

Message IST367I is a member of the message-flooding prevention table:

IST367I NO STORAGE TO DEFINE NODE nodename CONFIG configname

This message is issued when VTAM does not have enough storage to expand the host-node table. Therefore, a new host element address could not be assigned for the defined resource nodename in the configuration configname. The specifications for the message in the table are:

- Time-span: 30 seconds
- Variable fields: configname

Without message flooding prevention, this message is displayed every time it is issued without regard to time-span or configname.

Time 1: IST367I NO STORAGE TO DEFINE NODE NODE1 CONFIG CONFIG1
Time 2: IST367I NO STORAGE TO DEFINE NODE NODE1 CONFIG CONFIG2
Time 3: IST367I NO STORAGE TO DEFINE NODE NODE2 CONFIG CONFIG1
Time 35: IST367I NO STORAGE TO DEFINE NODE NODE1 CONFIG CONFIG1

With message flooding prevention, this message at Time 3 would not be displayed for the following reasons:
- configname is the same as configname at Time 1.
• The message at Time 3 was issued within 30 seconds of the message at Time 1.
The message at Time 2 would be displayed because configname is not the same.

The message at Time 35 would be displayed since it was issued after the 30 second time-span.

Time 1: IST367I NO STORAGE TO DEFINE NODE NODE1 CONFIG CONFIG1
Time 2: IST367I NO STORAGE TO DEFINE NODE NODE1 CONFIG CONFIG2
Time 35: IST367I NO STORAGE TO DEFINE NODE NODE1 CONFIG CONFIG1

Message-flooding prevention table full syntax
The full syntax for the message-flooding prevention table follows. The syntax for
each operand is repeated in "Message-flooding prevention table operand
descriptions" on page 1050.

FLDTAB

FLDENT

HARDCOPY

HARDCOPY

FLDEND
Message-flooding prevention table operand descriptions
The following information describes the operands you can code for the message-flooding prevention table.

**HARDCOPY**

- **HARDCOPY=YES**
  - Indicates that the message is suppressed from the operator console, but is sent to the system hardcopy log.

- **HARDCOPY=NO**
  - Indicates that the message is suppressed from the operator console and the system hardcopy log.

**LIST**
Specifies from one to five messages that are linked with the first message in a group to determine whether the message group should be suppressed. The messages you specify on LIST must be present as FLDENT entries in the message-flooding prevention table.

With the LIST operand, you can prevent messages that are not duplicates from being suppressed. For instance, if you specify message IST663I on the MESSAGE operand, and do not code the LIST operand, then all message groups beginning with IST663I are suppressed, even if the groups are not duplicates. An example of this problem follows:

```
IST663I  CDINIT REQUEST FROM IBMGW FAILED , SENSE=087D0001
IST664I  REAL OLU=USIBMEL.ELNA9171 REAL DLU=IBMIN.IBMN2CC9
(intervening messages removed)
IST663I  CDINIT REQUEST FROM IBMGW FAILED , SENSE=087D0001
IST664I  REAL OLU=USIBMPO.TF2LAADV REAL DLU=IBMIN.IBMJ1EP8
(intervening messages removed)
IST663I  CDINIT REQUEST FROM IBMGW FAILED , SENSE=087D0001
IST664I  REAL OLU=APPN.IKPASS0A REAL DLU=APPN.IKPASSHQ
(intervening messages removed)
```

In the above example, the three IST663I messages are treated as duplicates. The first IST663I multi-line message is kept and the others suppressed. However, the messages are not duplicates because they have different LU partner names in the IST664I messages.

If the second and third IST663I messages were suppressed, it would take you longer for problem determination, because of having to search trace tapes to find the failures that would have been displayed through the message.

But, if you were to specify IST664I on the LIST operand (LIST=0664) of the FLDENT entry for IST663I, VTAM would evaluate the IST664I messages for duplication, and not suppress them because they are not duplicates.

**Note:**

1. The values of the LIST operand are the message IDs of the other FLDENT entries (for instance, LIST=0664). All LIST references must be to other messages with corresponding FLDENT entries in the message-flooding prevention table.
2. A LIST entry which is the same as its FLDENT entry is not permitted.
3. If the FLDENT entry indicates a LIST, but the message is being issued as a single-line message, the FLDENT entry is ignored for the one message.
4. If the FLDENT entry indicates a LIST with multiple messages and all of the messages are not present in the message group being issued by VTAM, the FLDENT entry is ignored for the current group.
5. If the FLDENT entry indicates a LIST with multiple messages, the messages do not have to be present within the multiple line write-to-operator group in the order indicated in the FLDENT entry.
6. You cannot specify a particular message in more than one LIST operand.
7. The HARDCOPY and TIME operands are ignored if either operand is coded for a message that is not a single-line message or the first message within a group. For the "later messages", the HARDCOPY and TIME values for the first message apply to the entire group.

**MESSAGE**

```
MESSAGE = message_id
```

*macroinstruction: FLDENT*

Specifies a 1-digit to 4-digit identifier that corresponds to the message number to be suppressed. You can pad an identifier that has fewer than four digits with leading zeroes. For example, MESSAGE=118 and MESSAGE=0118 are viewed as the same.

**Note:** Messages do not have to be in numerical order in the table. However, for better performance, list the most common messages first.

**name**

```
[  name  ]
```

```
Specifies the name of the macroinstruction being defined.

For FLDTAB and FLDEN:  
name is optional and has no function. If used, it is recommended that name match the SYS1.VTAMLST member name containing these macroinstructions.

For FLDEN:  
Specifies the required and unique 1–8 character name of the table entry.

**TIME**

```
TIME=30
```

Specifies the amount of time, in seconds, that VTAM continues to suppress duplicate messages. The time interval begins when the first message is issued.

**VARTEXT**

```
VARTEXT=YES
```

Specifies which variable text fields in the specified message are used as the criteria for suppression.

Code a VARTEXT setting or allow the default for each variable text field in the candidate message. For example, if you code

```
VARTEXT=(NO,,NO)
```

the first and third variable text fields in the message are not used as criteria for suppression. However, the second field is used because the default value YES is used.

The number of variable text fields can be determined by referring to [z/OS Communications Server: SNA Messages](https://www.ibm.com/support/docview.wss?uid=swg21208361).

VARTEXT is not applicable for messages that have no variable text. If the candidate message does not have variable text, and you code VARTEXT, VTAM issues an error message.

**VARTEXT=NO**

Indicates that VTAM compares the field’s value in the duplicate message to the
field’s value in the original message when determining whether to suppress the duplicate message. If the fields contain different values, the message is not suppressed.

**VARTEXT=NO**

Indicates that VTAM does not compare the field values when determining whether to suppress the duplicate message.

---

**Model name table**

The model name table contains model names that can be passed to VTAM application programs in their LOGON exits. VTAM application programs use the model names to create dynamic definitions for their session-partner resources.

A model definition is usually the starting point for a dynamic resource definition. The model name passed by VTAM enables the application program to select the proper model definition to use.

Operands on an SLU’s resource definition associate that SLU with the proper model name data. The MDLTAB operand specifies the name of the model name table to be used, and the MDLENT operand specifies the proper entry within the table. These operands are described for each LU-type resource in [Chapter 2, “Major nodes,” on page 19](#).

---

### Table 84. Model name table macroinstructions and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>MDLTAB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>MDLENT</td>
<td>MODEL</td>
<td>1056</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>MDLPLU</td>
<td>PLU</td>
<td>1057</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODEL</td>
<td>1056</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

R  Required operand.

---

**Model name table coding notes**

This information contains coding notes for the model name table.

**MDLTAB macroinstruction**

The MDLTAB macroinstruction indicates the beginning of the model name table.

**MDLENT macroinstruction**

The MDLENT macroinstruction indicates the start of a model name entry and optionally sets up the default model name. The end of the entry is indicated either by another MDLENT macroinstruction or by an end of file.
**MDLPLU macroinstruction**

The MDLPLU macroinstruction defines model name data for a specific PLU within the table entry started by the previous MDLENT macroinstruction.

**Installing a model name table**

Store the model name table macroinstructions in SYS1.VTAMLST, and use the member name as the name of the table.

No assembly or link-edit is required to install the table. The table will be dynamically built upon the first activation of any resource that has a defined association to the table. When the last resource using it is deactivated, the table is deleted and its storage is freed.

IBM does not supply a default model name table.

**Model name table full syntax**

The full syntax for the model name table follows. The syntax for each operand is repeated in "Model name table operand descriptions" on page 1056.

```
MDLTAB
   name
```

MDLENT
Model name table operand descriptions

The following information describes the operands you can code for the model name table.

**MODEL**

```
MODEL = model_name
```

Macroinstructions: MDLENT, MDLPLU

Specifies the 1–8 character model name expected by the subsystem for the terminal.

The model name is passed only when VTAM initiates the session with the application program, for example, when the SLU has the LOGAPPL = parameter coded in its definition. When the SLU initiates the session with the application program by means of the LOGON command or a USS command, the model name to be passed to the application program must be explicitly specified on the MODEL = parameter of the LOGON command or USS command.

If you code MODEL on the MDLENT macroinstruction, the `model_name` specified is a default to be used with any PLU that is not specified in a following MDLPLU macroinstruction.

If you omit MODEL on the MDLENT macroinstruction, no default data is created. Consequently, any PLU not specified in one of the MDLPLU macroinstructions within this entry will not receive any model name data unless the MODEL operand is overridden during logon.

If you omit MODEL on the MDLPLU macroinstruction, no `model_name` is sent to the application or to the subsystem during session initiation.

**name**
**macroinstructions:** MDLTAB, MDLPLU

Specifies the 1–8 character name for the macroinstructions.

**PLU**

```
   PLU-name
```

**macroinstructions:** MDLPLU

Specifies the 1–8 character name of the PLU as it is known in the SLU’s network.

---

**Subarea mapping table**

The subarea mapping (SAMAP) table is used to supply additional NCP connectivity information to APPN Topology and Routing Services, which can then use that information in an effort to optimize routes through the CNN. See [z/OS Communications Server: SNA Operation](https://www.ibm.com/support/docview.ws/docid/38325) and [z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/docview.ws/docid/38326) for more information about Subarea Mapping Tables.

[Figure 7 on page 1058](#) illustrates definition of a SAMAP mapping table.
### Table 85. Definition statements for SAMAP table

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=SAMAP</td>
<td>1060</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAx</td>
<td>MAPSTO</td>
<td>SUBAREA=SAxx</td>
<td>1060</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>NETID=netid</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:
- **R** Required operand.

---

**Subarea mapping table coding notes**

This information contains coding notes for the subarea mapping table.

**VBUILD definition statement**

Code a VBUILD definition statement as the first statement in a subarea mapping table.

**MAPSTO definition statement**

Code a MAPSTO definition statement to define a subarea mapping.

**Subarea mapping table full syntax**

The full syntax for the subarea mapping table follows. The syntax for each operand is repeated in “Subarea mapping table operand descriptions” on page 1059.

**VBUILD**

```plaintext
SAMAP1  VBUILD  TYPE=SAMAP
SA10   MAPSTO  SUBAREA=SA20
SA11   MAPSTO  NETID=NETB
```
Subarea mapping table operand descriptions

The following information describes the operands you can code for the subarea mapping table.

**name**

Assigns a name to the subarea mapping table.

**NETID**
NETID

statements: MAPSTO
dependencies: none

Assigns a name to the NETID.

SAxx

statements: MAPSTO
dependencies: none

Specifies a decimal CNN subarea number. The SA prefix is required.

SUBAREA

statements: MAPSTO
dependencies: none

Specifies a subarea number.

TYPE
VTAM provides a filter to reduce the amount of session awareness (SAW) data that is passed to communication network management (CNM) application programs, such as the NetView program. Using the SAW data filter, only data for sessions that match predefined PLU-SLU name combinations are sent over the CNM interface to the CNM application program. The PLU-SLU combination definitions can contain pattern-matching characters so that one definition can apply to a variety of sessions.

Note: If the VTAM SAW data filter definition contains SSCP-PU SAW data filter definitions, the CNM application (for example, the NetView program) does not receive any SSCP-PU SAW data from VTAM and cannot collect response time monitor (RTM) data for active physical units.

The KCLASS and MAPSESS macroinstructions define the session monitor filter. If you use the NetView program, you can copy your existing session monitor filter definitions and make some changes to create equivalent VTAM filters. You must place the KEEPMEM START macroinstruction at the beginning of the filter definitions and add KEEPMEM STOP and END macroinstructions at the end of the existing definitions. For continuation lines, use the conventions described in "Rules for coding comments" on page 13. VTAM ignores any NetView operands that do not apply to the VTAM SAW data filter.

VTAM includes a default filter, ISTMGC10 in VTAMLIB, that allows data for all sessions to be passed across the CNM interface. ISTMGC10 needs to be assembled. See "Restrictions on use of assembler features" on page 16 for information on assembling modules.

You can modify ISTMGC10 or replace it with one of your own using the MODIFY TABLE command. See z/OS Communications Server: SNA Operation for more information about the MODIFY TABLE command.

You can display the name of the SAW data filter table by specifying ISTNOP, the name of the application program that represents the network operator, on the DISPLAY ID command. See z/OS Communications Server: SNA Operation.

Table 86. Session awareness data filter macroinstructions and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>KEEPMEM</td>
<td>START-STOP</td>
<td>[1066]</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>KCLASS</td>
<td>SAW</td>
<td>[1065]</td>
<td>R</td>
<td>YES</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>MAPSESS</td>
<td>KCLASS</td>
<td>[1065]</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRINET</td>
<td>[1064]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Session awareness data filter macroinstructions and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRI</td>
<td>1064</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC</td>
<td>1065</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SECNET</td>
<td>1066</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

- **END**

### Session awareness data filter coding notes

This information contains coding notes for the session awareness data filter.

**KEEPMEM macroinstruction**

The KEEPMEM macroinstruction defines the beginning or end of a set of SAW data filter definition macroinstructions.

**KCLASS macroinstruction**

The KCLASS macroinstruction defines whether VTAM passes SAW data over the CNM interface for a particular session. The MAPSESS macroinstruction refers to the names on the KCLASS macroinstructions to instruct VTAM on how to treat SAW data for sessions that match the MAPSESS PLU-SLU name specifications.

**MAPSESS macroinstruction**

The MAPSESS macroinstruction specifies a PLU-SLU name combination and specifies which KCLASS definition VTAM should use to determine if SAW data should be passed over the CNM interface.

**Note:**

1. The maximum number of MAPSESS macroinstructions you can code is 2000.
2. If the PRI operand has the host CDRM name coded, an additional MAPSESS macroinstruction with PRI=VTAM must be specified. PRI=VTAM is used to obtain saw data before the activation of the host CDRM major node.

**END macroinstruction**

The END macroinstruction indicates the end of a SAW data filter.

### Session awareness data filter full syntax

The full syntax for the session awareness data filter follows. The syntax for each operand is repeated in "Operand descriptions" on page 1063.

**KEEPMEM**
Operand descriptions
The following information describes the operands you can code for the saw data filter.

KCLASS
KCLASS—kclass_name

macroinstructions: MAPSESS
dependencies: none

Specifies the name of the preceding KCLASS macroinstruction. This operand indicates whether SAW data is to be transmitted over the CNM interface for a particular session.

name

macroinstructions: MAPSESS, END

Specifies 1–8 character names for the macroinstructions.

PRINET

macroinstructions: MAPSESS

dependencies: none

Specifies a network ID for the PLU.

PRI
PRI = pluname_pattern

macroinstructions: MAPSESS
dependencies: none

Specifies the pattern used to match the PLU name. It can contain 1–8 characters from the set (A–Z, 0–9, @, $, ?, *). A question mark (?) matches any single letter or digit. An asterisk (*) matches more than one letter or digit, and can be used only as the last character in the string. The pattern can be used to compare names only up to 8 characters in length.

The following information shows some examples:
• T3??* matches T3A7, T3B7A, or T3C7ABCD. It does not match T3AA7A.
• T3*? is not a valid pattern.
• T3??? matches T3AB7 or T3CD7. It does not match T3A7.

SAW

macroinstructions: KCLASS
dependencies: none

Specifies whether to pass SAW data over the CNM interface.

SEC
SEC = sluname_pattern

macroinstructions: MAPSESS
dependencies: none

Specifies the pattern used to match the SLU name. See the description of the PRI operand for details on the pattern-matching rules.

Note: SEC and PRI are optional. If one is not specified, the default value of * is taken.

SECNET = slu_network_id

macroinstructions: MAPSESS
dependencies: none

Specifies a network ID for the SLU.

START-STOP

START

STOP

macroinstructions: KEEPMEM
dependencies: none

Specifies whether this macroinstruction starts or stops a set of session awareness data filter definition statements.

SAW sense filter

The SAW sense filter contains a list of session initialization failure sense codes that are not to be reported to a communication network management (CNM) application program collecting SAW data.

The SNSCODE definition statement defines the sense codes that are excluded from the SAW data being reported to the CNM application program. For more information about how to use the SAW sense filter, see z/OS Communications Server: SNA Network Implementation Guide.

Table 87. SAW sense filter definition statements and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>VBUILD</td>
<td>TYPE=SNSFILTR</td>
<td>1068</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>SNSCODE</td>
<td>SENSE</td>
<td>1068</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 87. SAW sense filter definition statements and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition statement</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Sift effect</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
</table>

Legend:
R    Required operand.

**Coding notes**

This information contains coding notes for the SAW sense filter.

**VBUILD definition statement**

Code one VBUILD TYPE=SNSFILTR definition statement to indicate that this set of definition statements is for a SAW sense filter.

**SNSCODE definition statement**

Code one SNSCODE definition statement for each session initiation failure sense code to be filtered from SAW data sent to a communication network management (CNM) application program.

**Full syntax**

The full syntax for the SAW sense filter follows. The syntax for each operand is repeated in "Operand descriptions" on page 1068.

**VBUILD**
Operand descriptions

The following information describes the operands you can code for the SAW data filter.

name

Specifies the optional name of definition statement.

SENSE

Specifies the session initiation failure sense code that is to be filtered out of the SAW data sent to CNM application program receiving SAW data. Specify either the two-byte or four-byte session initiation failure sense code (in hexadecimal). If one or three bytes of data is specified, the data is ignored. Any sense code specified that is not related to session initiation failures is also ignored.

Note: If only 2 bytes are specified, the whole category specified by the 2 bytes will be filtered.

TYPE
Unformatted system services tables

VTAM uses unformatted system services (USS) tables to:
• Issue messages to the VTAM operator console or a program operator application
• Process VTAM operator commands
• Issue messages to a terminal operator user
• Process commands received from a terminal operator user

Other messages, such as TSO/VTAM messages, are provided through other facilities. For an explanation of these messages, see z/OS Communications Server: SNA Messages.

VTAM uses two types of USS tables to define messages and commands:

Session-level table
Contains:
• Definitions for terminal user commands (such as LOGON) that can be received from a terminal.
• Messages that VTAM sends to a terminal.
• A translation table that is used for character-coded input from the terminal.

See “Session-level USS table (ISTINCDT)” on page 1132 for an example of a session-level USS table.

Operation-level table
Contains USS commands (such as DISPLAY ROUTE) that can be received from the VTAM operator or a program operator application and messages issued in response to those commands. See “Operation-level USS table (ISTINCNO)” on page 1133 for an example of an operation-level USS table.

IBM supplies default USS tables with the VTAM product. The source language for these tables is contained in SYS1.ASAMPLIB. Table 88 on page 1070 describes the default tables.
Table 88. Description of the IBM-supplied USS tables

<table>
<thead>
<tr>
<th>Table name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTINCDT</td>
<td>Default session-level USS table used by terminal operator users. Contains the following items:</td>
</tr>
<tr>
<td></td>
<td>• The following terminal operator commands:</td>
</tr>
<tr>
<td></td>
<td>– LOGON</td>
</tr>
<tr>
<td></td>
<td>– LOGOFF</td>
</tr>
<tr>
<td></td>
<td>– IBMTEST</td>
</tr>
<tr>
<td></td>
<td>• USSMSGxx (where xx equals 1–9 and 11–14) that are sent to a terminal operator. See [z/OS Communications Server: SNA Messages] for detailed descriptions of USS messages.</td>
</tr>
<tr>
<td></td>
<td>• Translation table that is used for character-coded input from the terminal. The translation table, named STDTRANS, converts lowercase characters to uppercase and converts horizontal tabs to spaces.</td>
</tr>
<tr>
<td>ISTINCNO</td>
<td>Default operation-level USS table used by the VTAM operator (commands and messages) and program operator applications (commands). Contains the following items:</td>
</tr>
<tr>
<td></td>
<td>• Operands and defaults for the following VTAM operator commands:</td>
</tr>
<tr>
<td></td>
<td>– All DISPLAY commands</td>
</tr>
<tr>
<td></td>
<td>– All MODIFY commands</td>
</tr>
<tr>
<td></td>
<td>– All VARY commands</td>
</tr>
<tr>
<td></td>
<td>For more details about these commands, see [z/OS Communications Server: SNA Operation].</td>
</tr>
<tr>
<td></td>
<td>• VTAM operator messages listed in the [z/OS Communications Server: SNA Messages].</td>
</tr>
<tr>
<td>ISTCFCMM</td>
<td>Default operation-level USS table for a program operator application containing operation-level VTAM messages.</td>
</tr>
</tbody>
</table>

Customizing USS tables

You can customize VTAM messages, commands, and the translation table by coding your own USS table. For example, you can change messages to provide non-English text, change characteristics of a message, or change the syntax or default values for a command.

Any changes to VTAM messages or commands should be made with supplementary user-defined USS tables. The IBM-supplied USS tables should not be changed or removed because they are used to define all commands and messages that are not defined in a user-written table. The name of the user-defined USS table should be different from any of the default tables.

If you plan to make extensive changes in your user-defined USS table, you might find that the easiest way to generate the table is to copy the IBM-supplied USS tables and make changes to the new copies.

**Note:** Because some program operator applications depend on VTAM messages, changes to an operator message could disrupt the functioning of a program operator application.

A USS table can be associated with a particular resource by the following items:
VTAM operator

- USSTAB operand on a USS command
- USSTAB operand on the logical unit's definition statement

**For more information about the MODIFY TABLE command, see** [z/OS Communications Server: SNA Operation](https://www.ibm.com/servers/resourcelink/systm5000/w3snavhome.nsf/Content/csm2226.html).

Program operator

- USSTAB operand on the application definition statement
- SSCP FM operand on the application definition statement

VTAM messages

To customize VTAM messages, you only need to code the message or the part of the message you wish to redefine. VTAM will use the user-defined supplementary USS message table in conjunction with the IBM-supplied default USS tables to locate the complete message definition. Several USS tables might be searched before locating a complete message definition. The tables are searched in a specific order (see "Order of USS table use").

For more information on coding VTAM messages in a USS table, see "Customizing messages" on page 1098.

VTAM commands

Although a VTAM message need only be partially defined in a supplementary user-defined table, a command must be completely defined in a supplementary user-defined USS table. VTAM uses either the user-defined table or the IBM-supplied default USS tables to locate the complete command definition.

For more information on coding VTAM commands in a USS table, see "Defining VTAM operator commands" on page 1101.

Order of USS table use

When VTAM requires a message, command, or input translation table that is defined in the USS tables, it searches the USS tables in a specific order. The search order depends on the destination or source (VTAM operator, program operator, or terminal operator) and the target search item (a command, message, or translation table).

Tables [Table 89](#) through [Table 92](#) show the search order used by VTAM for each target search item (messages, commands, and translation tables). Within each table the search order is shown for the first table searched to the last (top to bottom) for each destination (terminal operator, VTAM operator, and program operator).

For VTAM messages

VTAM uses the search order shown in [Table 89](#).

<table>
<thead>
<tr>
<th>Terminal operator</th>
<th>VTAM operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVS Message Service</td>
<td>USSTAB START option</td>
</tr>
<tr>
<td>LANGTAB USS table</td>
<td>ISTINCNO</td>
</tr>
<tr>
<td>LU's definition</td>
<td></td>
</tr>
</tbody>
</table>

For more information about the MODIFY TABLE command, see [z/OS Communications Server: SNA Operation](https://www.ibm.com/servers/resourcelink/systm5000/w3snavhome.nsf/Content/csm2226.html).
Table 89. VTAM messages search order for VTAM messages (continued)

<table>
<thead>
<tr>
<th>Terminal operator</th>
<th>VTAM operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTINCDT</td>
<td></td>
</tr>
</tbody>
</table>

1. You can use this service to define USS messages for terminal operators. See z/OS Communications Server: SNA Network Implementation Guide for information about how the MVS Message Service can be used to define USS messages for terminal operators.

For VTAM messages for a program operator

VTAM uses the search order shown in Table 90.

Table 90. VTAM messages search order for a program operator

<table>
<thead>
<tr>
<th>Program operator SSCP FM=USSNOP</th>
<th>Program operator SSCP FM=USSPOI</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPL definition statement</td>
<td>APPL definition statement</td>
</tr>
<tr>
<td>ISTINCN 0</td>
<td>ISTCFCM 0</td>
</tr>
<tr>
<td>ISTINCN 0</td>
<td>ISTINCN 0</td>
</tr>
</tbody>
</table>

If no USS message is found, VTAM will issue USS message 14 to indicate that the message was not found. (USS message 14 is defined the same as IST458I in ISTINCN0.)

**Note:** Do not modify or replace the IBM-supplied default tables.

If no VTAM message is found, VTAM issues message IST998E to indicate that a VTAM message is not defined.

A VTAM message might be rerouted from the program operator application to the system console. A rerouted message will appear in the format as defined in the USS table designated for the VTAM operator based on the search order.

If a VTAM message is not completely defined in the first USS table searched, the next USS table in the hierarchy will be used. Because the IBM-supplied USS table (ISTINCN0) contains complete messages, you should define supplemental operation-level USS tables to specify only those parts of a message that you want to change. If VTAM cannot locate all the required parts of a message, it issues message IST998E to indicate that the message does not exist.

For VTAM commands

VTAM uses the search order shown in Table 91.

Table 91. VTAM commands search order for VTAM commands

<table>
<thead>
<tr>
<th>Terminal operator</th>
<th>VTAM operator</th>
<th>Program operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>LANGTAB USS table</td>
<td>USSTAB START option</td>
<td>APPL definition</td>
</tr>
<tr>
<td>LU’s definition</td>
<td>ISTINCN0</td>
<td>ISTINCN0</td>
</tr>
<tr>
<td>ISTINCDT</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

If the USS command entered by the operator is not found, VTAM uses the command as entered.
For input translation tables

VTAM uses the search order shown in Table 92.

Table 92. Input translation tables search order

<table>
<thead>
<tr>
<th>Terminal operator</th>
<th>VTAM operator</th>
<th>Program operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>LU's definition</td>
<td>USSTAB START option</td>
<td>APPL definition</td>
</tr>
<tr>
<td>ISTINCDT</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

If no input translation table is found, VTAM performs no translation.

Installing a user-defined USS table

Each USS table must be assembled and link-edited as a non-executable module into a library associated with the SYS1.VTAMLIB DD statement.

Coding a user-defined USS table

To code a supplementary USS table, create a module using the following macroinstructions:
- USSTAB
- USSCMD
- USSPARM
- USSMSG
- USSEND

If a user-defined table is coded as part of another module, code an assembler EXTRN definition statement for the table name in that module so the table will be known externally and can be accessed by other modules.

If a character translation is to be specified, code the table using assembler DC statements as shown in the sample operation-level table "Operation-level USS table (ISTINCN0)" on page 1133.

Table 93 shows the USS macroinstructions, their associated operands, and where the operands are described.

Table 93. USS table macroinstructions and operands

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>USSTAB</td>
<td>DATEDLM</td>
<td>1081</td>
<td>/</td>
<td>MDY</td>
<td>FORMAT; V4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DATEFRM</td>
<td>1081</td>
<td></td>
<td></td>
<td>FORMAT; DATEDLM; V4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FORMAT</td>
<td>1084</td>
<td></td>
<td>OLD</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TABLE</td>
<td>1095</td>
<td></td>
<td></td>
<td>FORMAT; V4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TIMEDLM</td>
<td>1097</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USSCMD</td>
<td>CMD</td>
<td>1081</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FORMAT</td>
<td>1084</td>
<td></td>
<td>PL1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>REP</td>
<td>1092</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM</td>
<td>1091</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DEFAULT</td>
<td>1082</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>REP</td>
<td>1092</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRANSLATE</td>
<td>1097</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 93. USS table macroinstructions and operands (continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALUE</td>
<td>1097</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro instruction</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>USSMSG</td>
<td>BUFFER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSG</td>
<td>1087</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>1096</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DESC</td>
<td>1083</td>
<td></td>
<td>VTAM operator</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCSFLAG</td>
<td>1086</td>
<td></td>
<td>VTAM operator</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSGLEVEL</td>
<td>1087</td>
<td>START</td>
<td>VTAM operator</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OPT</td>
<td>1090</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ROUTCDE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUPP</td>
<td>1094</td>
<td>NEVER</td>
<td></td>
<td></td>
<td>terminal operator</td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

| R | Required operand. |

---

**Unformatted system services tables coding notes**

This information contains coding notes for the unformatted system services (USS) table.

**USSTAB macroinstruction**

The USSTAB macroinstruction indicates the beginning of a USS table.

**USSCMD macroinstruction**

The USSCMD macroinstruction is used to define VTAM operator or terminal operator commands.

**USSPARM macroinstruction**

The USSPARM macroinstruction defines an operand or positional parameter that can be specified on a command identified by the USSCMD macroinstruction. It also defines default values for the operand or positional parameter.

There can be multiple USSPARM macroinstructions associated with a USSCMD macroinstruction. For each operand (keyword and positional), code a USSPARM macroinstruction.

**USSMSG macroinstruction**

The USSMSG macroinstruction defines VTAM operator messages (ISTxxxx) or terminal operator messages (USSMSGxx).

To determine under what circumstances these messages are issued, see [z/OS Communications Server: SNA Messages](#).
**USSEND macroinstruction**

The USSEND macroinstruction delimits the end of the USS table.

**Unformatted system services (USS) tables full syntax**

The full syntax for the USS table follows. The syntax for each operand is repeated in "Operand descriptions" on page 1078.

**USSTAB**

```
name USSTAB
  DATEDLM = delimiter
  DATEFRM = MDY
  DYNAMIC
  TABLE = name
  TIMEDLM = delimiter
```

**USSCMD**

```
name USSCMD command_name
  FORMAT = PL1
  BAL
  REP = replace_command_name
```

**USSPARM**
USSMSG (terminal operator messages)

- name
  - REPL operand_name
    - TRANSLATE = YES
      - VALUE = value
    - TRANSLATE = NO
      - VALUE = value
  - REP operand_name
    - TRANSLATE = YES
      - VALUE = value
    - TRANSLATE = NO
      - VALUE = value

**USSMSG (VTAM operator messages)**
**USSEND**

Chapter 5. User-defined tables and data filter

1077
Operand descriptions

The following information describes the operands you can code for the USS table.

BUFFER

buffer_address

((buffer_address, LUNAME)

((buffer_address, SCAN))

macroinstructions: USSMSG
dependencies: TEXT, OPT, SUPP, terminal operator messages

buffer_address

Specifies the address (name) of an area of storage defined to contain the message text and a header indicating the length of the message text. The storage area must be formatted as shown in Figure 8.

message length (m) message text

2 bytes m bytes

Figure 8. USS message layout in storage

The message text defined in the storage area must follow the rules listed in “Syntax rules for messages” on page 1099.

The message text is sent to the terminal operator as it appears in the storage area. VTAM does not modify or translate the message text (even if FEATUR2=LOWERCSE is specified for the device). You are responsible for including any device-dependent control characters within the message.

LUNAME | SCAN

Specifies that the character strings listed in Table 94 on page 1079 will be replaced with the appropriate values in the position in the message where the character string occurred. The entire string specified by BUFFER will be searched, using the character @.

For example, if the following data was in the message text storage area

'@LUNAME IS NOW IN SESSION WITH VTAM, @DATE'

the message received at the terminal would look like the following information:

'TERM IS NOW IN SESSION WITH VTAM, 01/04/94'

Note: For terminals with large screen sizes, searching the storage area for a character string might be a performance consideration.
<table>
<thead>
<tr>
<th>Character string</th>
<th>In message text</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>@@@@@DATE</td>
<td>Current Date</td>
<td>In the format and with the delimiter specified in the DATEFRM and DATEDLM operands on the USSTAB macroinstruction.</td>
</tr>
<tr>
<td>@HOSTNET</td>
<td>NETID from Start Options</td>
<td>The network ID is left-aligned and trailing blanks are inserted, if the network ID is fewer than eight characters.</td>
</tr>
<tr>
<td>@...@ZONEID</td>
<td>Client IPv6 zone ID</td>
<td>If the IPv6 zone ID exists, it will be included.</td>
</tr>
<tr>
<td>@...@IPADD</td>
<td>Client IP Address</td>
<td>39 bytes, leading zeros suppressed, left justified, with trailing blanks if needed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the IP address type is IPv4 then it will be in dotted-decimal format. If the IP address type is IPv6 it will be in colon–hexadecimal format.</td>
</tr>
<tr>
<td>@...@IPHOSTNAME*</td>
<td>DNS Name of client</td>
<td>Name left justified with trailing blanks if fewer than 40 characters in length.</td>
</tr>
<tr>
<td>@@LUNAME</td>
<td>Terminal name</td>
<td>The name is left-aligned and trailing blanks are inserted, if the name is fewer than eight characters.</td>
</tr>
<tr>
<td>@@@NETID</td>
<td>Network ID</td>
<td>The network ID is left-aligned and trailing blanks are inserted, if the network ID is fewer than eight characters.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The network identifier for a nonnative-network-attached terminal is the real network name (the network identifier of the network where the terminal resides), not the name of the network it is attached to.</td>
</tr>
<tr>
<td>@@@@@@@@@@@@@@@@@NQN</td>
<td>Network-qualified name</td>
<td>The network-qualified name is left-aligned and trailing blanks are inserted if the network-qualified name is fewer than 17 characters, as networkID.name.</td>
</tr>
<tr>
<td>@@PRT</td>
<td>Client Port Address</td>
<td>Five bytes and leading zeros are not suppressed.</td>
</tr>
<tr>
<td>@@@@RUNAME</td>
<td>Request/response unit</td>
<td>The name of the request or response unit is left-aligned and trailing blanks are inserted, if the RU name is less than ten characters.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Applies only to USSMSG07 when coded using the BUFFER operand.</td>
</tr>
<tr>
<td>@@@SENSE</td>
<td>Sense code</td>
<td>Eight-character sense code.</td>
</tr>
<tr>
<td>@@SSCPNM</td>
<td>SSCPNAME from Start Options</td>
<td>The SSCP name is left-aligned and trailing blanks are inserted, if the SSCP name is fewer than 8 characters.</td>
</tr>
</tbody>
</table>
Table 94. Valid character strings for message definition (continued)

<table>
<thead>
<tr>
<th>Character string</th>
<th>In message text</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>@@@@TIME</td>
<td>Current Time</td>
<td>As HH_MM_SS, where an underscore (_) is the delimiter specified on the TIMEDLM operand of the USSTAB macroinstruction.</td>
</tr>
</tbody>
</table>

*IPHOSTNAME must be preceded by 40 @ symbols.

Note:
1. Although a message size of 65535 (X'FFFF') can be specified, ensure that the terminal can support the size of the message specified.
2. BUFFER and TEXT are mutually exclusive. VTAM produces an error message if both the BUFFER and TEXT operands are coded.
3. If start option IPINFO=NONE or IPINFO=DISPONLY is in effect, then the client IP address and D5 name of client will not be available for substitution even if they were given to VTAM by the LU.
4. If you code either the OPT or SUPP operands in addition to BUFFER, the OPT or SUPP operand will be ignored.
5. Various ABENDs can occur during USS processing if the USS table is improperly constructed. This is typically a user error because the USS table was incorrectly defined or modified.
   ABEND0C4 can occur in the following modules:
   • ISTINCF1
   • ISTUSCMG
   ABEND0A9 RCFF18 can occur in module ISTSSCCU. The most common user error occurs when the USS message is defined using the BUFFER=operand (instead of the TEXT=operand). When defining a USS message using BUFFER=, the message buffer must be constructed as follows:
   • USS Message length (2 bytes)
   • USS Message text (up to 65535 bytes)
   The USS message length field contains the length of the USS message text only (do not include the length of the 2-byte length field in the calculation of the message length).
   The following assembler statements show a popular method which can be used to define a USS message and the corresponding buffer when the BUFFER= operand is used:
   Somewhere after the USSTAB macro, code the USMSG macro for the message you wish to define (this example uses USMSG10):
   • USMSG MSG=10,BUFFER=(MSG10)
   Later, usually after the USSEND macro, define the USS message buffer as follows:
   MSG10 DC AL2(MSG10E-MSG10S) * Msg length (2 byte field)
   MSG10S EQU * * Start of USS message
   DC C'Place up to 65535 bytes of USS message text here'
   MSG10E EQU * * End of USS message

Note: The BUFFER= operand specifies the location of the 2-byte length field which is immediately followed by the message text.
**CMD**

```
CMD = command_name
```

**macroinstructions:**  
USSCMD

**dependencies:**  
none

Specifies the user-defined command name to which this USSCMD macroinstruction applies. `command_name` can consist of from 1–8 characters, the first of which must be alphabetical (A–Z) or a national character (@, #, or $).

**Note:** If two commands are defined with the same `command_name`, the assembly of the table will fail.

**DATEDLM**

```
DATEDLM = delimiter
```

**macroinstructions:**  
USSTAB

**dependencies:**  
FORMAT=DYNAMIC

Specifies the character to be used as a delimiter to separate the month, day, and year parts of the date where @@@@DATE is specified in the message. The slash (/) is used if DATEDLM is not specified.

DATEDLM is valid only when FORMAT=DYNAMIC.

**Note:** An ampersand (&) and single quotation mark (’) are not valid delimiters.

**DATEFRM**
Specifies the date format to be used where `@@DATE` is specified in the message text. Note that the delimiter used between the month, day, and year is specified on the DATEDLM operand.

DATEFRM is valid only when FORMAT=DYNAMIC.

**DATEFRM=DMY**

Specifies the day, followed by month, followed by year as `dd_mm_yy`, where an underscore (_) is the delimiter specified on the DATEDLM operand.

**DATEFRM=MDY**

Specifies the month, followed by day, followed by year as `mm_dd_yy`, where an underscore (_) is the delimiter specified on the DATEDLM operand.

**DATEFRM=YMD**

Specifies the year, followed by month, followed by day as `yy_mm_dd`, where an underscore (_) is the delimiter specified on DATEDLM operand.

The DATEFRM operand overrides the DATEFORM start option. If the DATEFRM operand is not specified, the DATEFORM start option takes effect.

**DEFAULT**
Specifies a default value to be used if the operand is omitted when the command is entered. If DEFAULT is not specified, the operand is treated as if it were not entered.

If the parameter in the PARM operand allows a network-qualified name to be specified, then the value of DEFAULT can be a network-qualified name.

The DEFAULT and VALUE operands cannot be coded on the same USSPARM macroinstruction. Instead, code two USSPARM macroinstructions with the same value specified for PARM. The macroinstruction specifying VALUE must precede the one containing the DEFAULT operand. If REP is to be specified, it must be on the macroinstruction containing the VALUE operand.

The default value (value) must follow the rules for coding assembler DC statements for character (C-type) constants. For example, if you want the following character string for value

'DUMP'

code

DEFAULT='DUMP'

DESC
$DESC=\{\texttt{descriptor\_code}\}$

**macroinstructions:** USSMSG

**dependencies:** VTAM operator messages only

Specifies the descriptor code to be used for this message. This operand is coded exactly like, and has the same meaning as, the DESC operand of the WTO or WTOR macroinstruction. For information about how to code this operand, see [z/OS MVS Programming: Authorized Assembler Services Reference SET-WTO](https://publib.boulder.ibm.com/infocenter/zos/v2r1/index.jsp?topic=%2Fcom.ibm.zos.v2r1.ref平稳.aaa.doc%2Fchapmvsas15.htm).

This operand applies only to messages that are sent to the VTAM operator. VTAM ignores this operand when sending a message to a program operator.

If the DESC operand is omitted or is specified with a null value (DESC=), VTAM uses the DESC value specified for this message ID in the next USS table searched.

**FORMAT (USSTAB)**

$FORMAT=\{\texttt{DYNAMIC}\}$

$FORMAT=\{\texttt{OLD}\}$

**macroinstructions:** USSTAB

**dependencies:** none

Specifies how the USS table will be formatted.

**FORMAT=DYNAMIC**

Specifies that the MODIFY TABLE command can be used to dynamically replace the USS table. Also, use FORMAT=DYNAMIC to permit terminal operators to specify the USS table using the LANGTAB operand on a USS command.

Specifying FORMAT=DYNAMIC can significantly reduce the CPU time required to search for VTAM operator messages. See [z/OS Communications Server: SNA Operation](https://publib.boulder.ibm.com/infocenter/zos/v2r1/index.jsp?topic=%2Fcom.ibm.zos.v2r1.doc平稳.sna00991.doc%2FCSN经营ss00991.html) for more information about dynamically replacing the USS table.

FORMAT=DYNAMIC is coded in both the IBM-supplied USS tables — ISTINCNO and ISTINCDT.

**FORMAT=OLD**

Specifies that the table is not dynamically replaceable.

**FORMAT (USSCMD)**
macroutines: USSCMD
dependencies: none

Specifies the syntax of the user-defined command to be entered by the terminal operator.

FORMAT=BAL
Specifies the user-defined command specified on this USSCMD macroinstruction is in Basic Assembler Language (BAL) syntax.

command
command
Identifies the command. It is followed by one or more blanks.

p
Used to specify one or more positional operand. Positional operands are entered in the format Pn, where n is the position number of the operand. Each operand (unless it is the last in the command) is followed by a comma. Positional operands must appear before any keyword operands.

keyword
Used to specify keyword operand associated with the command. Each operand (unless it is the last in a command) is followed by a comma.

value
The value assigned to a keyword operand.

FORMAT=PL1
Specifies the user-defined command specified on this USSCMD macroinstruction is in PL/1 programming syntax.
command

Identifies the command. It is followed by one or more blanks or by a left parenthesis (that is, positional operands).

p

Specifies one or more positional operands. Positional operands are entered in the format P\textit{n}, where \textit{n} is the position number. If positional operands are used, the parentheses must be coded.

\textit{keyword}

Used to enter each operand parameter. Each operand must be followed by one or more blanks or by a value enclosed in parentheses.

\textit{value}

The value assigned to a keyword operand.

See "Syntax rules for terminal operator commands" on page 1102 for a description of the syntax, the input character set, and BAL and PL/I value restrictions.

\textbf{Note:} If the USS table is to be searched twice (REP=VERB coded), this operand is ignored on the second pass of USS processing.

\textbf{MCSFLAG}
**MCSFLAG**

*macroinstructions:* USSMSG
*dependencies:* VTAM operator messages only

Specifies the multiple console support flags to be used for this message. This operand is coded exactly like, and has the same meaning as, the MCSFLAG parameter of the WTO or WTOR macroinstruction. For information about how to code this operand, see [z/OS MVS Programming: Authorized Assembler Services Reference](https://www.ibm.com/support/docview.wss?uid=swg27046809) Reference SET-WTO.

If the MCSFLAG operand is omitted or is specified with a null value (MCSFLAG=), VTAM uses the MCSFLAG value specified for this message ID in the next USS table searched.

If MCSFLAG is not coded in any of the USS tables that are searched, its value will default to zero.

**MSG**

*macroinstructions:* USSMSG
*dependencies:* none

Specifies which message or messages will be defined by this macroinstruction.

**Terminal operator messages:**
Enter decimal integers in the range 0–14. The numbers 0–14 correspond to the USS messages with message IDs of USSMSG00 through USSMSG14, respectively. See [z/OS Communications Server: SNA Messages](https://www.ibm.com/support/docview.wss?uid=swg27046809) for a description of USSMSG00 through USSMSG14.

**Note:** USSMSG00 and USSMSG10 are not defined in the IBM-supplied USS table. If you do not define these messages, no messages are sent for these conditions.

**VTAM operator messages:**
To define a VTAM message, specify a valid VTAM message ID. Enter the entire message ID in its correct format (for example, MSG=IST123I).

**MSGLVL**
Specifies the version of the message to issue on a message-by-message basis. MSGLVL is valid only for those messages listed in z/OS Communications Server: SNA Messages. The value specified on the MSGLVL operand overrides the MSGLEVEL start option value for this message.

The V4R1 and above messages often use network-qualified names for session-capable resources to enable you to know specifically which resource is affected by the message. You can choose to have VTAM issue the message that uses non-network qualified names to the VTAM operator console, but be aware that the messages that use network-qualified names might be more descriptive. Because the MSGLEVEL start option has a default value of BASE, you do not have to do anything if you want VTAM to issue the pre-V4R1 messages.

For example, message

IST073I command for ID=nodename FAILED - MORE POWERFUL REQUEST IN PROGRESS

has been replaced by the following message group

IST1129I command FAILED, nodename - DEACTIVATE PENDING
IST1045I NODE TYPE = nodetype
IST314I END

If you want to continue to get IST073I, code the following information:

[name] USSMSG MSG=IST1129I,MSGLVL=BASE

MSGLVL=BASE

Specifies that for the message number indicated on the MSG operand, issue the pre-V4R1 message that this message number replaces, regardless of the value coded on the MSGLEVEL start option.

For messages replaced by message groups, to get the base message, code MSGLVL=BASE for the first message in the group. See z/OS Communications Server: SNA Messages for a list of messages supported by the MSGLVL operand.

Note: If you specify MSGLVL=BASE and if there are duplicate names in your SNA interconnected networks, you might not know which resource the message applies to.

MSGLVL=START

Specifies that for the message number indicated on the MSG operand, issue the version of the message specified on the MSGLEVEL start option. START is the default for only those messages supported by the MSGLVL operand.
**MSGLVL=V4R1**
Specifies that for the message number indicated on the MSG operand, issue the V4R1 message or messages, regardless of the value specified on the MSGLEVEL start option. See [z/OS Communications Server: SNA Messages](https://www.ibm.com/support/docview.wss?uid=swg21417120) for a list of V4R1 messages supported by the MSGLVL operand.

**MSGLVL=V4R2**
Specifies that for the message number indicated on the MSG operand, issue the V4R2 message or messages, regardless of the MSGLVL value specified on the MSGLEVEL start option. See [z/OS Communications Server: SNA Messages](https://www.ibm.com/support/docview.wss?uid=swg21417120) for a list of messages supported by the MSGLVL operand.

**MSGLVL=V4R3**
Specifies that for the message number indicated on the MSG operand, issue the V4R2 message or messages, regardless of the MSGLVL value specified on the MSGLEVEL start option. See [z/OS Communications Server: SNA Messages](https://www.ibm.com/support/docview.wss?uid=swg21417120) for a list of messages supported by the MSGLVL operand.

**MSGLVL=V4R4**
Specifies that for the message number indicated on the MSG operand, issue the V4R2 message or messages, regardless of the MSGLVL value specified on the MSGLEVEL start option. See [z/OS Communications Server: SNA Messages](https://www.ibm.com/support/docview.wss?uid=swg21417120) for a list of messages supported by the MSGLVL operand.

**MSGLVL=CS390**
Specifies that for the message number indicated on the MSG operand, issue the z/OS Communications Server message or messages, regardless of the MSGLVL value specified on the MSGLEVEL start option. See [z/OS Communications Server: SNA Messages](https://www.ibm.com/support/docview.wss?uid=swg21417120) for a list of messages supported by the MSGLVL operand.

**name**
For USSCMD, USSPARAM, USSMSG, USSEND:
Specifies the name assigned to the macroinstruction.

For USSTAB:
Specifies the required CSECT name for the USS table.

OPT

For terminal operator

- OPT → BLKSUP
- OPT → NOBLKSUP

For VTAM operator

- OPT → NOBLKSUP
- OPT → BLKSUP

macroinstruction: USSTAB

macroinstruction: USSCMD; USSPARAM; USSMSG; USSEND

OPT = BLKSUP
OPT = NOBLKSUP

dependency: BUFFER

macroinstructions: USSMSG

/SM590000/SM590000
/SM590000/SM630000

z/OS V2R1.0 Communications Server: SNA Resource Definition Reference
For VTAM operator

<table>
<thead>
<tr>
<th>macroinstructions:</th>
<th>USSMSG</th>
</tr>
</thead>
<tbody>
<tr>
<td>dependency:</td>
<td>none</td>
</tr>
</tbody>
</table>

Indicates if extraneous blanks are to be suppressed from the defined message.

**OPT=BLKSUP**
Specifications that extraneous blanks are to be suppressed from the message. Any sequence of two or more blanks is to be converted into a single blank.

**OPT=NOBLKSUP**
Specifications that extraneous blanks are not to be suppressed from the message. Any sequence of two or more blanks is to be presented unchanged in the message.

**Note:**
1. For terminal operators VTAM ignores the OPT operand, if both OPT and BUFFER are specified.
2. It is recommended that OPT=NOBLKSUP be used for any message that is part of a tabular display.
3. If the OPT operand is omitted or is specified with a null value (OPT=), VTAM uses the OPT value specified for this message ID in the next USS table searched.

**PARM**
Specifies a user-defined keyword or positional parameter that can be coded in the user-defined command identified by the USSCMD definition.

**PARM=** **operand_name**

Specifies the keyword parameter in the user-entered command to which this USSPARM macroinstruction applies. **operand_name** must be 1–8 alphanumeric characters.

**PARM=P_number**

Specifies a positional parameter, where **number** is a decimal integer from 1 to the maximum number of positional parameters for the command. **P_number** indicates the positional parameter in the user-entered command to which this USSPARM macroinstruction applies.

See Example 2 in Appendix B, “Examples of USS command conversion,” on page 1369 for an example of positional and keyword parameters.

**REP (USSCMD)**

Specifies the valid command that is to replace the user-defined command specified by the CMD operand. See Example 2 in Appendix B, “Examples of USS command conversion,” on page 1369 for an example of a user-defined command that is to be replaced.

If the REP operand is not coded, the value specified in the CMD operand is used. In this case, CMD must specify a valid command. See Example 5 in Appendix B, “Examples of USS command conversion,” on page 1369 for an example.

**REP (USSPARM)**
**REP=** operand_name

Specifies the parameter is replaced with operand_name. The value for operand_name must be 1–8 alphanumeric characters. The value of the operand is assigned from the parameter specified by PARM. If PARM specifies a keyword parameter, its value is assigned to the operand specified by REP. If PARM specifies a positional parameter, its value is treated as if it were an operand value and it is assigned to the operand specified by REP.

If REP is not coded, it takes the value of PARM. (That is, the user-entered parameter is used as entered.)

Positional parameters such as P1 and P2 can also be used as operands. For multiple specifications of the same parameter, the last value specified is used (as shown in Example 3 in Appendix B, “Examples of USS command conversion,” on page 1369).

**REP=VERB**

Indicates that the command is replaced with the value of the VALUE operand. The USS table is then searched a second time, matching this new command to the value of a CMD operand on a USSCMD macroinstruction.

For example using the following USS macroinstructions, if the operator enters MODIFY DUMP, during a second pass of the USS table VTAM will look for USSCMD CMD=DUMP:

```
USSCMD CMD=MODIFY
USSPARM PARM=DUMP,REP=VERB,VALUE=DUMP

USSCMD CMD=DUMP
USSPARM PARM=DUMPDS
```

**ROUTCDE**
**ROUTECD**

```plaintext
ROUTECD = _route_code_
```

**macroinstructions:** `USSMSG`

**dependencies:** `VTAM operator messages only`

Specifies the routing code to be used for this message. This operand is coded exactly like, and has the same meaning as, the **ROUTECD** operand of the **WTO** or **WTOR** macroinstruction. For information on how to specify this operand, see 

[z/OS MVS Programming: Authorized Assembler Services Reference SET-WTO](https://www.ibm.com/support/knowledgecenter/SSEPGG_2.2.0/com.ibm.zos.v2r2.bse.ase.doc/ase_ref_index.html)

This operand applies only to messages that are sent to the VTAM operator. VTAM ignores this operand when sending a message to a program operator.

If the **ROUTECD** operand is omitted or is specified with a null value (`ROUTECD=```), VTAM uses the **ROUTECD** value specified for this message ID in the next USS table searched.

### SUPP

**Terminal operator**

```plaintext
SUPP = NEVER
```

**macroinstruction:** `USSMSG`

**dependency:** `BUFFER`

**VTAM operator**

```plaintext
SUPP = ALWAYS
```

**dependencies:** `INFO`, `NEVER`, `NORM`, `SER`, `WARN`
Specifies the message suppression class to be used for this message. The message suppression class for a message, along with the SUPP start option and the MODIFY SUPP command, determines whether a particular message will be suppressed by VTAM.

If the SUPP operand is omitted or is specified with a null value (SUPP=), VTAM uses the SUPP value specified for this message ID in the next USS table searched.

You cannot suppress individual VTAM messages in a multi-line WTO (MLWTO) message (that is, messages identified as issued as part of a message group). You can suppress message groups by suppressing the message identified as the first in the group. See [z/OS Communications Server: SNA Messages](https://www.ibm.com/support/knowledgecenter/en/SSLVMB_7.2.0/com.ibm.doc.msmgns2/msgmsgnmr.htm) to identify messages issued as part of a group.

**SUPP=ALWAYS**
- Specifies that this message is always to be suppressed, regardless of what level of message suppression is in effect.

**SUPP=INFO**
- Specifies that this message is to be in the informational message suppression class.

**SUPP=NEVER**
- Specifies that this message is never to be suppressed, regardless of what level of message suppression is in effect. That is, this message should always be written.

**SUPP=NORM**
- Specifies that this message is to be in the normal message suppression class.

**SUPP=SER**
- Specifies that this message is to be in the serious message suppression class.

**SUPP=WARNING**
- Specifies that this message is to be in the warning message suppression class.

For terminal operators:
- INFO, NORM, SER, or WARNING may also be coded for SUPP, but each of these is processed as though SUPP=NEVER were specified.
- If you code both the SUPP and BUFFER operands, VTAM ignores SUPP.

**TABLE**
**TABLE**

```
TABLE=name
```

**macroinstructions:** USSTAB  
**dependencies:** none

Specifies a translation table to be used by VTAM to translate character-coded
commands. If the table is not part of the module containing USSTAB, an EXTRN
statement must be coded for the specified name.

If no translation table is specified, VTAM uses the translation table associated with
the IBM-supplied USS table (or its user-written replacement). If the IBM-supplied
table (or its user-written replacement) does not have a translation table, VTAM
does no character translation.

**TEXT**

```
TEXT='message_text'
```

**macroinstructions:** USSMSG  
**dependencies:** BUFFER (terminal operator)

Specifies the text to replace the USS messages identified by the MSG operand. The
message text provided here must follow the rules listed in "Syntax rules for
messages" on page 1099.

**Note:** For a terminal operator message, VTAM produces an error message if you
code both the BUFFER and TEXT operands.

On TEXT, you can place any combination of the character strings described in
Table 94 on page 1079 VTAM will replace the strings with the values shown in the
table.

For example, the message definition
TEXT='@@LUNAME IS NOW IN SESSION WITH VTAM'

produces the following message:
TERMA IS NOW IN SESSION WITH VTAM

**Note:** The message appears as shown above, even if OPT=BLKSUP is coded.

If the LUNAME has 8 characters, it produces a message like the following
information:
TERMBBBB IS NOW IN SESSION WITH VTAM

An example using date and time follows:
TEXT='IT IS @@@@TIME ON @@@@DATE'

produces a message like the following information:
IT IS 10:14:51 ON 10/10/91
TIMEDLM

\[ \text{TIMEDLM} = \text{delimiter} \]

**macroinstructions:** USSTAB
**dependencies:** FORMAT=DYNAMIC

Specifies the character to be used as a delimiter to separate the hour, minutes, and seconds parts of the time. The colon (:) is used if TIMEDLM is not specified. The ampersand (&) and single quote (’) are not valid.

TIMEDLM is valid only when FORMAT=DYNAMIC, and is not supported by pre-V4 VTAMs.

TRANSLATE

\[ \text{TRANSLATE} = \text{YES} \]

**macroinstructions:** USSPAM
**dependencies:** none

Specifies if translation will occur for the specified USSPAM

**TRANSLATE=YES**

Specifies the USSPAM will be translated using the translation table associated with the USS table this USSPAM is coded in, unless the character string is within single quotation marks. Character strings within single quotation marks are not translated. TRANSLATE=YES is the default.

**TRANSLATE=NO**

Specifies that the USSPAM will not be translated. TRANSLATE=NO is intended to be coded on the USSPAM for DATA only when the data contains a mixed case password and the destination application supports mixed case passwords. For more information about mixed case passwords see [Z/OS Communications Server: SNA Network Implementation Guide](https://www.ibm.com/support/knowledgecenter/SSWGIW_6.3.0/com.ibm.zos.v6r1.wg39/zos_security.gs).

VALUE
Specifies the default value to be used if the operand specified by the PARM operand is entered without a value (see Example 3 in Appendix B, “Examples of USS command conversion,” on page 1369).

VALUE is in contrast with the DEFAULT operand, which specifies the default to be used if the operand itself is not entered.

Note:
1. If multiple VALUE operands are specified for the same operand, the first VALUE operand is used.
2. If the parameter in the PARM operand allows a network-qualified name to be specified, then the value of VALUE can be a network-qualified name.
3. The DEFAULT and VALUE operands cannot be coded on the same USSPARM macroinstruction. To use both operands, code two USSPARM macroinstructions with the same value specified for PARM. The macroinstruction specifying VALUE must precede the one containing the DEFAULT operand. If REP is to be specified, it must be on the macroinstruction containing the VALUE operand.

For example,

USSPARM P=T,REP=TYPE,VALUE=COND
USSPARM P=T,REP=TYPE,DEFAULT=COND

Customizing messages

VTAM provides an extensive set of messages that provide information to a terminal operator, to a program operator, or to a VTAM operator. You can redefine any of these messages by coding a USSMSG macroinstruction for each message to be redefined. These USSMSG macroinstructions are included in a user-defined USS table that is associated with the terminal, a program operator application, or the system console (for messages to the VTAM operator).

Using the USSMSG macroinstruction, you can change the message text and suppression class of any message.

Note:
1. For information on changing message characteristics, see z/OS Communications Server: SNA Messages.
2. For more information on how you can use the MVS Message Service to change the text of USS messages received by terminal users, see z/OS Communications Server: SNA Network Implementation Guide.
3. Because program operator applications depend on VTAM messages, changes to the operator messages could disrupt the functioning of a program operator applications.

A message might be only partially defined in a USS table. This allows options on messages to be received from a USS table that is next in the search order hierarchy. See “Order of USS table use” on page 1071 for more information on the search order for USS tables.
More than one message can be defined using a single USSMSG macroinstruction. If more than one message ID is specified, all of the corresponding messages are changed by the operands specified with this macroinstruction.

The USSMSGGxx messages have VTAM equivalents defined in the IBM-supplied USS tables. For example, ISTINCNO specifies

```
USSMSG MSG=(IST455I,11)
```

to define VTAM message IST455I and USSMSG11. These messages can be redefined using a single USSMSG macroinstruction.

Table 95 shows the VTAM message equivalent of USSMSG00 to USSMSG14.

<table>
<thead>
<tr>
<th>USS message number</th>
<th>VTAM operator message</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>IST457I</td>
</tr>
<tr>
<td>1</td>
<td>IST450I</td>
</tr>
<tr>
<td>2</td>
<td>IST451I</td>
</tr>
<tr>
<td>3</td>
<td>IST452I</td>
</tr>
<tr>
<td>4</td>
<td>IST453I</td>
</tr>
<tr>
<td>6</td>
<td>IST792I</td>
</tr>
<tr>
<td>8</td>
<td>IST454I</td>
</tr>
<tr>
<td>11</td>
<td>IST455I</td>
</tr>
<tr>
<td>12</td>
<td>IST456I</td>
</tr>
<tr>
<td>14</td>
<td>IST458I</td>
</tr>
</tbody>
</table>

**Note:** Allowing the default USSMSG10 to be sent to a non-SNA printer could cause printer forms to be out of alignment. You can avoid this by coding a separate USSTAB for these printers and coding USSMSG10 in this table as follows:

```
USSMSG MSG=10,BUFFER=BUF10
BUF10 DC X'0003F57A19'
```

**Syntax rules for messages**

When preparing USS messages to be sent to a terminal operator, to a VTAM operator, or to a program operator, the following rules apply:

- Single quotation marks in the message text must be specified as in the assembler DC statement for character (C-type) constants.

- The message text must conform to the rules for coding an assembler DC statement for character (C-type) constants. Alternatively, if noncharacter message text is required, the TEXT operand can be specified as a sublist, with each entry coded as the full operand of an assembler DC statement. If the sublist form is used, the assembler does boundary alignment processing. For example,

```
TEXT=(A(0),C'SAMPLE MESSAGE')
```

might result in some leading zeroes being supplied by the assembler to ensure that the “A(0)” part of the text is located on a fullword boundary. This could be avoided by specifying

```
TEXT=(AL4(0),C'SAMPLE MESSAGE')
```

- If the message text contains blanks (X’40’), OPT=BLKSUP should not be specified for this message, because VTAM might delete some of the X’40’ characters (unless that is what you want).
• You cannot use the 3270 control commands and orders when creating USS messages for SNA devices. For further information, see the component description manual for the appropriate control unit and attached devices.

• When VTAM issues a message, it can supply one or more variable data character strings to be inserted in the message text. The location of the variable data character strings in the message text is indicated by numbered or unnumbered percent signs (%).
  – Numbered percent signs are used when a message has more than one variable character string. The first variable data string is represented by %%(1), the second string by %%(2), and so on, for as many strings as are provided by VTAM.
  – An unnumbered percent sign is the same as specifying %%(1).

You can rearrange the order of the percent signs in a message to change the order of the variable data in a message.

For example, if the message definition for IST380I:

```
TEXT='ERROR FOR ID = %(1) — REQUEST: %(2), SENSE: %(3)'
```

results in the message

```
ERROR FOR ID = NODE1234 — REQUEST: CDINIT, SENSE: 087D0001
```

Then this message definition for IST380I:

```
TEXT='%(2) REQUEST RECEIVED SENSE %(3) FOR NODE %'
```

results in the message

```
CDINIT REQUEST RECEIVED SENSE 087D0001 FOR NODE NODE1234
```

**Note:** Noncharacter data should not contain percent signs (X'6C') because VTAM interprets a percent sign as a point at which to insert variable data into the message.

• Variable text can be used more than once in the same message or can be omitted in the redefined message.

```
TEXT='ERROR FOR ID = %(1) — REQUEST: %(2), SENSE: %(3)—%(2)'
```

would result in the message

```
ERROR FOR ID = NODE1234 — REQUEST: CDINIT, SENSE: 087D0001—CDINIT
```

• If the number of a percent sign is greater than the number of variable data character strings provided for that message, that percent sign is replaced by the last available variable data.

For example, if %%(7) were specified for the following message, the values for %%(3) will be used.

```
TEXT='ERROR FOR ID = %(1) — REQUEST: %(2), SENSE: %(3)' %%(7)
```

z/OS V2R1.0 Communications Server: SNA Messages shows the number and location of the variable data character strings in each VTAM message. It also provides a message explanation that includes a description of what type of data can appear in each character string.

• The maximum length of a message after replacement of any percent signs is 251 characters. If a message exceeds 251 characters, the USS table will not assemble. The maximum limit for messages in a message group is 60 characters for each message. Message will be truncated if maximum is exceeded.

• Do not change the message ID that appears in the text of a VTAM operator message. If you do, you could affect IBM's ability to service your product.

• The following characters can be used for USS messages:
  – 26 uppercase and lowercase letters: A–Z, a–z
  – 3 national characters: $ # @
- 10 numerical digits: 0–9
- All graphic characters greater than or equal to X'40' (with the exception of X'6C' which is the percent sign)

National characters (and any graphic or control characters not listed above) are sent to a terminal user only if present in user-specified message replacements specified through the BUFFER operand of the USSMSG macroinstruction.

**Note:** (For terminal operator messages only.) Because control characters are device-dependent, you should not include them in the data defined by the TEXT or BUFFER operands, unless you can select the appropriate character for the device to which the message will be sent. You may use the percent sign and blanks in data defined by the BUFFER operand. However, do not include the percent sign or blanks in the TEXT operand data because USS treats them differently.

### Defining VTAM operator commands

VTAM operator commands use the Basic Assembler Language (BAL) format. You have the option of specifying PL/I format for all USS commands in your USS table. This is not recommended, however, because using different formats for the VTAM commands might be confusing.

Unlike messages, a command must be completely defined in a single USS table.

#### Syntax rules for VTAM operator commands

The first two parts of each command, the command name and the first operand cannot be changed by the USS table.

<table>
<thead>
<tr>
<th>Command</th>
<th>1st Operand</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISPLAY</td>
<td>NET</td>
</tr>
<tr>
<td>MODIFY</td>
<td>procsname</td>
</tr>
<tr>
<td>VARY</td>
<td>NET</td>
</tr>
</tbody>
</table>

All other operands of a VTAM command can be changed in a user-defined table.

**Note:** The VTAM START and HALT commands cannot be changed through a USS table.

Include definitions for all operands, even if you are not redefining all of them. Any operand that does not appear in the user table will be passed unconverted to the command processor.

- Operands that describe the command function (such as TERM on the VARY NET,TERM command, ADJSSCPS on the DISPLAY NET,ADJSSCPS) cannot be changed. However, other operands can be redefined (such as LU1, LU2, and NOTIFY on the VARY NET,TERM command).
  
  VARY NET,TERM,LU1=lu1name,LU2=lu2name,NOTIFY=YES

- Keyword operands can be converted into another operand with the same value. For example

  MODIFY MSG=YES

  can be converted into

  MODIFY MSGMOD=YES

- It is also possible to specify a default value to be used if the operand is omitted or specified without a value.
USSPARM PARM=MSGMOD, VALUE=YES
USSPARM PARM=MSGMOD, DEFAULT=YES
MODIFY procname, MSGMOD=,
MODIFY procname

• An operand value specified by the operator cannot be replaced using the USS table.
  For example,
  MODIFY procname, MSGMOD=NO

  cannot be converted into
  MODIFY procname, MSGMOD=YES

• Commands cannot be redefined to perform more than one function. For example, the following commands
  DISPLAY NET, CDRMS
  DISPLAY NET, CLSTRS

  cannot be converted to one command
  DISPLAY NET, CDRMS, CLSTRS

**Defining terminal operator commands**

VTAM recognizes the following terminal operator commands that originate from a terminal operator. These terminal operator commands are received from a logical unit that sends unformatted character-coded logon and logoff requests:

- “LOGON command” on page 1103
- “LOGOFF command” on page 1108
- “IBMTEST command” on page 1113

VTAM converts each of these unformatted commands into a field-formatted SNA request.

**Syntax rules for terminal operator commands**

The following syntax rules apply to commands you define for terminal operator commands. If these rules are not followed, use an interpret table to convert the character-coded command into a formatted SNA request.

• The following characters are allowed:
  - All graphics characters (greater than or equal to X'40')
  - BS (backspace: X'16')
  - HT (horizontal tab: X'05')
  - IRS (interchange record separator: X'1E')
  - NL (new line: X'15')

  **Note:** New line characters are deleted from the character string before translation if you specify SSCPFM=FSS or SSCP=USSCS on the LU definition statement.

• Command names and operand values that identify a name can contain 1–8 characters in the following format:
  1st character  alphabetical (upper or lowercase) or the national characters @, #, or $
  2nd to 8th characters alphanumeric or the national characters @, #, or $
- Command names cannot contain commas except between paired single quotation marks or parentheses.
- Commands names in the BAL format cannot contain blanks or horizontal tab characters except between the command and first operand or between paired single quotation marks.
- For PL/1 format, operand values cannot contain semicolons except between paired single quotation marks Positional operand values cannot contain commas except between paired single quotation marks.
- Operand values cannot contain blanks, horizontal tabs, or unpaired parentheses except between paired single quotation marks.

**Note:** There cannot be an odd number of single quotation marks.

- For BAL commands, the terminal operator cannot use the following items for operand values:
  - Commas except between paired single quotation marks or parentheses
  - A positional parameter cannot contain equal signs except between paired parentheses or single quotation marks
- Operand values can contain any of the following characters:
  - All graphics characters (greater than or equal to X'40')
  - BS (backspace: X'16')
  - HT (horizontal tab: X'05')
  - Data entered from a magnetic card reader

**Note:** Magnetic card data from a BSC or type 1 PU 3270 device is supported only if the card data is used as the last data in a value within quotation marks for the last operand of a USS command. Press the CLEAR key before you enter a character-coded command that contains magnetic card reader data.

### LOGON command

The LOGON command allows the terminal operator to request a session with an application program.

**Table 96. LOGON command and operands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOGON</td>
<td>APPLID(appl prog name or uservar name)</td>
<td>1104</td>
<td>R</td>
<td></td>
<td>R</td>
</tr>
<tr>
<td></td>
<td>DATA(user_data)</td>
<td>1105</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LANG(language_code or language name)</td>
<td>1105</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LANGTAB(uss_table_name,language_code)</td>
<td>1105</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LOGMODE(logon_mode_entry)</td>
<td>1106</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MODEL(model_name)</td>
<td>1107</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PRINTER1(primary_printer_name)</td>
<td>1107</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PRINTER2(alternate_printer_name)</td>
<td>1107</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

R Required operand.
Full syntax
The full syntax for the LOGON command follows. The syntax for each operand is repeated in "Operand descriptions."

```
LOGON

APPLID (appl_prog_name)

(uservar_name)

DATA (user_data)

LANG (language_code)

(language_name)

LANGTAB (USS_table, language_code)

LOGMODE (logon_mode_entry)

MODEL (model_name)

PRINTER1 (primary_printer_name)

PRINTER2 (alternate_printer_name)
```

Operand descriptions
The following information describes the operands you can code for the LOGON command.

APPLID
APPLID

Specifies the name of the application program or character string specified by the SEQUENCE entry in the interpret table (USERVAR) with which a session is to be established. For more information on network regulations regarding name specification, see “APPL definition statement” on page 36.

DATA

Specifies user data to be made available to the application program’s logon exit routine. To embed blanks, code user_data as a single-quoted string. The application receiving the user data will not receive the quotes.

LANG

Specifies the language in which the terminal operator will receive USS messages from the MVS Message Service. You are responsible for defining these USS messages to the MVS Message Service and for having the MVS Message Service active in the host. Otherwise, the terminal operators will not receive USS messages from the MVS Message Service in the language of their choice. The terminal user continues to receive USS messages in the specified language, until the terminal user enters another valid USS command using the LANG operand.

For the LOGON command only, the specified language is also passed to the application program during logon processing.

If you enter this operand without a language_code or language_name specified the MVS Message Service is not used for USS messages.

Note: If the MVS Message Service is not currently active, the LANG operand is ignored.

LANG(language_code)

Specifies a 3-character code that indicates a valid language. For a list of valid language code values, see z/OS Communications Server: SNA Programming.

(LANG=language_name)

Specifies the character string associated with a language. You can use the MVS Message Service to define your own values for the language_name.

For information on the MVS Message Service, see z/OS MVS Planning: Operations and IBM Tivoli NetView for z/OS Installation: Getting Started.
Specifies the session-level USS table that VTAM uses for USS messages and commands. It also specifies the language that is passed to the application program. This operand provides the same function as the LANG operand, except it does not use the MVS Message Service.

**USS_table_name**
Specifies the USS table that VTAM will search first for USS messages and commands. VTAM uses this USS table until the terminal operator issues another USS command using the LANGTAB operand.

**language_code**
Specifies a 3-character code used to specify the language that is passed to the application program during logon processing. If a valid LANG operand is also specified and the MVS Message Service is active, this parameter is overridden by the language specified with the LANG operand. For a list of valid values and for details about how the application obtains this information, see [z/OS Communications Server: SNA Programming](http://www.ibm.com/香/OSV2R1.0CommunicationsServer:SNAResourceDefinitionReference).

**Note:**
1. If you enter this operand without a USS table name, it removes the LANGTAB USS table (specified earlier) from the search hierarchy.
2. If you enter this operand without a language code, the application receives the language from the LANG operand of the MODEENT macroinstruction (specified by LOGMODE).

**LOGMODE**
LOGMODE (logon_mode_entry)

Specifies the logon mode entry that is used to select a set of session parameters for the session to be established.

MODEL

MODEL (model_name)

Specifies the 1–8 character model name to be passed to the application program during logon processing.

If this operand is entered with a name, the data it contains is used. Table specifications on the terminal’s definition statement are ignored for that type of data during the current session.

If this operand is entered without a name value, an override with null data results, and no model name data is sent to the application program for the requested session.

PRINTER1

PRINTER1 (primary_printer_name)

Specifies the primary printer name to be passed to the application program during logon processing.

If this operand is entered with a name, the data it contains is used. Table specifications on the terminal’s definition statement are ignored for that type of data during the current session.

If this operand is entered without a name value, an override with null data results, and no primary printer name data is sent to the application program for the requested session.

PRINTER2
Specifies the alternate printer name to be passed to the application program during logon processing.

If this operand is entered with a name, the data it contains is used. Table specifications on the terminal's definition statement are ignored for that type of data during the current session.

If this operand is entered without a name value, an override with null data results, and no alternate printer name data is sent to the application program for the requested session.

**LOGOFF command**

The LOGOFF command allows the terminal operator to request that a session with an application program be ended.

### Table 97. LOGOFF command and operands

<table>
<thead>
<tr>
<th>Command</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOGOFF</td>
<td>APPLID(appl_prog_name)</td>
<td>1109</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HOLD(YES or NO)</td>
<td>1110</td>
<td></td>
<td>YES</td>
<td>token ring</td>
</tr>
<tr>
<td></td>
<td>LANG(language_code or language_name)</td>
<td>1110</td>
<td></td>
<td></td>
<td>MVS Message Service</td>
</tr>
<tr>
<td></td>
<td>LANGTAB(uss_table_name,language_code)</td>
<td>1111</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TYPE(COND or FORCE or UNCOND)</td>
<td>1112</td>
<td></td>
<td>UNCOND</td>
<td></td>
</tr>
</tbody>
</table>

**Full syntax**

The full syntax for the LOGOFF command follows. The syntax for each operand is repeated in "Operand descriptions" on page 1109.
Operand descriptions
The following information describes the operands you can code for the LOGOFF command.

APPLID
APPLID:

Specifies the name of the application program with which a session is to be terminated. If omitted, the application program with which a current session exists is assumed.

**HOLD**

- **HOLD (YES)**
- **HOLD (NO)**

**dependencies:** do not code for token ring

Specifies the action the logical unit expects VTAM to take when physically terminating the SSCP-PU session after the logical unit has been disconnected. A HOLD value of YES corresponds to a NOT LAST indicator on a Terminate Self request. The effect of HOLD depends on the setting of the DISCNT operand specified on the PU statement. The relationship between HOLD and DISCNT is described in the z/OS Communications Server: SNA Network Implementation Guide.

**Note:** HOLD(NO) should not be specified for token ring connections as this might cause errors in attached gateways which require operator intervention.

**LANG**
LANG {language_code} {language_name}

dependencies: MVS Message Service active

Specifies the language in which the terminal operator will receive USS messages from the MVS Message Service. You are responsible for defining these USS messages to the MVS Message Service and for having the MVS Message Service active in the host. Otherwise, the terminal operators will not receive USS messages from the MVS Message Service in the language of their choice. The terminal user continues to receive USS messages in the specified language, until the terminal user enters another valid USS command using the LANG operand.

If you enter this operand without a language code or name, USS messages are no longer retrieved from the MVS Message Service.

**Note:** If the MVS Message Service is not currently active, the LANG operand is ignored.

**language_name**
- Specifies the character string associated with a language. You can use the MVS Message Service to define your own values for the `language_name`.

**language_code**
- Specifies a 3-character code that indicates a valid language. For a list of valid language code values, see `z/OS Communications Server: SNA Programming`.

For information on the MVS Message Service, see `IBM Tivoli NetView for z/OS Installation: Getting Started` and the `z/OS MVS Programming: Assembler Services Guide`.

**LANGTAB**
LANGTAB(USS_table_name,language_code)

Specifies the session-level USS table that VTAM uses for USS messages and commands. It also specifies the language that is passed to the application program. This operand provides the same function as the LANG operand, except it does not use the MVS Message Service.

**USS_table_name**
Specifies the USS table that VTAM will search first for USS messages and commands. VTAM uses this USS table until the terminal operator issues another USS command using the LANGTAB operand.

**language_code**
Specifies a 3-character code used to specify the language that is passed to the application program during logon processing. If a valid LANG operand is also specified and the MVS Message Service is active, this parameter is overridden by the language specified with the LANG operand. For a list of valid values and for details on how the application obtains this information, see [z/OS Communications Server: SNA Programming](https://www.ibm.com/support/knowledgecenter/en/SSLDCP_2.2.1/com.ibm.zos.v2r11.cqsicps.doc/COMCHT20100.html).

**Note:**
1. If you enter this operand without a USS table name, it removes the LANGTAB USS table (specified earlier) from the search hierarchy.
2. If you enter this operand without a language code, the application receives the language from the LANG operand of the MODEENT macroinstruction (specified by LOGMODE).

**TYPE**
Specifies the manner in which an active session between an application acting as a primary logical unit (PLU) and another logical unit is to be ended.

**TYPE (COND)**

Specifies that the LOSTERM exit routine of the PLU receives a code indicating conditional session ending.

**TYPE (FORCE)**

Specifies that the PLU’s NSEXIT exit routine receives a code indicating forced ending and the session is brought down immediately by the SSCP. If there is no NSEXIT exit routine, VTAM sends the code to the LOSTERM exit routine, which must issue a CLSDST macroinstruction.

**TYPE (UNCOND)**

Specifies that the PLU’s LOSTERM exit is driven with a code indicating unconditional session ending. SENDs and RECEIVEs stop immediately, and the PLU is to clean up storage used for the session by issuing a CLSDST macroinstruction.

UNCOND is used for normal situations when the session is assumed to be viable and an UNBIND can be sent to end the session. FORCE is used in abnormal situations, for example, when either one of the session ends cannot communicate with the SSCP. When use of the UNCOND operand has no effect, the FORCE operand should be specified.

**IBMTEST command**

The IBMTEST command allows the terminal operator to test the physical path between the terminal and VTAM. IBMTEST causes VTAM to return test data to the terminal a specified number of times. VTAM returns the test data with the prefix “IBMECHO”.

**Table 98. IBMTEST command and operands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Operands</th>
<th>Page</th>
<th>Req</th>
<th>Default</th>
<th>Restrictions, dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBMTEST</td>
<td>number</td>
<td>1115</td>
<td>R</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LANG(language_code or language_name)</td>
<td>1114</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LANGTAB(uss_table_name,language_code)</td>
<td>1114</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>test_data</td>
<td>1116</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:

R Required operand.

**Full syntax**

The full syntax for the IBMTEST command follows. The syntax for each operand is repeated “Operand descriptions” on page 1114.
Operand descriptions

The following information describes the operands you can code for the IBMTEST command.

LANG

Specifies the language in which the terminal operator will receive USS messages from the MVS Message Service. You are responsible for defining these USS messages to the MVS Message Service and for having the MVS Message Service active in the host. Otherwise, the terminal operators will not receive USS messages from the MVS Message Service in the language of their choice. The terminal user continues to receive USS messages in the specified language, until the terminal operator enters another valid USS command using the LANG operand.

If you enter this operand without a language_code or language_name, USS messages are no longer retrieved from the MVS Message Service.

Note: If the MVS Message Service is not currently active, the LANG operand is ignored.

language_name
Specifies the character string associated with a language. You can use the MVS Message Service to define your own values for the language_name.

language_code
Specifies a 3-character code that indicates a valid language. For a list of valid language code values, see z/OS Communications Server: SNA Programming.

For information on the MVS Message Service, see z/OS MVS Planning: Operations and IBM Tivoli NetView for z/OS Installation: Getting Started.

LANGTAB
LANGTAB ("USS_table_name", "language_code")

Specifies the session-level USS table that VTAM uses for USS messages and commands. It also specifies the language that is passed to the application program. This operand provides the same function as the LANG operand, except it does not use the MVS Message Service.

**uss_table_name**
- Specifies the USS table that VTAM will search first for USS messages and commands. VTAM uses this USS table until the terminal operator issues another USS command using the LANGTAB operand.

**language_code**
- Specifies a 3-character code used to specify the language that is passed to the application program during logon processing. If a valid LANG operand is also specified and the MVS Message Service is active, this parameter is overridden by the language specified with the LANG operand. For a list of valid values and for details about how the application obtains this information, see [z/OS Communications Server: SNA Programming](https://www.ibm.com/support/docview.ws/docid/137269).

**Note:**
1. If you enter this operand without a USS table name, it removes the LANGTAB USS table (specified earlier) from the search hierarchy.
2. If you enter this operand without a language code, the application receives the language from the LANG operand of the MODEENT macroinstruction (specified by LOGMODE).

**number**
Specifies the number of times the test data should be returned to the terminal.

**test_data**

Specifies the test data to be returned. Specify a character string of up to 247 characters, or the maximum message length of the terminal, whichever is smaller. If you do not specify a character string, VTAM sends out and returns a string consisting of the letters A–Z followed by the decimal numbers 0–9. (“ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789”) You can include blanks as part of this message if the string is surrounded by single quotes.
Appendix A. IBM-supplied tables

This appendix includes:

- The default logon mode table (ISTINCLM)
- The session-level USS table (ISTINCIDT)
- The operation-level USS table (ISTINCNO)
- The APPN Class of Service (CoS) definitions (COSAPPN)
- The APPN transmission group (TG) profile definitions (IBMTGPS)
- The message flooding table (ISTMSFLD)
- The CMIP services directory definition file (ACYDDF)

If you use this appendix to code replacements for or changes to any of these tables, be sure to refer to "Format of macroinstructions" on page 10 for correct coding formats, especially when continuing lines of code.

Default logon mode table (ISTINCLM)

```
EJECT ISTINCLM MODETAB EJECT
IBM3767 MODEENT LOGMODE=INTERACT,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B*1',SECPROT=X'A0',COMPROT=X'3040',APPNCOS=#INTER
EJECT TITLE 'TWXDEVPT'
**********************************************************************
** TWX DEVICE WITH THE DCODE SET TO KEYBOARD AND PRINTER. THIS IS THE DEFAULT SETTING. **
**********************************************************************
TWXDEVPT MODEENT LOGMODE=TWXDECPT,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B*1',SECPROT=X'A0',COMPROT=X'3040',DCODE=X'80', APPNCOS=#CONNECT
EJECT TITLE 'TWXDEVDP'
**********************************************************************
** TWX DEVICE WITH THE DCODE SET TO KEYBOARD AND DISPLAY. **
**********************************************************************
TWXDEVDP MODEENT LOGMODE=TWXDEVPD,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B*1',SECPROT=X'A0',COMPROT=X'3040',DCODE=X'80', APPNCOS=#CONNECT
EJECT IBM3770 MODEENT LOGMODE=BATCH,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'A3',SECPROT=X'A3',COMPROT=X'7080',APPNCOS=#BATCH
EJECT IBM3270 MODEENT LOGMODE=S3270,FMPROF=X'02',TSPROF=X'02',PRIPROT=X'71',SECPROT=X'40',COMPROT=X'2000', APPNCOS=#CONNECT
EJECT IBM3600 MODEENT LOGMODE=IBM3600,FMPROF=X'04',TSPROF=X'04',PRIPROT=X'F1',SECPROT=X'F1',COMPROT=X'7000',APPNCOS=#CONNECT
EJECT IBM3650I MODEENT LOGMODE=INTRACT,FMPROF=X'04',TSPROF=X'04',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'6000',APPNCOS=#INTER
EJECT
```
IBM3650U MODEENT LOGMODE=INTRUSER,FMPROF=X'04',TSPROF=X'04',PRIPROT=X'3'* 1',SECPROT=X'30',COMPROT=X'6000',APPNCOS=#INTER
EJECT
IBM3650 MODEENT LOGMODE=IBM3650,FMPROF=X'04',TSPROF=X'04',PRIPROT=X'B'* 0',SECPROT=X'80',COMPROT=X'4000',APPNCOS=#CONNECT
EJECT
IBM3650P MODEENT LOGMODE=PIPELINE,FMPROF=X'04',TSPROF=X'04',PRIPROT=X'3'* 0',SECPROT=X'10',COMPROT=X'0000',APPNCOS=#CONNECT
EJECT
IBM3660 MODEENT LOGMODE=SMAPPL,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'A0'*, SECPROT=X'00',COMPROT=X'0000',APPNCOS=#CONNECT
EJECT
IBM3660A MODEENT LOGMODE=SMSNA100,FMPROF=X'00',TSPROF=X'00',PRIPROT=X'0'* 0',SECPROT=X'00',COMPROT=X'0000',APPNCOS=#CONNECT
TITLE 'D6327801'
**************************************************************************
**
* 3276 SNA WITH 3278 MODEL 1 SCREEN *
* PRIMARY SCREEN 12 X 40 (480) *
* ALTERNATE SCREEN 12 X 80 (960) *
*
**************************************************************************
D6327801 MODEENT LOGMODE=D6327801,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B'* 1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'8BF8',PSERVIC* =X'02000000000C280C507F00',APPNCOS=#CONNECT
TITLE 'D6327802'
**************************************************************************
**
* 3276 SNA WITH 3278 MODEL 2 SCREEN *
* PRIMARY SCREEN 24 X 80 (1920) *
* NO ALTERNATE SCREEN DEFINED *
*
**************************************************************************
D6327802 MODEENT LOGMODE=D6327802,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B'* 1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'8BF8',PSERVIC* =X'02000000000185020507F00',APPNCOS=#CONNECT
TITLE 'D6327803'
**************************************************************************
**
* 3276 SNA WITH MODEL 3 SCREEN *
* PRIMARY SCREEN 24 X 80 (1920) *
* ALTERNATE SCREEN 32 X 80 (2560) *
*
**************************************************************************
D6327803 MODEENT LOGMODE=D6327803,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B'* 1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'8BF8',PSERVIC* =X'02000000000185020507F00',APPNCOS=#CONNECT
TITLE 'D6327804'
**************************************************************************
**
* 3276 SNA WITH MODEL 4 SCREEN *
* PRIMARY SCREEN 24 X 80 (1920) *
* ALTERNATE SCREEN 43 X 80 (3440) *
*
**************************************************************************
D6327804 MODEENT LOGMODE=D6327804,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B'* 1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'8BF8',PSERVIC* =X'02000000000185020507F00',APPNCOS=#CONNECT
TITLE 'D6327805'
**************************************************************************
**
* 3276 SNA WITH MODEL 5 SCREEN *
* PRIMARY SCREEN 24 X 80 (1920) *
* ALTERNATE SCREEN 27 X 132 (3564) *
*
**************************************************************************
D6327805 MODEENT LOGMODE=D6327805,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B'*
1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'8787',PSERVIC* =X'030000000000185018507F00',APPNCOS=#CONNECT
TITLE 'D6328904'
******************************************************************************
**
* 3276 SNA WITH 3289 MODEL 4 PRINTER *
*
******************************************************************************
D6328904 MODEENT LOGMODE=D6328904,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'8787',PSERVIC* =X'03000000000018502B507F00',APPNCOS=#CONNECT
TITLE 'D6328902'
******************************************************************************
**
* 3276 SNA WITH 3289 MODEL 2 PRINTER *
*
******************************************************************************
D6328902 MODEENT LOGMODE=D6328902,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'8787',PSERVIC* =X'030000000000185000007E00',APPNCOS=#CONNECT
TITLE 'D4A32781'
******************************************************************************
**
* 3274 MODEL 1A WITH MODEL 1 SCREEN (LOCAL SNA) *
* PRIMARY SCREEN 12 X 40 (480)
* ALTERNATE SCREEN 12 X 80 (960)
*
******************************************************************************
D4A32781 MODEENT LOGMODE=D4A32781,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC* =X'0200000000000C280C507F00',APPNCOS=#CONNECT
TITLE 'D4A32782'
******************************************************************************
**
* 3274 MODEL 1A WITH MODEL 2 SCREEN (LOCAL SNA) *
* PRIMARY SCREEN 24 X 80 (1920)
* NO ALTERNATE SCREEN DEFINED *
*
******************************************************************************
D4A32782 MODEENT LOGMODE=D4A32782,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC* =X'020000000000185000007E00',APPNCOS=#CONNECT
TITLE 'LSK32782'
******************************************************************************
**
* 3274 MODEL 1A WITH MODEL 2 SCREEN (LOCAL SNA) *
* PRIMARY SCREEN 24 X 80 (1920)
* NO ALTERNATE SCREEN DEFINED *
* KATAKANA *
*
******************************************************************************
LSK32782 MODEENT LOGMODE=LSK32782,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC* =X'020000000000185000007E00',LANG=X'11',APPNCOS=#CONNECT
TITLE 'D4A32783'
******************************************************************************
**
* 3274 MODEL 1A WITH MODEL 3 SCREEN (LOCAL SNA) *
* PRIMARY SCREEN 24 X 80 (1920)
* ALTERNATE SCREEN 32 X 80 (2560)
*
******************************************************************************
D4A32783 MODEENT LOGMODE=D4A32783,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC* =X'020000000000185020507F00',APPNCOS=#CONNECT
TITLE 'D4A32784'
******************************************************************************
**
* 3274 MODEL 1A WITH MODEL 3 SCREEN (LOCAL SNA) *
* PRIMARY SCREEN 24 X 80 (1920)
* ALTERNATE SCREEN 32 X 80 (2560)
*
******************************************************************************
D4C32782 MODEENT LOGMODE=D4C32782,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87F8',PSERVIC+X'0200000000018500007E00',APPNCOS=#CONNECT
TITLE 'RSK32782'

**********************************************************************

** 3274 MODEL 1C WITH MODEL 2 SCREEN(REMOTE SNA)  
** PRIMARY SCREEN 24 X 80 (1920)  
** NO ALTERNATE SCREEN DEFINED  
** KATAKANA  
**
**********************************************************************

RSK32782 MODEENT LOGMODE=RSK32782,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87F8',PSERVIC+X'02000000000185000007E00',LANG=X'11',  
APPNCOS=#CONNECT
TITLE 'D4C32783'

**********************************************************************

** 3274 MODEL 1C WITH MODEL 3 SCREEN(REMOTE SNA)  
** PRIMARY SCREEN 24 X 80 (1920)  
** ALTERNATE SCREEN 32 X 80 (2560)  
**
**********************************************************************

D4C32783 MODEENT LOGMODE=D4C32783,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87F8',PSERVIC+X'0200000000018500007F00',APPNCOS=#CONNECT
TITLE 'D4C32784'

**********************************************************************

** 3274 MODEL 1C WITH MODEL 4 SCREEN(REMOTE SNA)  
** PRIMARY SCREEN 24 X 80 (1920)  
** ALTERNATE SCREEN 43 X 80 (3440)  
**
**********************************************************************

D4C32784 MODEENT LOGMODE=D4C32784,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87F8',PSERVIC+X'0200000000018502050F00',APPNCOS=#CONNECT
TITLE 'D4C32785'

**********************************************************************

** 3274 MODEL 1C WITH MODEL 5 SCREEN(REMOTE SNA)  
** PRIMARY SCREEN 24 X 80 (1920)  
** ALTERNATE SCREEN 27 X 132 (3564)  
**
**********************************************************************

D4C32785 MODEENT LOGMODE=D4C32785,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87F8',PSERVIC+X'02000000000185010B8F00',APPNCOS=#CONNECT
TITLE 'D4C32XX3'

**********************************************************************

** 3274 MODEL 1C (REMOTE SNA)  
** PRIMARY SCREEN 24 X 80 (1920)  
** ALTERNATE SCREEN TO BE DETERMINED BY APPLICATION  
**
**********************************************************************

D4C32XX3 MODEENT LOGMODE=D4C32XX3,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87F8',PSERVIC+X'02000000000185010B47F00',APPNCOS=#CONNECT
TITLE 'D4C32771'

**********************************************************************

** 3274 MODEL 1C WITH 3277 MODEL 1 SCREEN  
**
**********************************************************************

Appendix A. IBM-supplied tables
1121
**3274 MODEL 1C WITH 3277 MODEL 2 SCREEN**

**3274 MODEL 1B/1D WITH MODEL 1 SCREEN (LOCAL NON-SNA)**

**3274 MODEL 1B/1D WITH MODEL 2 SCREEN (LOCAL NON-SNA)**

**3274 MODEL 1B/1D WITH MODEL 2 SCREEN (LOCAL NON-SNA)**

**3274 MODEL 1B/1D WITH MODEL 2 SCREEN (LOCAL NON-SNA)**

**3274 MODEL 1B/1D WITH MODEL 2 SCREEN (LOCAL NON-SNA)**
Appendix A. IBM-supplied tables
* PRINTER WITH 4K BUFFER *

**********************************************************************
DSC4K MODEENT LOGMODE=DSC4K,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',
SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'8787',
PSERVIC=X'03000000000185028507F00',
APPNCOS=#CONNECT
TITLE 'DSC4K'
**********************************************************************

* PRINTER WITH 2K BUFFER *

**********************************************************************
DSC2K MODEENT LOGMODE=DSC2K,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',
SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'8787',
PSERVIC=X'03000000000185018507F00',
APPNCOS=#CONNECT
TITLE 'BAT13790'
**********************************************************************

** 3790 BATCH **

**********************************************************************
BAT13790 MODEENT LOGMODE=BAT13790,FMPROF=X'03',TSPROF=X'03',
PRIPROT=X'00',SECPROT=X'00',COMPROT=X'0080',
RUSIZES=X'0000',APPNCOS=#BATCH
TITLE 'EMU3790'
**********************************************************************

** 3790 IN DATA STREAM COMPATIBILITY MODE **

**********************************************************************
EMU3790 MODEENT LOGMODE=EMU3790,FMPROF=X'03',TSPROF=X'03',
PRIPROT=X'B1',SECPROT=X'B0',COMPROT=X'7080',
RUSIZES=X'85C7',PSERVIC=X'020000000000000000000200',
APPNCOS=#CONNECT
TITLE 'RJE3790A'
**********************************************************************

** 3790 RJE **

**********************************************************************
RJE3790A MODEENT LOGMODE=RJE3790A,FMPROF=X'03',TSPROF=X'03',
PRIPROT=X'A3',SECPROT=X'A1',COMPROT=X'7080',
RUSIZES=X'8585',PSERVIC=X'01106000F100800000010040',
APPNCOS=#CONNECT
TITLE 'RJE3790B'
**********************************************************************

** 3790 JRE **

**********************************************************************
RJE3790B MODEENT LOGMODE=RJE3790B,FMPROF=X'03',TSPROF=X'03',
PRIPROT=X'A3',SECPROT=X'A1',COMPROT=X'7080',
RUSIZES=X'8585',PSERVIC=X'01102000F100800000010040',
APPNCOS=#CONNECT
TITLE 'BAT23790'
**********************************************************************

** 3790 BATCH **

**********************************************************************
BAT23790 MODEENT LOGMODE=BAT23790,FMPROF=X'03',TSPROF=X'04',
PRIPROT=X'B1',SECPROT=X'B0',COMPROT=X'7080',
RUSIZES=X'8585',PSERVIC=X'013100000000000000000000',
APPNCOS=#BATCH
TITLE 'BLK3790'
**********************************************************************
**3790 BULK PRINT**

**3790 WITH SNA CHARACTER SET**

**3790 IN DPCX EMULATION MODE**

**3790 LOGMODE TABLE FOR BSC, LOCAL, SDLC 3275, 3277, 3278, 3279**

**3790 IN DPCX EMULATION MODE**

**3790 LOGMODE TABLE FOR THE 3290 TERMINAL OR EXTENDED DATA STREAM TERMINAL OFF 3274-1A**

**LOGMODE TABLE ENTRY FOR NON-SNA 3270 DEVICES WITH**
* EXTENDED DATA STREAMS (3278 OR 3279).
* SCREEN SIZE IS 24 X 80.

**********************************************************************
NSX32702 MODEENT LOGMODE=NSX32702,FMPROF=X'02',TSPROF=X'02',
PRIPROT=X'71',SECPROT=X'40',COMPROT=X'2000',
RUSIZES=X'0000',
PSERVIC=X'0080000000018500007EO0',
APPNCOS=#CONNECT
TITLE 'NSX32703'
**********************************************************************

* LOGMODE TABLE ENTRY FOR NON-SNA 3270 DEVICES WITH
* EXTENDED DATA STREAMS (3278 OR 3279).
* PRIMARY SCREEN 24 X 80
* ALTERNATE SCREEN 32 X 80

**********************************************************************
NSX32703 MODEENT LOGMODE=NSX32703,FMPROF=X'02',TSPROF=X'02',
PRIPROT=X'71',SECPROT=X'40',COMPROT=X'2000',
RUSIZES=X'0000',
PSERVIC=X'0080000000018502057EO0',
APPNCOS=#CONNECT
TITLE 'NSX32704'
**********************************************************************

* LOGMODE TABLE ENTRY FOR NON-SNA 3270 DEVICES WITH
* PRIMARY SCREEN 24 X 80
* ALTERNATE SCREEN 43 X 80

**********************************************************************
NSX32704 MODEENT LOGMODE=NSX32704,FMPROF=X'02',TSPROF=X'02',
PRIPROT=X'71',SECPROT=X'40',COMPROT=X'2000',
RUSIZES=X'0000',
PSERVIC=X'0080000000018502B57EO0',
APPNCOS=#CONNECT
TITLE 'NSX32705'
**********************************************************************

* LOGMODE TABLE ENTRY FOR NON-SNA 3270 DEVICES WITH
* PRIMARY SCREEN 24 X 80
* ALTERNATE SCREEN 27 X 132

**********************************************************************
NSX32705 MODEENT LOGMODE=NSX32705,FMPROF=X'02',TSPROF=X'02',
PRIPROT=X'71',SECPROT=X'40',COMPROT=X'2000',
RUSIZES=X'0000',
PSERVIC=X'0080000000018501B847EO0',
APPNCOS=#CONNECT
TITLE 'SNX32702'
**********************************************************************

* LOGMODE TABLE ENTRY FOR Remote SNA 3270 DEVICES
* WITH EXTENDED DATA STREAMS (3278 OR 3279).
* SCREEN SIZE IS 24 X 80.

**********************************************************************
SNX32702 MODEENT LOGMODE=SNX32702,FMPROF=X'03',TSPROF=X'03',
PRIPROT=X'81',SECPROT=X'90',COMPROT=X'3080',
RUSIZES=X'87FB',
PSERVIC=X'02B0000000018500007EO0',
APPNCOS=#CONNECT
TITLE 'SNX32703'
**********************************************************************

* LOGMODE TABLE ENTRY FOR Remote SNA 3270 DEVICES
* WITH EXTENDED DATA STREAMS (MOD3).
* PRIMARY SCREEN 24 X 80 (1920)
* ALTERNATE SCREEN 32 X 80 (2560)

**********************************************************************
SNX32703 MODEENT LOGMODE=SNX32703,FMPROF=X'03',TSPROF=X'03',
PRIPROT=X'81',SECPROT=X'90',COMPROT=X'3080',
RUSIZES=X'87FB',
PSERVIC=X'02B0000000018502057EO0',
APPNCOS=#CONNECT

1126  z/OS V2R1.0 Communications Server: SNA Resource Definition Reference
TITLE 'SNX32704'

*****************************************************************************
* LOGMODE TABLE ENTRY FOR REMOTE SNA 3270 DEVICES *
* WITH EXTENDED DATA STREAMS (MOD4). *
* PRIMARY SCREEN 24 X 80 (1920) *
* ALTERNATE SCREEN 43 X 80 (3440) *
*****************************************************************************

SNX32704 MODEENT LOGMODE=SNX32704,FMPROF=X'03',TSPROF=X'03',
PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',
RUSIZES=X'87F8',
PSERVIC=X'02800000000018502B507F00',
APPNCOS=#CONNECT

TITLE 'SNX32705'

*****************************************************************************
* LOGMODE TABLE ENTRY FOR REMOTE SNA 3270 DEVICES *
* WITH EXTENDED DATA STREAMS (MOD5). *
* PRIMARY SCREEN 27 X 132 (3564) *
*****************************************************************************

SNX32705 MODEENT LOGMODE=SNX32705,FMPROF=X'03',TSPROF=X'03',
PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',
RUSIZES=X'87F8',
PSERVIC=X'02800000000018501B847F00',
APPNCOS=#CONNECT

TITLE 'LSX32702'

*****************************************************************************
* 3274 MODEL 1A WITH MODEL 2 SCREEN (LOCAL SNA) *
* WITH EXTENDED DATA STREAMS (MOD2) *
* PRIMARY SCREEN 24 X 80 (1920) *
* NO ALTERNATE SCREEN DEFINED *
*****************************************************************************

LSX32702 MODEENT LOGMODE=LSX32702,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',
SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC=
X'028000000000185000007E00',APPNCOS=#CONNECT

TITLE 'LSX32703'

*****************************************************************************
* 3274 MODEL 1A WITH MODEL 3 SCREEN (LOCAL SNA) *
* WITH EXTENDED DATA STREAMS (MOD3) *
* PRIMARY SCREEN 24 X 80 (1920) *
* ALTERNATE SCREEN 32 X 80 (2560) *
*****************************************************************************

LSX32703 MODEENT LOGMODE=LSX32703,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',
SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC=
X'028000000000185020507F00',APPNCOS=#CONNECT

TITLE 'LSX32704'

*****************************************************************************
* 3274 MODEL 1A WITH MODEL 4 SCREEN (LOCAL SNA) *
* WITH EXTENDED DATA STREAMS (MOD4) *
* PRIMARY SCREEN 24 X 80 (1920) *
* ALTERNATE SCREEN 43 X 80 (3440) *
*****************************************************************************

LSX32704 MODEENT LOGMODE=LSX32704,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',
SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC=
X'02800000000018502B507F00',APPNCOS=#CONNECT

TITLE 'LSX32705'

*****************************************************************************
* 3274 MODEL 1A WITH MODEL 5 SCREEN (LOCAL SNA) *
* WITH EXTENDED DATA STREAMS (MOD5) *
*****************************************************************************
LSX32705 MODEENT LOGMODE=LSX32705,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC=X'0280000000018501B847F00',APPNCOS=#CONNECT
TITLE 'NED32702'

LOGMODE TABLE ENTRY FOR NON-SNA 3270 DEVICES WITH
EXTENDED DATA STREAMS (3278 OR 3279).
SCREEN SIZE IS 24 X 80.
LANGUAGE IS ENGLISH.
QUERY FOR DOUBLE BYTE CAPABILITY.

NED32702 MODEENT LOGMODE=NED32702,FMPROF=X'02',TSPROF=X'02',
PRIPROT=X'71',SECPROT=X'40',COMPROT=X'2000',
RUSIZES=X'0000',
PSERVIC=X'0080000000185000007E00',LANG=X'81',
APPNCOS=#CONNECT
TITLE 'NKD32702'

NAME: NON-SNA KATAKANA, DOUBLE BYTE CAPABLE, 3270-2
LOGMODE TABLE ENTRY FOR NON-SNA 3270 DEVICES WITH
EXTENDED DATA STREAMS (3278 OR 3279).
SCREEN SIZE IS 24 X 80.
LANGUAGE IS KATAKANA.
QUERY FOR DOUBLE BYTE CAPABILITY.

NKD32702 MODEENT LOGMODE=NKD32702,FMPROF=X'02',TSPROF=X'02',
PRIPROT=X'71',SECPROT=X'40',COMPROT=X'2000',
RUSIZES=X'0000',
PSERVIC=X'0080000000185000007E00',LANG=X'91',
APPNCOS=#CONNECT
TITLE 'LKD32702'

LOGMODE TABLE ENTRY FOR LOCAL SNA
3274 MODEL 1A WITH MODEL 2 SCREEN (LOCAL SNA)
PRIMARY SCREEN 24 X 80 (1920)
NO ALTERNATE SCREEN DEFINED
EXTENDED DATA STREAMS
ENGLISH LANGUAGE
QUERY FOR DOUBLE BYTE CAPABILITY

LED32702 MODEENT LOGMODE=LED32702,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC=X'0280000000018500007E00',LANG=X'81',
APPNCOS=#CONNECT
TITLE 'LKD32702'

LOGMODE TABLE ENTRY FOR LOCAL SNA
3274 MODEL 1A WITH MODEL 2 SCREEN (LOCAL SNA)
PRIMARY SCREEN 24 X 80 (1920)
NO ALTERNATE SCREEN DEFINED
EXTENDED DATA STREAMS
KATAKANA LANGUAGE
QUERY FOR DOUBLE BYTE CAPABILITY

LKD32702 MODEENT LOGMODE=LKD32702,FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87C7',PSERVIC=X'0280000000018500007E00',LANG=X'91',
APPNCOS=#CONNECT
TITLE 'LND32702'
* LOGMODE TABLE ENTRY FOR LOCAL SNA *  
* 3274 MODEL 1A WITH MODEL 2 SCREEN (LOCAL SNA) *  
* PRIMARY SCREEN 24 X 80 (1920) *  
* NO ALTERNATE SCREEN DEFINED *  
* EXTENDED DATA STREAMS *  
* NO LANGUAGE SPECIFIC FILTERING *  
* QUERY FOR DOUBLE BYTE CAPABILITY *  
*  
**********************************************************************

LND32702 MODEENT LOGMODE=LND32702,FMPROF=X'03',TSPROF=X'03',PRI
PRIPROT=X'B1',SECPROT=X'90',COMPR0T=X'3080',RUSIZES=X'87C7',PSERVIC
SENC=X'02800000001B5000007E00',LANG=X'FF', *
APPNCOS=#CONNECT
TITLE 'SED32702'

**********************************************************************

* LOGMODE TABLE ENTRY FOR REMOTE SNA 3270 DEVICES *  
* WITH EXTENDED DATA STREAMS (3278 OR 3279). *  
* SCREEN SIZE IS 24 X 80. *  
* LANGUAGE IS ENGLISH. *  
* QUERY FOR DOUBLE BYTE CAPABILITY *  
*  
**********************************************************************

SED32702 MODEENT LOGMODE=SED32702,FMPROF=X'03',TSPROF=X'03', *
PRIPROT=X'B1',SECPROT=X'90',COMPR0T=X'3080', *
RUSIZES=X'87F8', *
PSERVIC=X'028000000001B5000007E00',LANG=X'81', *
APPNCOS=#CONNECT
TITLE 'SKD32702'

**********************************************************************

* LOGMODE TABLE ENTRY FOR REMOTE SNA 3270 DEVICES *  
* WITH EXTENDED DATA STREAMS (3278 OR 3279). *  
* SCREEN SIZE IS 24 X 80. *  
* LANGUAGE IS KATAKANA. *  
* QUERY FOR DOUBLE BYTE CAPABILITY *  
*  
**********************************************************************

SKD32702 MODEENT LOGMODE=SKD32702,FMPROF=X'03',TSPROF=X'03', *
PRIPROT=X'B1',SECPROT=X'90',COMPR0T=X'3080', *
RUSIZES=X'87F8', *
PSERVIC=X'028000000001B5000007E00',LANG=X'91', *
APPNCOS=#CONNECT
TITLE 'SND32702'

**********************************************************************

* LOGMODE TABLE ENTRY FOR REMOTE SNA 3270 DEVICES *  
* WITH EXTENDED DATA STREAMS (3278 OR 3279). *  
* SCREEN SIZE IS 24 X 80. *  
* NO LANGUAGE SPECIFIC FILTERING *  
* QUERY FOR DOUBLE BYTE CAPABILITY *  
*  
**********************************************************************

SND32702 MODEENT LOGMODE=SND32702,FMPROF=X'03',TSPROF=X'03', *
PRIPROT=X'B1',SECPROT=X'90',COMPR0T=X'3080', *
RUSIZES=X'87F8', *
PSERVIC=X'028000000001B5000007E00',LANG=X'FF', *
APPNCOS=#CONNECT
TITLE 'SNASVCMG'

**********************************************************************

* LOGMODE TABLE ENTRY FOR RESOURCES CAPABLE OF ACTING *  
* AS LU 6.2 DEVICES *  
*  
**********************************************************************

SNASVCMG MODEENT LOGMODE=SNASVCMG,TYPE=0,FMPROF=X'13',TSPROF=X'07', *
PRIPROT=X'B0',SECPROT=X'B0',COMPR0T=X'D081', *
RUSIZES=X'9797',ENC=8'0000', *
PSERVIC=X'060200000000000002300', *
APPNCOS=SNASVCMG *
SPECIFY SYNC LEVEL = CONFIRM
TITLE '#BATCH'

**********************************************************************

* LOGMODE TABLE FOR BATCH SESSIONS ON RESOURCES CAPABLE *  
* OF ACTING AS LU 6.2 DEVICES *  
*  
**********************************************************************
#BATCH MODEENT LOGMODE=#BATCH,FMPROF=X'13',TSPROF=X'07',
  ENCR=B'0000',SSNDPAC=3,RUSIZES=X'F7F7',
  SRCVPAC=3,PSNDPAC=3,APPNCOS=#BATCH
TITLE '#INTER'

#BATCHC MODEENT LOGMODE=#BATCHC,FMPROF=X'13',TSPROF=X'07',
  ENCR=B'0000',SSNDPAC=3,RUSIZES=X'F7F7',
  SRCVPAC=3,PSNDPAC=3,APPNCOS=#BATCH
TITLE '#INTERC'

#INTER MODEENT LOGMODE=#INTER,FMPROF=X'13',TSPROF=X'07',
  ENCR=B'0000',SSNDPAC=7,RUSIZES=X'F7F7',
  SRCVPAC=7,PSNDPAC=7,APPNCOS=#INTER
TITLE '#BATCHSC'

#INTERC MODEENT LOGMODE=#INTERC,FMPROF=X'13',TSPROF=X'07',
  ENCR=B'0000',SSNDPAC=7,RUSIZES=X'F7F7',
  SRCVPAC=7,PSNDPAC=7,APPNCOS=#INTER
TITLE '#INTERSC'

CPSVCMG MODEENT LOGMODE=CPSVCMG,
RUSIZES=X'9797',ENCR=B'0000',

* LOGMODE TABLE FOR BATCH COMPRESSION SESSIONS ON RESOURCES
  * CAPABLE OF ACTING AS LU 6.2 DEVICES
  *
* LOGMODE TABLE FOR INTERACTIVE COMPRESSION SESSIONS ON
  RESOURCES CAPABLE OF ACTING AS LU 6.2 DEVICES
  *
* LOGMODE TABLE FOR BATCH SESSIONS REQUIRING SECURE
  TRANSPORT ON RESOURCES CAPABLE OF ACTING AS LU 6.2
  DEVICES
  *
* LOGMODE TABLE FOR INTERACTIVE SESSIONS REQUIRING
  SECURE TRANSPORT ON RESOURCES CAPABLE OF ACTING AS
  LU 6.2 DEVICES
  *
* LOGMODE TABLE FOR CP-CP SESSIONS ON RESOURCES CAPABLE
  OF ACTING AS LU 6.2 DEVICES
  *

**LOGMODE TABLE ENTRY THAT SUPPLIES A DEFAULT COS AND USES LU 6.2 DEVICE CHARACTERISTICS**

ISTCOSDF MODEENT LOGMODE=ISTCOSDF,FMPROF=X'13', TSPROF=X'07',PRIPROT=X'B0',SECPROT=X'B0', COMPROT=X'D0B1',PSERVIC=X'060200000000000000000300', RUSIZES=X'9999',ENC'0000',TYPE=0, APPNCOS=#CONNECT

**LOGMODE TABLE ENTRY FOR AS/400 PC SUPPORT**

QPCSUPP MODEENT LOGMODE=QPCSUPP,FMPROF=X'13',TSPROF=X'07', ENCR=B'0000',SSNDPAC=7, SRCVPAC=7,PSNDPAC=7, APPNCOS=#CONNECT

**LOGMODE TABLE ENTRY FOR DRDA SUPPORT**

IBMRDB MODEENT LOGMODE=IBMRDB, DRDA DEFAULT MODE TYPE=0, NEGOTIABLE BIND PSNDPAC=X'00', PRIMARY SEND PACING COUNT SSNDPAC=X'02', SECONDARY SEND PACING COUNT SRCVPAC=X'00', SECONDARY RECEIVE PACING COUNT RUSIZES=X'9999', RUSIZES IN-4K OUT-4K FMPROF=X'13', LU6.2 FM PROFILE TSPROF=X'07', LU6.2 TS PROFILE PRIPROT=X'B1', LU6.2 PRIMARY PROTOCOLS SECPROT=X'B0', LU6.2 SECONDARY PROTOCOLS COMPROT=X'50A5', LU6.2 COMMON PROTOCOLS PSERVIC=X'06020000000000000000010200', LU6.2 LU TYPE APPNCOS=#CONNECT

**LOGMODE TABLE FOR SESSIONS BETWEEN A DLS (DEPENDENT LU) AND A DLR (DEPENDENT LU REQUESTOR).**

CPSVRMGR MODEENT LOGMODE=CPSVRMGR, ENCR=B'0000', RUSIZES=X'9979', SSNDPAC=7, SRCVPAC=7, PSNDPAC=7, APPNCOS=SNASVCMG, COS=ISTVTCOS

**LOGMODE ENTRY FOR 3174 DLUR**

DLUR MODEENT LOGMODE=DLUR,FMPROF=X'03',TSPROF=X'03', PRIPROT=X'B1',SECPROT=X'90',COMPROT=X'30B0', SSNDPAC=X'04',PSNDPAC=X'04', RUSIZES=X'8787', PSERVIC=X'0280000000000185000007E00', APPNCOS=#CONNECT
TITLE 'SNADAUTH'
***********************************************************************
**
* LOGMODE ENTRY FOR DCE
*
***********************************************************************

**
SNADAUTH MODEENT LOGMODE=SNADAUTH, PSNDPAC=X'03',
SSNDPAC=X'03', SRCVPAC=X'03', ENCR=B'0000'
MODEEND, END OF DEFAULT TABLE ENTRIES
END, END OF ISTINCLM

EJECT
ISTINCDT USSTAB TABLE=STDTRANS,FORMAT=DYNAMIC
SPACE 4
LOGON USSCMD CMD=LOGON,FORMAT=PL1
USSPARM PARM=APPLID
USSPARM PARM=LOGMODE
USSPARM PARM=DATA
EJECT
LOGOFF USSCMD CMD=LOGOFF,FORMAT=PL1
USSPARM PARM=APPLID
USSPARM PARM=TYPE,DEFAULT=UNCOND
USSPARM PARM=HOLD,DEFAULT=YES
EJECT
IBMTEST USSCMD CMD=IBMTEST,FORMAT=BAL
USSPARM PARM=P1,DEFAULT=10
USSPARM PARM=P2,DEFAULT=ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
EJECT

USSMSG01 USSMSG MSG=1,TEXT='INVALID % COMMAND SYNTAX'
USSMSG02 USSMSG MSG=2,TEXT='% COMMAND UNRECOGNIZED'
USSMSG03 USSMSG MSG=3,TEXT='% PARAMETER EXTRANEOUS'
USSMSG04 USSMSG MSG=4,TEXT='% PARAMETER VALUE %= NOT VALID'
USSMSG05 USSMSG MSG=5,TEXT='UNSUPPORTED FUNCTION'
USSMSG06 USSMSG MSG=6,TEXT='SEQUENCE ERROR'
USSMSG07 USSMSG MSG=7,TEXT='%(1) UNABLE TO ESTABLISH SESSION - %(2) FAILED WITH SENSE %(3)'
USSMSG08 USSMSG MSG=8,TEXT='INSUFFICIENT STORAGE'
USSMSG09 USSMSG MSG=9,TEXT='MAGNETIC CARD DATA ERROR'
USSMSG10 USSMSG MSG=10,TEXT='% SESSIONS ENDED'
USSMSG11 USSMSG MSG=11,TEXT='REQUIRED PARAMETER OMITTED'
USSMSG12 USSMSG MSG=12,TEXT='IBMECHO % '
USSMSG13 USSMSG MSG=13,TEXT='% SESSIONS ENDED'
USSMSG14 USSMSG MSG=14,TEXT='USS MESSAGE % NOT DEFINED'
EJECT

STDTRANS DC X'0001203044006070B90A0B0C0D0E0F'
DC X'101112131315161718191A1B1C1D1E1F'
DC X'202122232425262728292A2B2C2D2E2F'
DC X'303132333435363738393A3B3C3D3E3F'
DC X'404142434445464748494A4B4C4D4E4F'
DC X'505152535455565758595A5B5C5D5E5F'
DC X'606162636465666768696A6B6C6D6E6F'
DC X'707172737475767778797A7B7C7D7E7F'
DC X'808182838485868788898A8B8C8D8E8F'
DC X'909192939495969798999A9B9C9D9E9F'
DC X'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'
DC X'B0B1B2B3B4B5B6B7B8B9BABAABBABCBD'BEF'
DC X'C0C1C2C3C4C5C6C7C8C9CAACCACDCEECCF'
DC X'D0D1D2D3D4D5D6D7D8D9DADBCDDEDEF'
DC X'E0E1E2E3E4E5E6E7E8E9EAEABEBECEDFEEF'
DC X'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFFEEF'
EJECT

STDTRANS DC X'0001203044006070B90A0B0C0D0E0F'
DC X'101112131315161718191A1B1C1D1E1F'
DC X'202122232425262728292A2B2C2D2E2F'
DC X'303132333435363738393A3B3C3D3E3F'
DC X'404142434445464748494A4B4C4D4E4F'
DC X'505152535455565758595A5B5C5D5E5F'
DC X'606162636465666768696A6B6C6D6E6F'
DC X'707172737475767778797A7B7C7D7E7F'
DC X'808182838485868788898A8B8C8D8E8F'
DC X'909192939495969798999A9B9C9D9E9F'
DC X'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'
DC X'B0B1B2B3B4B5B6B7B8B9BABAABBABCBD'BEF'
DC X'C0C1C2C3C4C5C6C7C8C9CAACCACDCEECCF'
DC X'D0D1D2D3D4D5D6D7D8D9DADBCDDEDEF'
DC X'E0E1E2E3E4E5E6E7E8E9EAEABEBECEDFEEF'
DC X'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFFEEF'
EJECT

END USSEND
END, END OF ASSEMBLY

Session-level USS table (ISTINCDT)
### Operation-level USS table (ISTINCNO)

<table>
<thead>
<tr>
<th>USSCMD</th>
<th>CMD=DISPLAY, FORMAT=BAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>USSPARM</td>
<td>PARM=A, REP=SCOPE, VALUE=ACT</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=ACT, REP=SCOPE, VALUE=ACT</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=ACTIVITY, REP=SCOPE, VALUE=ACTIVITY</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=ACTONLY, REP=SCOPE, VALUE=ACTONLY</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=ACTSES, REP=SCOPE, VALUE=ACTSES</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=ADJCLUST, REP=VERB, VALUE=ADJCLUST</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=ADJCP, REP=VERB, VALUE=ADJCP</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=ADJSCS, REP=VERB, VALUE=ADJSCS</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=APIING, REP=VERB, VALUE=APIING</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=APIINGQTP, REP=VERB, VALUE=APIINGQTP</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=APIINGTP, REP=VERB, VALUE=APIINGTP</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=APPLS, REP=VERB, VALUE=APPLS</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=APNTOSA, REP=VERB, VALUE=APNTOSA</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=AUTOLOG, REP=VERB, VALUE=AUTOLOG</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=BFRUSE, REP=VERB, VALUE=BFRUSE</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=BCOSMAP, REP=VERB, VALUE=BCOSMAP</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=CNOS, REP=VERB, VALUE=CNOS</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=CONCT, REP=SCOPE, VALUE=CONCT</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=COS, REP=VERB, VALUE=COS</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=CPS, REP=VERB, VALUE=CPS</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=CPCP, REP=VERB, VALUE=CPCP</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=CPSB, REP=VERB, VALUE=CPSB</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=CPSDUMP, REP=VERB, VALUE=CPSDUMP</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=DET, REP=LIST, VALUE=DETAIL</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=DETAIL, REP=LIST, VALUE=DETAIL</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=DEVSTATS, REP=DEVSTATS, VALUE=YES</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=DIOLOG, REP=VERB, VALUE=DIOLOG</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=E, REP=SCOPE, VALUE=E</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=EE, REP=VERB, VALUE=EE</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=EEDIAG, REP=VERB, VALUE=EEDIAG</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=EVERY, REP=SCOPE, VALUE=ALL</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=EXIT, REP=VERB, VALUE=EXIT</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=GRAFFIN, REP=VERB, VALUE=GRAFFIN</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=GROUPS, REP=VERB, VALUE=GROUPS</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=HPD, REP=VERB, VALUE=HPD</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=INACT, REP=SCOPE, VALUE=INACT</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=ID</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=INACT, REP=SCOPE, VALUE=INACT</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=INACTONLY, REP=SCOPE, VALUE=INACTONLY</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=INOPCODE, REP=VERB, VALUE=INOPCODE</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=IPDPDUMP, REP=VERB, VALUE=IPDPDUMP</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=LINE, REP=VERB, VALUE=_LINES</td>
</tr>
<tr>
<td>USSPARM</td>
<td>PARM=LINE, REP=VERB, VALUE=_LINES</td>
</tr>
</tbody>
</table>
USSPARM PARM=LIST
USSPARM PARM=LMTBL,REP=VERB,VALUE=LMTBL
USSPARM PARM=LUGROUPS,REP=VERB,VALUE=LUGROUPS
USSPARM PARM=MAJNODES,REP=VERB,VALUE=MAJNODES
USSPARM PARM=MODELS,REP=VERB,VALUE=MODELS
USSPARM PARM=NETID
USSPARM PARM=NETSRVR,REP=VERB,VALUE=NETSRVR
USSPARM PARM=NONE,REP=SCOPE,VALUE=ONLY
USSPARM PARM=NCPSTOR,REP=VERB,VALUE=NCPSTOR
USSPARM PARM=N,REP=SCOPE,VALUE=ONLY
USSPARM PARM=PATHS,REP=VERB,VALUE=PATHS
USSPARM PARM=PATHTAB,REP=VERB,VALUE=PATHTAB
USSPARM PARM=P,REP=VERB,VALUE=PATHS
USSPARM PARM=PENDING,REP=SCOPE,VALUE=PENDING
USSPARM PARM=RECENT,REP=SCOPE,VALUE=RECENT
USSPARM PARM=RELSD,REP=SCOPE,VALUE=RELSD
USSPARM PARM=RESET,REP=SCOPE,VALUE=RESET
USSPARM PARM=ROUTE,REP=VERB,VALUE=ROUTE
USSPARM PARM=RSTLIST,REP=VERB,VALUE=RSTLIST
USSPARM PARM=RTPS,REP=VERB,VALUE=RTPS
USSPARM PARM=S,REP=VERB,VALUE=S
USSPARM PARM=SAMAP,REP=VERB,VALUE=SAMAP
USSPARM PARM=SATOAPPN,REP=VERB,VALUE=SATOAPPN
USSPARM PARM=SCOPE
USSPARM PARM=SESSION,REP=VERB,VALUE=SESSIONS
USSPARM PARM=SESSIONS,REP=VERB,VALUE=SESSIONS
USSPARM PARM=SNSFILTER,REP=VERB,VALUE=SNSFILTER
USSPARM PARM=SCHINFO,REP=VERB,VALUE=SCHINFO
USSPARM PARM=STATS,REP=VERB,VALUE=STATS
USSPARM PARM=STATIONS,REP=VERB,VALUE=STATIONS
USSPARM PARM=STORUSE,REP=VERB,VALUE=STORUSE
USSPARM PARM=SUM,REP=LIST,VALUE=SUMMARY
USSPARM PARM=SUMMARY,REP=LIST,VALUE=SUMMARY
USSPARM PARM=TABLE,REP=VERB,VALUE=TABLE
USSPARM PARM=TERMS,REP=VERB,VALUE=TERMS
USSPARM PARM=TP,REP=VERB,VALUE=TP
USSPARM PARM=TSOUSER,REP=VERB,VALUE=TSOUSER
USSPARM PARM=USERVAR,REP=VERB,VALUE=USERVAR
USSPARM PARM=VTAMOPTS,REP=VERB,VALUE=VTAMOPTS
USSPARM PARM=VTAMSTOR,REP=VERB,VALUE=VTAMSTOR

******************************************************************************
** DISPLAY ADJCP COMMAND
******************************************************************************
USSCMD CMD=ADJCP,FORMAT=BAL

******************************************************************************
** DISPLAY ADJSSCPS COMMAND
******************************************************************************
USSCMD CMD=ADJSSCPS,FORMAT=BAL
USSPARM PARM=CDRM
USSPARM PARM=CDRSRC
USSPARM PARM=SCOPE,DEFAULT=ONLY
USSPARM PARM=MXX

******************************************************************************
** DISPLAY APPLS COMMAND
******************************************************************************
USSCMD CMD=APPLS,FORMAT=BAL
USSPARM PARM=MAX
USSPARM PARM=SCOPE,DEFAULT=ALL

******************************************************************************
** DISPLAY APING COMMAND
******************************************************************************
USSCMD CMD=APING,FORMAT=BAL
USSPARM PARM=CONSEC,DEFAULT=1
USSPARM PARM=ECHO,DEFAULT=YES
USSPARM PARM=ITER,DEFAULT=2
USSPARM PARM=LIST,DEFAULT=ALL
USSPARM PARM=LOGMODE,DEFAULT=#INTER
USSPARM PARM=PASSWORD
USSPARM PARM=SIZE,DEFAULT=100
USSPARM PARM=TP,DEFAULT=APINGD
USSPARM PARM=USERID

******************************************************************************
1134 z/OS V2R1.0 Communications Server: SNA Resource Definition Reference
**DISPLAY APINGDTF COMMAND**

USSCMD CMD=APINGDTF,FORMAT=BAL
USSPARAM PARM=LIST
USSPARAM PARM=MAX

**DISPLAY APINGTP COMMAND**

USSCMD CMD=APINGTP,FORMAT=BAL
USSPARAM PARM=LIST
USSPARAM PARM=MAX

**DISPLAY APPNTOSA COMMAND**

USSCMD CMD=APPNTOSA,FORMAT=BAL

**DISPLAY AUTOLOG COMMAND**

USSCMD CMD=AUTOLOG,FORMAT=BAL
USSPARAM PARM=SCOPE,DEFAULT=ONLY

**DISPLAY BFRUSE COMMAND**

USSCMD CMD=BFRUSE,FORMAT=BAL
USSPARAM PARM=BUF,REP=BUFFER
USSPARAM PARM=BUFFER,DEFAULT=*  

**DISPLAY BNCSMAP COMMAND**

USSCMD CMD=BNCSMAP,FORMAT=BAL

**DISPLAY CDRMS COMMAND**

USSCMD CMD=CDRMS,FORMAT=BAL
USSPARAM PARM=MAX
USSPARAM PARM=SCOPE,DEFAULT=ALL

**DISPLAY CDRSCS COMMAND**

USSCMD CMD=CDRSCS,FORMAT=BAL
USSPARAM PARM=MAX
USSPARAM PARM=SCOPE,DEFAULT=ALL

**DISPLAY CLSTRS COMMAND**

USSCMD CMD=CLSTRS,FORMAT=BAL
USSPARAM PARM=MAX
USSPARAM PARM=OWNER
USSPARAM PARM=SCOPE,DEFAULT=ALL
USSPARAM PARM=LIST,DEFAULT=ALL

**DISPLAY CNOS COMMAND**

USSCMD CMD=CNOS,FORMAT=BAL
USSPARAM PARM=LOG,REP=LOGMODE
USSPARAM PARM=LU,REP=LUNAME

**DISPLAY CNON COMMAND**

USSCMD CMD=CNON,FORMAT=BAL
USSPARAM PARM=LOG,REP=LOGMODE
USSPARAM PARM=LU,REP=LUNAME

**DISPLAY CONVID COMMAND**

USSCMD CMD=CONVID,FORMAT=BAL
USSPARAM PARM=ETIME,DEFAULT=0
USSPARAM PARM=LOG,REP=LOGMODE
USSPARAM PARM=LU,REP=LUNAME

**DISPLAY COS COMMAND**

USSCMD CMD=COS,FORMAT=BAL

**DISPLAY CPCP COMMAND**

USSCMD CMD=CPCP,FORMAT=BAL
USSPARAM PARM=SCOPE,DEFAULT=ALL
USSPARAM PARM=LIST,DEFAULT=ALL
USSPARAM PARM=MAX
+ **DISPLAY CPS COMMAND**
+ **DISPLAY CSDUMP COMMAND**
+ **DISPLAY DIRECTRY COMMAND**
+ **DISPLAY DISK COMMAND**
+ **DISPLAY EE COMMAND**
+ **DISPLAY EEDIAG COMMAND**
+ **DISPLAY EXIT COMMAND**
+ **DISPLAY GRAFFIN COMMAND**
+ **DISPLAY GROUPS COMMAND**
+ **DISPLAY GRPREFS COMMAND**
+ **DISPLAY INOPCODE COMMAND**
+ **DISPLAY INOPDUMP COMMAND**
+ **DISPLAY LINES COMMAND**
Appendix A. IBM-supplied tables

1137
DISPLAY RTPS COMMAND

USSCMD CMD=RTPS,FORMAT=BAL
USSPARAM PARM=APPNCOS
USSPARAM PARM=CLEAR,REP=CLEAR,VALUE=ALL
USSPARAM PARM=CONGEST,DEFAULT=ALL
USSPARAM PARM=CNAME
USSPARAM PARM=LIST,DEFAULT=DETAIL
USSPARAM PARM=MAX
USSPARAM PARM=SWITCH,DEFAULT=ALL
USSPARAM PARM=STALL,DEFAULT=ALL

DISPLAY SAMAP COMMAND

USSCMD CMD=SAMAP,FORMAT=BAL

DISPLAY SATOAPPN COMMAND

USSCMD CMD=SATOAPPN,FORMAT=BAL

DISPLAY SESSIONS COMMAND

USSCMD CMD=SESSIONS,FORMAT=BAL
USSPARAM PARM=LIST,DEFAULT=COUNT
USSPARAM PARM=LUI
USSPARAM PARM=LUI2
USSPARAM PARM=LUI
USSPARAM PARM=MAX
USSPARAM PARM=PLUI
USSPARAM PARM=SCOPE,DEFAULT=ALL
USSPARAM PARM=SID
USSPARAM PARM=SLUI
USSPARAM PARM=PATH,REP=PATHINFO,VALUE=YES
USSPARAM PARM=PATHINFO,DEFAULT=YES

DISPLAY SNSFILTR COMMAND

USSCMD CMD=SNSFILTR,FORMAT=BAL

DISPLAY SRCHINFO COMMAND

USSCMD CMD=SRCHINFO,FORMAT=BAL
USSPARAM PARM=DU
USSPARAM PARM=FROMCP
USSPARAM PARM=FROMSSCP
USSPARAM PARM=LIST,DEFAULT=SUMMARY
USSPARAM PARM=LUI
USSPARAM PARM=LUI
USSPARAM PARM=MAX
USSPARAM PARM=PLUI
USSPARAM PARM=PCID,REP=SID
USSPARAM PARM=SID
USSPARAM PARM=TCP
USSPARAM PARM=TYPE,DEFAULT=ALL

DISPLAY STATIONS COMMAND

USSCMD CMD=STATIONS,FORMAT=BAL
USSPARAM PARM=MAX
USSPARAM PARM=SCOPE,DEFAULT=ALL

DISPLAY STATS COMMAND

USSCMD CMD=STATS,FORMAT=BAL
USSPARAM PARM=NUM,REP=MAX
USSPARAM PARM=MAX
USSPARAM PARM=NUM,REP=STRNAME
USSPARAM PARM=STRNAME
USSPARAM PARM=TYPE

DISPLAY STORUSE COMMAND

USSCMD CMD=STORUSE,FORMAT=BAL
USSPARAM PARM=APPL
USSPARAM PARM=APPL
USSPARAM PARM=APPL
USSPARAM PARM=JOBNAME
USSPARAM PARM=LIST
USSPARAM PARM=POOLTYPE
USSPARAM PARM=NUM,REP=MAX
USSPARM PARM=MAX
USSPARM PARM=POOL

DISPLAY TABLE COMMAND
USSCMD CMD=DTABLE,FORMAT=BAL
USSPARM PARM=SCOPE,DEFAULT=ONLY
USSPARM PARM=MAX

DISPLAY TERMS COMMAND
USSCMD CMD=TERMS,FORMAT=BAL
USSPARM PARM=MAX
USSPARM PARM=SCOPE,DEFAULT=ALL

DISPLAY TGPS COMMAND
USSCMD CMD=TPS,FORMAT=BAL
USSPARM PARM=MAX

DISPLAY TNSTAT COMMAND
USSCMD CMD=TNSTAT

DISPLAY TOPO COMMAND
USSCMD CMD=TOPO,FORMAT=BAL
USSPARM PARM=CLEAR,REP=CLEAR,VALUE=YES

DISPLAY TRACES COMMAND
USSCMD CMD=TRACES,FORMAT=BAL
USSPARM PARM=IDTYPE,DEFAULT=RESOURCE
USSPARM PARM=TYPE,DEFAULT=VTAM

DISPLAY TRL COMMAND
USSCMD CMD=TRL,FORMAT=BAL
USSPARM PARM=CONTROL,DEFAULT=ALL
USSPARM PARM=DEVSTATS
USSPARM PARM=XCFCP
USSPARM PARM=TRL

DISPLAY TSOUSER COMMAND
USSCMD CMD=TSOUSER,FORMAT=BAL

DISPLAY USERVAR COMMAND
USSCMD CMD=USERVAR,FORMAT=BAL
USSPARM PARM=MAX

DISPLAY VTAMOPTS COMMAND
USSCMD CMD=VTAMOPTS,FORMAT=BAL
USSPARM PARM=COMP,REP=FORMAT,VALUE=COMPLETE
USSPARM PARM=CUR,REP=FORMAT,VALUE=CURRENT
USSPARM PARM=FUNCTION
USSPARM PARM=MOD,REP=FORMAT,VALUE=MODIFIED
USSPARM PARM=OPT,REP=OPTION

DISPLAY VTAMSTOR COMMAND
USSCMD CMD=VTAMSTOR,FORMAT=BAL
USSPARM PARM=MOD,REP=MODULE
USSPARM PARM=MODULE
USSPARM PARM=RSC,REP=RESOURCE
USSPARM PARM=RESOURCE
USSPARM PARM=NETADDR
USSPARM PARM=NETID
USSPARM PARM=ADDR,REP=ADDRESS
USSPARM PARM=ADDRESS
USSPARM PARM=LEN,REP=LENGTH
USSPARM PARM=LENGTH
USSPARM PARM=XLEN,REP=XLENGTH
USSPARM PARM=XLENGTH
*****************************************************************************
* MODIFY COMMAND
*****************************************************************************
USSCMD CMD=MODIFY, FORMAT=BAL
USSPARM PARM=ALSLIST, REP=VERB, VALUE=ALSLIST
USSPARM PARM=APIGTP, REP=VERB, VALUE=APIGTP
USSPARM PARM=BFRUSE, REP=VERB, VALUE=BFRUSE
USSPARM PARM=CANCEL, REP=OPTION, VALUE=CANCEL
USSPARM PARM=CDMM
USSPARM PARM=CHGPT, REP=VERB, VALUE=CHGPT
USSPARM PARM=CNS, REP=VERB, VALUE=CNS
USSPARM PARM=CNOSL, REP=CNOSL, VALUE=YES
USSPARM PARM=CMPRESS, REP=VERB, VALUE=CMPRESS
USSPARM PARM=CSL, REP=VERB, VALUE=CSL
USSPARM PARM=CSL, REP=VERB, VALUE=CSL
USSPARM PARM=DEFULTS, REP=VERB, VALUE=DEFULTS
USSPARM PARM=DEFINE, REP=VERB, VALUE=DEFINE
USSPARM PARM=DETACH, REP=VERB, VALUE=DETACH
USSPARM PARM=DIRECTRY, REP=VERB, VALUE=DIRECTRY
USSPARM PARM=DUMP, REP=VERB, VALUE=DUMP
USSPARM PARM=DUMP, REP=VERB, VALUE=DUMP
USSPARM PARM=ENCE, REP=SCOPE, VALUE=ENCE
USSPARM PARM=EVERY, REP=SCOPE, VALUE=EVERY
USSPARM PARM=EXIT, REP=VERB, VALUE=EXIT
USSPARM PARM=F, REP=TYPE, VALUE=F
USSPARM PARM=FORCE, REP=TYPE, VALUE=FORCE
USSPARM PARM=GR, REP=VERB, VALUE=GR
USSPARM PARM=I, REP=TYPE, VALUE=I
USSPARM PARM=IMM, REP=TYPE, VALUE=IMM
USSPARM PARM=INP, REP=VERB, VALUE=INP
USSPARM PARM=INPMP, REP=VERB, VALUE=INPMP
USSPARM PARM=IOPD
USSPARM PARM=LINEDEF, REP=VERB, VALUE=LINEDEF
USSPARM PARM=L2, REP=VERB, VALUE=L2
USSPARM PARM=LOAD, REP=VERB, VALUE=LOAD
USSPARM PARM=MSG, REP=VERB, VALUE=MSG
USSPARM PARM=N, REP=TYPE, VALUE=N
USSPARM PARM=NCP, REP=VERB, VALUE=NCP
USSPARM PARM=NEGPOOL
USSPARM PARM=NCNSL, REP=CNOSL, VALUE=NO
USSPARM PARM=NOR, REP=TYPE, VALUE=NOR
USSPARM PARM=NOSTAT, REP=VERB, VALUE=NOSTAT
USSPARM PARM=NOTRACE, REP=VERB, VALUE=NOTRACE
USSPARM PARM=OPT, REP=OPTION
USSPARM PARM=POLL
USSPARM PARM=PROFILES, REP=VERB, VALUE=PROFILES
USSPARM PARM=QUERY, REP=VERB, VALUE=QUERY
USSPARM PARM=RESOURCE, REP=VERB, VALUE=RESOURCE
USSPARM PARM=RMOPO, REP=RMOPO, VALUE=NO
USSPARM PARM=RT, REP=VERB, VALUE=RT
USSPARM PARM=SECURITY, REP=VERB, VALUE=SECURITY
USSPARM PARM=SESSION
USSPARM PARM=SUBTASK, REP=VERB, VALUE=SUBTASK
USSPARM PARM=TABLE, REP=VERB, VALUE=TABLE
USSPARM PARM=TG, REP=VERB, VALUE=TG
USSPARM PARM=THG, REP=VERB, VALUE=THG
USSPARM PARM=TNSTAT, REP=VERB, VALUE=TNSTAT
USSPARM PARM=TP, REP=VERB, VALUE=TP
USSPARM PARM=TPRINT, REP=VERB, VALUE=TPRINT
USSPARM PARM=TR, REP=VERB, VALUE=TR
USSPARM PARM=TYPE
USSPARM PARM=USERVAR, REP=VERB, VALUE=USERVAR
USSPARM PARM=VTAMOPTS, REP=VERB, VALUE=VTAMOPTS
*****************************************************************************
* MODIFY ALSLIST COMMAND
*****************************************************************************
USSCMD CMD=ALSLIST
USSPARM PARM=ACTION
USSPARM PARM=OPTION
USSPARM PARM=NEWALS
USSPARM PARM=OLDALS
USSPARM PARM=OLDALS

1140  z/OS V2R1.0 Communications Server: SNA Resource Definition Reference
* MODIFY APINGDTP COMMAND
******
** USSCMD CMD=APINGDTP,FORMAT= BAL
** USSPARM PARM=INSTANCE
******
* MODIFY APINGTP COMMAND
******
** USSCMD CMD=APINGTP,FORMAT= BAL
** USSPARM PARM=INSTANCE
******
* MODIFY ATTACH COMMAND (VM AND VSE ONLY)
******
** USSCMD CMD=ATTACH,REP=SUBTASK
** USSPARM PARM=FUNCTION,DEFAULT=ATTACH
** USSPARM PARM=PARM (VM ONLY)
** USSPARM PARM=SUPVR,DEFAULT=NO (VM ONLY)
******
* MODIFY BFRUSE COMMAND
******
** USSCMD CMD=BFRUSE,FORMAT= BAL
** USSPARM PARM=BUF,REP=BUFFER
** USSPARM PARM=BUFFER,DEFAULT=IOBUF
******
* MODIFY CHKPT COMMAND
******
** USSCMD CMD=CHKPT,FORMAT= BAL
** USSPARM PARM=TYPE,DEFAULT=ALL
******
* MODIFY COMPRESS COMMAND
******
** USSCMD CMD=COMPRESS,FORMAT= BAL
** USSPARM PARM=CMPAPLI
** USSPARM PARM=CMPAPPLO
** USSPARM PARM=CMPVTAM
******
* MODIFY CNOS COMMAND
******
** USSCMD CMD=CNOS
** USSPARM PARM=CSCECL,REP=CONVSECL
** USSPARM PARM=CONVSECL
** USSPARM PARM=DRL,REP=DRAINL,DEFAULT=NO
** USSPARM PARM=DRAINL,DEFAULT=NO
** USSPARM PARM=DLL,REP=DRAINR,DEFAULT=NO
** USSPARM PARM=DRAINR,DEFAULT=NO
** USSPARM PARM=LUNAME
** USSPARM PARM=LUNAME
** USSPARM PARM=NBM,REP=NBRMODE,DEFAULT=ONE
** USSPARM PARM=NBRMODE,DEFAULT=ONE
** USSPARM PARM=RESP,REP=RESP,DEFAULT=LOCAL
** USSPARM PARM=SNG,REP=SNGSLESU,DEFAULT=NO
** USSPARM PARM=SNGSLESU,DEFAULT=NO
******
* MODIFY CSDUMP COMMAND
******
** USSCMD CMD=CSDUMP
** USSPARM PARM=SENSE
** USSPARM PARM=RU
** USSPARM PARM=MESSAGE
** USSPARM PARM=MATCHLIM,DEFAULT=1
** USSPARM PARM=DELETE
** USSPARM PARM=REMOTE,DEFAULT=NO
******
* MODIFY DEFAULTS COMMAND
******
** USSCMD CMD=DEFAULTS
** USSPARM PARM=DLOG,REP=DLOGMOD
** USSPARM PARM=DLOGMOD
******
* MODIFY DEFINE COMMAND
******
** USSCMD CMD=DEFINE
** USSPARM PARM=AUTO,REP=AUTOSES
** USSPARM PARM=AUTOSES
** USSPARM PARM=DEL,REP=DELETE,DEFAULT=NALLOW
** USSPARM PARM=DELETE,DEFAULT=NALLOW
USSPARM PARM=USE

* MODIFY LL2 COMMAND

USSCMD CMD=LL2
USSPARM PARM=CANCEL,REP=OPTION,VALUE=CANCEL
USSPARM PARM=CONT,REP=OPTION,VALUE=CONT
USSPARM PARM=DATA
USSPARM PARM=NFRAMES,DEFAULT=1
USSPARM PARM=NTRANS,DEFAULT=10

* MODIFY LOAD COMMAND

USSCMD CMD=LOAD,FORMAT=BAL
USSPARM PARM=ACTION,DEFAULT=ADD
USSPARM PARM=LM,REP=LOADMOD
USSPARM PARM=LOADMOD
USSPARM PARM=NEWN,REP=NEWNAME
USSPARM PARM=NEWNAME
USSPARM PARM=IPLT,REP=IPLTIME
USSPARM PARM=IPLTIME
USSPARM PARM=NOTIFY
USSPARM PARM=NT,REP=NOTIFY

* MODIFY MSG COMMAND (VSE ONLY)

USSCMD CMD=MSG,REP=SUBTASK
USSPARM PARM=FUNCTION,DEFAULT=MSG

* MODIFY NCP COMMAND

USSCMD CMD=NCP,FORMAT=BAL
USSPARM PARM=COM,REP=COMMAND
USSPARM PARM=CMD,REP=COMMAND
USSPARM PARM=COMMAND,DEFAULT=TRSWITCH
USSPARM PARM=RES,REP=RESNM
USSPARM PARM=NAME,REP=RESNM
USSPARM PARM=RESNAME,REP=RESNM
USSPARM PARM=RESNM

* MODIFY NOTNSTAT COMMAND

USSCMD CMD=NOTNSTAT

* MODIFY NOTRACE COMMAND

USSCMD CMD=NOTRACE
USSPARM PARM=ALS,REP=ALSNAME
USSPARM PARM=DEV,REP=DEVICE
USSPARM PARM=IDTYPE,DEFAULT=RESOURCE
USSPARM PARM=LINE
USSPARM PARM=MODE
USSPARM PARM=NONE,REP=SCOPE,VALUE=ONLY
USSPARM PARM=PU
USSPARM PARM=SCOPE,DEFAULT=ONLY

* MODIFY PROFILES COMMAND

USSCMD CMD=PROFILES,FORMAT=BAL

* MODIFY QUERY COMMAND (POA ONLY)

USSCMD CMD=QUERY,FORMAT=BAL
USSPARM PARM=CONT,REP=CONTINUE
USSPARM PARM=NO,REP=CONTINUE,VALUE=NO
USSPARM PARM=RESET,REP=CONTINUE,VALUE=RESET
USSPARM PARM=YES,REP=CONTINUE,VALUE=YES
USSPARM PARM=CONTINUE,DEFAULT=NO
USSPARM PARM=VECTOR,REP=VECTORS
USSPARM PARM=VECTOR,REP=VECTORS

* MODIFY RESOURCE COMMAND

USSCMD CMD=RESOURCE
USSPARM PARM=ACTION
USSPARM PARM=ADJLIST
USSPARM PARM=LOG,REP=LOGMOD
USSPARM PARM=LOGMOD

Appendix A. IBM-supplied tables 1143
**MODIFY SECURITY COMMAND**

**USSCMD** CMD=SECURITY
**USSPARM** PARM=CKEY
**USSPARM** PARM=ENCR
**USSPARM** PARM=MAC
**USSPARM** PARM=MACLNTH, DEFAULT=4
**USSPARM** PARM=MACTYPE, DEFAULT=CRC

**MODIFY SUBTASK COMMAND (VM AND VSE ONLY)**

**USSCMD** CMD=SUBTASK
**USSPARM** PARM=FUNCTION, DEFAULT=ATTACH
**USSPARM** PARM=PARM (VM ONLY)
**USSPARM** PARM=SUPVR, DEFAULT=NO (VM ONLY)

**MODIFY TABLE COMMAND**

**USSCMD** CMD=TABLE, FORMAT=BAL
**USSPARM** PARM=NET, REP=NETID
**USSPARM** PARM=NETID
**USSPARM** PARM=NEW, REP=NEWTAB
**USSPARM** PARM=NEWTAB
**USSPARM** PARM=OLD, REP=OLDTAB
**USSPARM** PARM=OLDTAB
**USSPARM** PARM=OPTION
**USSPARM** PARM=ORIGIN

**MODIFY TGP COMMAND**

**USSCMD** CMD=TGP, FORMAT=BAL
**USSPARM** PARM=TGN
**USSPARM** PARM=TGPNAME

**MODIFY TOPO COMMAND**

**USSCMD** CMD=TOPO, FORMAT=BAL
**USSPARM** PARM=DELETE, REP=FUNCTION, VALUE=DELETE
**USSPARM** PARM=DEST
**USSPARM** PARM=FUNCTION
**USSPARM** PARM=ORIG
**USSPARM** PARM=SCOPE
**USSPARM** PARM=TGN
**USSPARM** PARM=VRN

**MODIFY TNSTAT COMMAND**

**USSCMD** CMD=TNSTAT
**USSPARM** PARM=A, REP=ACTION, VALUE=ACTIVATE
**USSPARM** PARM=A, REP=ACTION, VALUE=ACTIVATE
**USSPARM** PARM=U, REP=ACTION, VALUE=UPDATE
**USSPARM** PARM=U, REP=ACTION, VALUE=UPDATE
**USSPARM** PARM=ACTION, DEFAULT=ACTIVATE
**USSPARM** PARM=CN5L, REP=CN5L, VALUE=YES
**USSPARM** PARM=CN5L
**USSPARM** PARM=CN5L, REP=CN5L, VALUE=NO
**USSPARM** PARM=TIME
**USSPARM** PARM=TRLE

**MODIFY TPRINT COMMAND (VSE ONLY)**

**USSCMD** CMD=TPRINT, REP=SUBTASK
**USSPARM** PARM=FUNCTION, DEFAULT=ATTACH
**USSPARM** PARM=ID, DEFAULT=TPRINT

**MODIFY TRACE COMMAND**

**USSCMD** CMD=TRACE
**USSPARM** PARM=ALS, REP=ALSNAME
**USSPARM** PARM=AL, REP=ALNAME
**USSPARM** PARM=AMOUNT, DEFAULT=PARTIAL
**USSPARM** PARM=COUNT, DEFAULT=ALL
**USSPARM** PARM=DEV, REP=DEVICE
**USSPARM** PARM=FRAMES, DEFAULT=DATA
**USSPARM** PARM=TYPE, DEFAULT=RESOURCE
**USSPARM** PARM=LEN, REP=LENGTH, DEFAULT=272
**USSPARM** PARM=LINE
**USSPARM** PARM=MODE
USSPARM PARM=NONE,REP=SCOPE,VALUE=ONLY
USSPARM PARM=PU
USSPARM PARM=SAVE,REP=SAVE,VALUE=YES
USSPARM PARM=SAVE,DEFAULT=NO
USSPARM PARM=SCOPE,DEFAULT=ONLY
USSPARM PARM=SIZE
USSPARM PARM=TRACEPT

* MODIFY USERVAR COMMAND

USSCMD CMD=FUSERVAR,FORMAT=BAL
USSPARM PARM=APPC,DEFAULT=NO
USSPARM PARM=OPT,REP=OPTION,DEFAULT=UPDATE
USSPARM PARM=OPTION,DEFAULT=UPDATE
USSPARM PARM=TYPE,DEFAULT=DYNAMIC
USSPARM PARM=UVEXIT,DEFAULT=NO
USSPARM PARM=VAL,REP=VALUE
USSPARM PARM=VALUE

* MODIFY VTAMOPTS COMMAND

USSCMD CMD=FVTAMOPT,FORMAT=BAL
USSPARM PARM=CSALIMIT
USSPARM PARM=INOPCODE
USSPARM PARM=INOPDUMP
USSPARM PARM=IOPD
USSPARM PARM=IOPURGE
USSPARM PARM=MSGMOD
USSPARM PARM=PPOLOG
USSPARM PARM=SGALIMIT
USSPARM PARM=SUPP

* VARY COMMANDS

USSCMD CMD=VARY,FORMAT=BAL
USSPARM PARM=ACQ,REP=ACQ
USSPARM PARM=ACT,REP=VERB,VALUE=ACT
USSPARM PARM=ANS
USSPARM PARM=AUTOLOG,REP=VERB,VALUE=AUTOLOG
USSPARM PARM=CFS,REP=VERB,VALUE=CFS
USSPARM PARM=DIAL,REP=VERB,VALUE=DIAL
USSPARM PARM=DRDS,REP=VERB,VALUE=DRDS
USSPARM PARM=HANGUP,REP=VERB,VALUE=HANGUP
USSPARM PARM=ID
USSPARM PARM=INACT,REP=VERB,VALUE=INACT
USSPARM PARM=INOP,REP=VERB,VALUE=INOP
USSPARM PARM=LM,REP=LOADMOD
USSPARM PARM=LOADMOD
USSPARM PARM=LOGMODE
USSPARM PARM=LOGON
USSPARM PARM=LOGON
USSPARM PARM=LOADSTA
USSPARM PARM=LOADSTA
USSPARM PARM=LOG,REP=LOGMODE
USSPARM PARM=TERM,REP=VERB,VALUE=TERM

* VARY ACT

USSCMD CMD=ACT,FORMAT=BAL
USSPARM PARM=ALL,REP=SCOPE,VALUE=ALL
USSPARM PARM=ALL,REP=SCOPE,VALUE=ALL
USSPARM PARM=COMP,REP=SCOPE,VALUE=COMP
USSPARM PARM=CPCP
USSPARM PARM=DL,REP=DUMPLoad
USSPARM PARM=DS,REP=DUMPLoad
USSPARM PARM=EXT,REP=DUMPLoad
USSPARM PARM=EXT,REP=DUMPLoad
USSPARM PARM=EXT,REP=DUMPLoad
USSPARM PARM=HOST,REP=DUMPLoad,VALUE=HOST
USSPARM PARM=Host
USSPARM PARM=HOST,REP=DUMPLoad,VALUE=HOST
USSPARM PARM=LOAD,DEFAULT=U
USSPARM PARM=LOADFORCE
USSPARM PARM=LOADSTA
USSPARM PARM=LOADSTA
USSPARM PARM=LOGMODE

Appendix A. IBM-supplied tables 1145
USSPARM PARM=GIVEBACK,REP=TYPE,VALUE=GIVEBACK
USSPARM PARM=I,REP=TYPE,VALUE=IMMED
USSPARM PARM=IMMED,REP=TYPE,VALUE=IMMED
USSPARM PARM=OWNER

***********************************************************************
* VARY TERM COMMAND (SESSION SERVICES)
***********************************************************************
USSCMD CMD=TERM,FORMAT=BAL
USSPARM PARM=LU1
USSPARM PARM=LU2
USSPARM PARM=NOTIFY,DEFAULT=YES
USSPARM PARM=PLU
USSPARM PARM=SCOPE,DEFAULT=ACT
USSPARM PARM=SID
USSPARM PARM=SLU
USSPARM PARM=TYPE,DEFAULT=UNCOND

***********************************************************************
* BEGINNING OF MESSAGE DEFINITIONS
***********************************************************************

USSMSG MSG=IST001I, MESSAGE ID-
  TEXT='IST001I VTAM START REJECTED - %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST003I, MESSAGE ID-
  TEXT='IST003I ABEND OCCURRED DURING NETWORK DEFINITION OF CONFIG %, CODE = %2',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=SER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST009I, MESSAGE ID-
  TEXT='IST009I VTAM ALREADY ACTIVE - START REJECTED',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST010I, MESSAGE ID-
  TEXT='IST010I % COMMAND INVALID',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=SER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST011I, MESSAGE ID-
  TEXT='IST011I % FOR %2 %3',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NORM, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST013I, MESSAGE ID-
  TEXT='IST013I I/O ERROR FOR % IN %2',
  ROUTCDE=(2,10), ROUTING CODE-
  DESC=4, DESCRIPTOR CODE-
  SUPP=INFO, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST014I, MESSAGE ID-
  TEXT='IST014I DEVICE SUPPORT MAY BE UNAVAILABLE - SYS000 NOT UNASSIGNED',
  ROUTCDE=1, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=BLKSUP, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST015A, MESSAGE ID-
  TEXT='IST015A ERROR PROCESSING LIST IDENTIFIER - ENTER LIST ID OR BLANK',
  ROUTCDE=2, ROUTING CODE-
  DESC=2, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST017I, MESSAGE ID-
  TEXT='IST017I UNABLE TO LOAD PHASE %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=BLKSUP, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST018I, MESSAGE ID-
  TEXT='IST018I CONFIG COULD NOT BE INITIALIZED - VTAM START CONTINUES',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=BLKSUP, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

Appendix A. IBM-supplied tables  1147
USSMSG MSG=IST097I, MESSAGE ID-
  TEXT='IST097I % ACCEPTED',
  ROUTCODE=8, ROUTING CODE-
  DESC=5, DESCRIP Code-
  SUPP=NEVER, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST101I, MESSAGE ID-
  TEXT='IST101I % FAILED - %(2) NOT SPECIFIED',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIP Code-
  SUPP=SER, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST102I, MESSAGE ID-
  TEXT='IST102I VTAM IS NOW INACTIVE',
  ROUTCODE=(2,8), ROUTING CODE-
  DESC=4, DESCRIP Code-
  SUPP=NEVER, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST105I, MESSAGE ID-
  TEXT='IST105I % NODE NOW INACTIVE',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIP Code-
  SUPP=NORM, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST107I, MESSAGE ID-
  TEXT='IST107I TIME AND DATE NOT SET IN % DUE TO INVALID -
  TIMER IN HOST',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIP Code-
  SUPP=WARN, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST109I, MESSAGE ID-
  TEXT='IST109I % IS NOW TERMINATED',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIP Code-
  SUPP=INFO, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST112I, MESSAGE ID-
  TEXT='IST112I VTAM INTERNAL TRACE MODIFY FAILED - CONFLI-
  CTING MODES',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIP Code-
  SUPP=NEVER, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST113I, MESSAGE ID-
  TEXT='IST113I % IS A USERVAR WITH VALUE %(2) IN NETWORK-
  %(3)',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIP Code-
  SUPP=NEVER, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST115I, MESSAGE ID-
  TEXT='IST115I INSUFFICIENT STORAGE TO READ % MEMBER OF V-
  TAM DEFINITION LIBRARY',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIP Code-
  SUPP=WARN, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST116I, MESSAGE ID-
  TEXT='IST116I MEMBER % NOT FOUND ON VTAM DEFINITION LIB-
  RARY',
  ROUTCODE=2, ROUTING CODE-
  DESC=6, DESCRIP Code-
  SUPP=WARN, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST117I, MESSAGE ID-
  TEXT='IST117I I/O ERROR READING % MEMBER OF VTAM DEFINIT-
  ION LIBRARY',
  ROUTCODE=(2,10), ROUTING CODE-
  DESC=4, DESCRIP Code-
  SUPP=WARN, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST118I, MESSAGE ID-
  TEXT='IST118I ANOMALY FOUND NEAR RECORD % IN MEMBER %(2)-
  CODE %(3)',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIP Code-
  SUPP=WARN, SUPPRESSION Code-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

Appendix A. IBM-supplied tables

1151
USSMSG MSG=IST167I, MESSAGE ID-
  TEXT='IST167I NO DIAL OUT PATH FOR %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST168I, MESSAGE ID-
  TEXT='IST168I % %(2) %(3) %(4) %(5)',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST169I, MESSAGE ID-
  TEXT='IST169I DISCONNECTION CAUSED VARY % FOR PU - %(2)',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=INFO, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST170I, MESSAGE ID-
  TEXT='IST170I LINES:',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST171I, MESSAGE ID-
  TEXT='IST171I ACTIVE SESSIONS = %, SESSION REQUESTS = %(-2)',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST172I, MESSAGE ID-
  TEXT='IST172I NO % %(2)',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST176I, MESSAGE ID-
  TEXT='IST176I % FAILED - %(2) AND %(3) ARE CONFLICTING O-
  PTIONS',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=SER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST180I, MESSAGE ID-
  TEXT='IST180I OPEN FAILED ON CKPT DS % MAJ NODE %(2) RTN-
  CD = %(3) %(4)',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST181I, MESSAGE ID-
  TEXT='IST181I CLOSE FAILED ON CKPT DS % MAJ NODE %(2) RT-
  NCD = %(3) %(4)',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST182I, MESSAGE ID-
  TEXT='IST182I UNABLE TO GET STORAGE FOR CKPT % MAJOR NOD-
  E %(2)',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST183A, MESSAGE ID-
  TEXT='IST183A % FOUND LOADED WITH %(2) - REPLY ''YES'' T-
  O REIPL OR ''NO'' TO CONTINUE',
  ROUTCDE=2, ROUTING CODE-
  DESC=2, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST184I, MESSAGE ID-
  TEXT='IST184I I/O ERROR ON CKPT DS % MAJ NODE %(2) RTN-
  CD = %(3) %(4)',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
Appendix A. IBM-supplied tables
USSMSG MSG=IST211I, MESSAGE ID-
  TEXT='IST211I NCP SLOWDOWN INITIATED FOR %',
  ROUTCD=2, ROUTING CODE-
  DESC=4, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST212I, MESSAGE ID-
  TEXT='IST212I ACBNAME = %',
  ROUTCD=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST213I, MESSAGE ID-
  TEXT='IST213I ACBNAME FOR ID = %',
  ROUTCD=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST214I, MESSAGE ID-
  TEXT='IST214I NCP SLOWDOWN TERMINATED FOR %',
  ROUTCD=2, ROUTING CODE-
  DESC=4, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST219I, MESSAGE ID-
  TEXT='IST219I I/O ERROR ON READ FOR % - BYTECNT MISMATCH-',
  ROUTCD=2, ROUTING CODE-
  DESC=4, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST221I, MESSAGE ID-
  TEXT='IST221I % : %(2) IS INVALID, %(3), UNSUPPORTED OPT-
  ION - %(4)',
  ROUTCD=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NORM, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST223I, MESSAGE ID-
  TEXT='IST223I MODIFY % COMMAND COMPLETED',
  ROUTCD=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST225I, MESSAGE ID-
  TEXT='IST225I % FOR ID = %(2) FAILED - %(3)',
  ROUTCD=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=SER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST226I, MESSAGE ID-
  TEXT='IST226I % FOR ID = %(2) NOT EFFECTIVE DURING CURRE-
  NT OR QUEUED SESSIONS',
  ROUTCD=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=INFO, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST228I, MESSAGE ID-
  TEXT='IST228I ENCRYPTION=%, TYPE = %(2)',
  ROUTCD=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST231I, MESSAGE ID-
  TEXT='IST231I % MAJOR NODE = %(2)',
  ROUTCD=(2,8), ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST232I, MESSAGE ID-
  TEXT='IST232I % %(2) %(3)',
  ROUTCD=(2,8), ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST234I, MESSAGE ID-
  TEXT='IST234I I/O ERROR %',
  ROUTCD=(2,8,10), ROUTING CODE-
  DESC=4, DESCRIPTOR CODE-
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables

SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3001I, MESSAGE ID-
   TEXT='IST3001I INSUFFICIENT PRIVATE FIXED STORAGE FOR I/O-
   TRACE TABLE', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=WARN, SUPPRESSION CODE-
   OPT=BLK SUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3021I, MESSAGE ID-
   TEXT='IST3021I INVALID DEFINITION TYPE IN MEMBER % IN VTAM-
   DEFINITION LIBRARY', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=WARN, SUPPRESSION CODE-
   OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3031I, MESSAGE ID-
   TEXT='IST3031I INSUFFICIENT STORAGE TO BUILD CONFIGURATIN-
   N %', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=WARN, SUPPRESSION CODE-
   OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3091I, MESSAGE ID-
   TEXT='IST3091I UNABLE TO LOAD MODULE % FROM LIBRARY %2)', -
   ROUTCDE=(2,4,8), ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=WARN, SUPPRESSION CODE-
   OPT=BLK SUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3101I, MESSAGE ID-
   TEXT='IST3101I INVALID SPACE REQUEST FOR CONFIGURATION %-
   ', ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=WARN, SUPPRESSION CODE-
   OPT=BLK SUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3111I, MESSAGE ID-
   TEXT='IST3111I NCP LOAD MODULE LIBRARY % - FAILED TO OPEN-
   ', ROUTCDE=(2,4,8), ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=WARN, SUPPRESSION CODE-
   OPT=BLK SUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3141I, MESSAGE ID-
   TEXT='IST3141I END', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=BLK SUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3151I, MESSAGE ID-
   TEXT='IST3151I VTAM INTERNAL TRACE ACTIVE - MODE = %, SIZ-
   E = %2) %3) ', ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NORM, SUPPRESSION CODE-
   OPT=BLK SUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3161I, MESSAGE ID-
   TEXT='IST3161I % TRACE USER OPTIONS ARE NOT ACTIVE', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NORM, SUPPRESSION CODE-
   OPT=BLK SUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3171I, MESSAGE ID-
   TEXT='IST3171I VTAM INTERNAL TRACE ACTIVATION FAILED - IN-
   SUFFICIENT STORAGE', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=SER, SUPPRESSION CODE-
   OPT=BLK SUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3191I, MESSAGE ID-
   TEXT='IST3191I CONFIGURATION % FIRST SPECIFICATION USED %-
   (2)', ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NORM, SUPPRESSION CODE-
   OPT=BLK SUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST3201I, MESSAGE ID-
   TEXT='IST3201I DEFINITION % NOT SUCCESSFUL - %2)', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables

1163
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=(IST456I,12), MESSAGE ID-
TEXT='IST456I % REQUIRED PARAMETER OMITTED'
ROUTECODE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=(IST457I,0), MESSAGE ID-
TEXT='IST457I POSITIVE % COMMAND RESPONSE'
ROUTECODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=(IST458I,14), MESSAGE ID-
TEXT='IST458I USS MESSAGE % NOT DEFINED'
ROUTECODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST459I, MESSAGE ID-
TEXT='IST459I % FAILED - ID = %(2) - ADJ NODE %(3)'
ROUTECODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST460I, MESSAGE ID-
TEXT='IST460I % FOR U/RNAME ENTRY ID = %(2) FAILED: %(3)'
ROUTECODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST461I, MESSAGE ID-
TEXT='IST461I % FOR U/RNAME ENTRY ID = %(2) STARTED'
ROUTECODE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST462I, MESSAGE ID-
TEXT='IST462I ACTIVATION OF LINK STATION % IS DEFERRED PENDING HIGHER LEVEL NODE ACTIVATION'
ROUTECODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST464I, MESSAGE ID-
TEXT='IST464I LINK STATION % HAS CONTACTED %(2) SA %(3)'
ROUTECODE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST465I, MESSAGE ID-
TEXT='IST465I % FOR ID = %(2) FAILED - NO %(3) STATION AVAILABLE'
ROUTECODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST466I, MESSAGE ID-
TEXT='IST466I % FOR ID = %(2) CONTINUES - UNABLE TO DO %(3)'
ROUTECODE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST467I, MESSAGE ID-
TEXT='IST467I CONTACTED ERROR TYPE % FOR ID = %(2)'
ROUTECODE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST473I, MESSAGE ID-
TEXT='IST473I CONNECTIVITY TEST TO % TERMINATED AFTER %(2) ECHOES DUE TO I/O ERROR, SENSE = %(3)'
ROUTECODE=(2,8), ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
USSMSG MSG=IST564I, MESSAGE ID-
TEXT='IST564I STORAGE UNAVAILABLE: COMMON AREA SUBPOOL %', -
ROUTECD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST565I, MESSAGE ID-
TEXT='IST565I STORAGE UNAVAILABLE: VTAM PRIVATE AREA SUB-
POOL %', -
ROUTECD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST566I, MESSAGE ID-
TEXT='IST566I STORAGE UNAVAILABLE: % %(%(2) SUBPOOL %(3)', -
ROUTECD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST567I, MESSAGE ID-
TEXT='IST567I % OF %(2) FOR %(3) %(4)', -
ROUTECD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST571I, MESSAGE ID-
TEXT='IST571I LOAD FAILED FOR ID = % REQ: %(2), SENSE: %-
(3)', -
ROUTECD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST572I, MESSAGE ID-
TEXT='IST572I REJECTING TG ADJACENT -
ER MASK', -
ROUTECD=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST574E, MESSAGE ID-
TEXT='IST574 START 1/O TIMEOUT OCCURRED FOR %
', -
ROUTECD=2, ROUTING CODE-
DESC=3, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST576I, MESSAGE ID-
TEXT='IST576I TSO TRACE = %', -
ROUTECD=(2,8), ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST577I, MESSAGE ID-
TEXT='IST577I TIME = % DATE = %(2) ID = %-
(3)', -
ROUTECD=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST578I, MESSAGE ID-
TEXT='IST578I CHNRM = % CHMAX = % RDBUF = %
', -
ROUTECD=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST579I, MESSAGE ID-
TEXT='IST579I ATTN = % TIMERS = % QDPTH = %
', -
ROUTECD=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST580I, MESSAGE ID-
TEXT='IST580I BUFCAP = % PRI = % SLODN = %
', -
ROUTECD=2, ROUTING CODE-
Appendix A. IBM-supplied tables
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST621I, MESSAGE ID-
  TEXT='IST621I % FOR NETWORK RESOURCE %2',
  ROUTCODE=(2,8), ROUTING CODE-
  DESC=4, DESCRIPTOR CODE-
  SUPP=NORM, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST623I, MESSAGE ID-
  TEXT='IST623I % ADJACENT SSCP TABLE %2',
  ROUTCODE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST624I, MESSAGE ID-
  TEXT='IST624I % % % %',
  ROUTCODE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST627I, MESSAGE ID-
  TEXT='IST627I % - INSUFFICIENT STORAGE',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=SER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST632I, MESSAGE ID-
  TEXT='IST632I BUFF BUFF CURR CURR MAX MAX TIMES-
  EXP/CONT EXP',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST633I, MESSAGE ID-
  TEXT='IST633I ID SIZE TOTAL AVAIL TOTAL USED EXP -
  THRESHOLD INCR',
  ROUTCODE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST634I, MESSAGE ID-
  TEXT='IST634I NAME STATUS SID SEND -
  RECV VR TP NETID',
  ROUTCODE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST635I, MESSAGE ID-
  TEXT='IST635I % % % % % % % %',
  ROUTCODE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST636I, MESSAGE ID-
  TEXT='IST636I CDRSCS OWNED BY % -',
  ROUTCODE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST637I, MESSAGE ID-
  TEXT='IST637I SUBAREA = % ELEMENT = %2 SSCPID = %3',
  ROUTCODE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST638I, MESSAGE ID-
  TEXT='IST638I ADJNETSA = %, ADJNETEL = %2',
  ROUTCODE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST639I, MESSAGE ID-
  TEXT='IST639I GWN = %, ADJNET = %2',
  ROUTCODE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST640I, MESSAGE ID-
  TEXT='IST640I % ADDR IN ADJNET - SA = %2, EL = %3',
  ROUTCODE=8, ROUTING CODE-
USSMSG MSG=IST663I, MESSAGE ID-
TEXT='IST663I % REQUEST %(2), SENSE=%(3)', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST664I, MESSAGE ID-
TEXT='IST664I % %(2)=%(3) %(4) %(5)=%(6)', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST670I, MESSAGE ID-
TEXT='IST670I VARY % PROCESSING FOR ID = %(2) COMPLETE', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST674I, MESSAGE ID-
TEXT='IST674I % FOR ID = %(2) CONTINUES - PARM: %(3) IGNORED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST675I, MESSAGE ID-
TEXT='IST675I VR = %, TP = %(2)', -
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST678I, MESSAGE ID-
TEXT='IST678I INSUFFICIENT STORAGE TO SCHEDULE TP END EXIT FOR %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST680I, MESSAGE ID-
TEXT='IST680I CONNECTION REQUEST DENIED - ID = % %(2)', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST683I, MESSAGE ID-
TEXT='IST683I CONNECTION REQUEST DENIED, ID = %', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST684I, MESSAGE ID-
TEXT='IST684I I/O ERR, CSW = %, SENSE = %(2)', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST688I, MESSAGE ID-
TEXT='IST688I VARY FAILED FOR ID = % - INSUFFICIENT STORAGE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST690I, MESSAGE ID-
TEXT='IST690I CONNECTION REQUEST DENIED - INVALID STATION ID = %', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST693I, MESSAGE ID-
TEXT='IST693I UNABLE TO DISCONNECT ID = %', -

ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST700I, MESSAGE ID-
TEXT='IST700I INVALID % - SKIPPING TO NEXT NETWORK STMT - OR EOF', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST700I1, MESSAGE ID-
TEXT='IST700I1 CONFIG % LABEL = %2 STMT TYPE = %3', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST702I, MESSAGE ID-
TEXT='IST702I CONFIG % - UNEXPECTED %2', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST703I, MESSAGE ID-
TEXT='IST703I CONFIG % ADJSSCP DEFINITIONS IGNORED - NO ADJCMD STMT', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST704I, MESSAGE ID-
TEXT='IST704I ADJSSCP TABLE FOR % IGNORED - INSUFFICIENT STORAGE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST707I, MESSAGE ID-
TEXT='IST707I ADJSSCP TABLE BEING MODIFIED BY ACTIVATION OF %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST708I, MESSAGE ID-
TEXT='IST708I % %2 %3', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST709I, MESSAGE ID-
TEXT='IST709I CONFIG % FAILED-%2', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST710I, MESSAGE ID-
TEXT='IST710I CONFIG % NETWORK = %2 %3', -
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST712I, MESSAGE ID-
TEXT='IST712I CONFIG % GWPATH %2 IGNORED - MISSING OPERANDS', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST713I, MESSAGE ID-
TEXT='IST713I CONFIG % GWPATH %2 %3 OPERAND IGNORED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST714I, MESSAGE ID-
TEXT='IST714I CONFIG % GWPATH %2 IGNORED - INVALID STM- T', -

1178 z/OS V2R1.0 Communications Server: SNA Resource Definition Reference
Appendix A. IBM-supplied tables

1179
IST732I % REJECTED DUE TO %(2)

IST734I ACTIVATION OF CDRM % USING GWN %(2) FAILED

IST735I NO ADDRESS TRANSFORMS - REQACTCDRM SENT

IST737I DEFAULT VR LIST USED FOR CDRM % USING GWN - %(2)

IST740I UNABLE TO FREE ALIAS ADDRESSES FOR CDRM % - GWN %(2)

IST742I ACTIVATION OF CDRM % %(2) - GWN PATH NOT AVAILABLE

IST744I CROSS-NETWORK SESSION SETUP FAILED, NETWORK = %

IST745I ACTCDRM TO CDRM = % FAILED, SENSE = %(2)

IST746I BIND FAILED FROM % TO %(2), SENSE = %(3)

IST751I SIO = %, ERROR CT = %(2), CUA = %(3)

IST752I GPT TRACE STATUS = % %(2)

IST755I ALERT FROM PU % FOLLOWS
<table>
<thead>
<tr>
<th>USSMSG</th>
<th>MSG=IST756E</th>
<th>TEXT='IST756E ALERT FROM PU % FOLLOWS',</th>
<th>ROUTCDE=8, ROUTING CODE-</th>
<th>DESC=3, DESCRIPTOR CODE-</th>
<th>SUPP=SER, SUPPRESSION CODE-</th>
<th>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>USSMSG</td>
<td>MSG=IST757E</td>
<td>TEXT='IST757E MOSS UNAVAILABLE - HARDWARE ERROR',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=3, DESCRIPTOR CODE-</td>
<td>SUPP=SER, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST758E</td>
<td>TEXT='IST758E MOSS RELOADED - HARDWARE ERROR',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=3, DESCRIPTOR CODE-</td>
<td>SUPP=SER, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST759E</td>
<td>TEXT='IST759E MOSS DISKETTE UNUSABLE',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=3, DESCRIPTOR CODE-</td>
<td>SUPP=SER, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST760E</td>
<td>TEXT='IST760E MOSS DISKETTE HARDWARE ERROR',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=3, DESCRIPTOR CODE-</td>
<td>SUPP=SER, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST761E</td>
<td>TEXT='IST761E MOSS CONSOLE UNAVAILABLE',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=3, DESCRIPTOR CODE-</td>
<td>SUPP=SER, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST762I</td>
<td>TEXT='IST762I MOSS IN MAINTENANCE MODE',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=4, DESCRIPTOR CODE-</td>
<td>SUPP=NORM, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST763I</td>
<td>TEXT='IST763I PHYSICAL UNIT RELOADED - HARDWARE ERROR',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=4, DESCRIPTOR CODE-</td>
<td>SUPP=NORM, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST764I</td>
<td>TEXT='IST764I PHYSICAL UNIT RELOADED - PRIOR ABEND CODE - WAS %',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=4, DESCRIPTOR CODE-</td>
<td>SUPP=NORM, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST765E</td>
<td>TEXT='IST765E CHANNEL ADAPTER % UNAVAILABLE - HARDWARE ERROR',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=3, DESCRIPTOR CODE-</td>
<td>SUPP=SER, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST766I</td>
<td>TEXT='IST766I DUMP FAILED - NO % DUMP ON %2 DISK(ETTE)- ',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=5, DESCRIPTOR CODE-</td>
<td>SUPP=NORM, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST767E</td>
<td>TEXT='IST767E SCANNER % (%2)-%3) UNAVAILABLE - HARDWARE ERROR',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=3, DESCRIPTOR CODE-</td>
<td>SUPP=SER, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
<tr>
<td>USSMSG</td>
<td>MSG=IST768E</td>
<td>TEXT='IST768E SCANNER % (%2)-%3) UNAVAILABLE - HARDWARE ERROR',</td>
<td>ROUTCDE=8, ROUTING CODE-</td>
<td>DESC=3, DESCRIPTOR CODE-</td>
<td>SUPP=SER, SUPPRESSION CODE-</td>
<td>OPT=BLKSUP MESSAGE PROCESSING OPTIONS</td>
</tr>
</tbody>
</table>

Appendix A. IBM-supplied tables 1181
Appendix A. IBM-supplied tables
SUPP=NORM, OPT=BLKSUP

USSMSG MSG=IST8441, MESSAGE ID- 
TEXT='IST8441 VTAM START REJECTED - % IS DUPLICATE NAME'
ROUTCDE=2, DESC=5
SUPP=NEVER, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8451, MESSAGE ID- 
TEXT='IST8451 X.25 DIAGNOSTIC PACKET %, %2'
ROUTCDE=2, DESC=4
SUPP=WARN, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8461, MESSAGE ID- 
TEXT='IST8461 REAL I/O NOT SUPPORTED BECAUSE %', 
ROUTCDE=2, DESC=5
SUPP=WARN, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8491, MESSAGE ID- 
TEXT='IST8491 % INCONSISTENT WITH USE OF %2 IN %3'
ROUTCDE=2, DESC=5
SUPP=WARN, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8601, MESSAGE ID- 
TEXT='IST8601 DEACTIVATION OF % INCOMPLETE - INSUFFICIENT STORAGE', 
ROUTCDE=2, DESC=5
SUPP=NEVER, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8611, MESSAGE ID- 
TEXT='IST8611 MODETAB=% USSTAB=%%(2) LOGTAB=%%(3)', 
ROUTCDE=2, DESC=5
SUPP=NEVER, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8621, MESSAGE ID- 
TEXT='IST8621 NETID = % COSTABLE = %2', 
ROUTCDE=2, DESC=5
SUPP=NEVER, SUPPRESSION CODE- OPT=NOLBSUP

USSMSG MSG=IST8631, MESSAGE ID- 
TEXT='IST8631 MODIFY TABLE COMMAND FAILED-%', 
ROUTCDE=2, DESC=5
SUPP=NEVER, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8641, MESSAGE ID- 
TEXT='IST8641 NEWTAB=%, OLDTAB=%2, OPT=%3, TYPE=%4'
ROUTCDE=2, DESC=5
SUPP=NEVER, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8651, MESSAGE ID- 
TEXT='IST8651 % COMMAND COMPLETE-%2', 
ROUTCDE=2, DESC=5
SUPP=NEVER, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8661, MESSAGE ID- 
TEXT='IST8661 % HAD NO EFFECT - %2', 
ROUTCDE=2, DESC=5
SUPP=NEVER, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8671, MESSAGE ID- 
TEXT='IST8671 SIT TRACE FOR % FAILED TO ACTIVATE', 
ROUTCDE=2, DESC=5
SUPP=INFO, SUPPRESSION CODE- OPT=BLKSUP

USSMSG MSG=IST8691, MESSAGE ID- 
TEXT='IST8691 USERID = %', 
ROUTCDE=2, DESC=5
SUPP=INFO, SUPPRESSION CODE- OPT=BLKSUP
Appendix A. IBM-supplied tables  1187
USSMSG MSG=IST897I, MESSAGE ID-
TEXT='IST897I % OF %(2) %(3) STARTED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST898I, MESSAGE ID-
TEXT='IST898I GWSELECT = %', -
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST899I, MESSAGE ID-
TEXT='IST899I RETRY OF AUTOLOGON(S) TO % %(2)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST900I, MESSAGE ID-
TEXT='IST900I TRACE PRINT UTILITY STARTED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST901A, MESSAGE ID-
TEXT='IST901A "PRINT" OR "CANCEL" MISSING - REENTER LINE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST902A, MESSAGE ID-
TEXT='IST902A INVALID PRINT OPTION - REENTER LINE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST903A, MESSAGE ID-
TEXT='IST903A INVALID PRINT NODENAME - REENTER LINE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST904A, MESSAGE ID-
TEXT='IST904A MAXIMUM OF 50 NODENAMES - ENTER ADDITIONAL OPTIONS', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST905A, MESSAGE ID-
TEXT='IST905A ENTER TRACE PRINT OPTIONS OR "CANCEL"', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST906A, MESSAGE ID-
TEXT='IST906A ENTER ADDITIONAL OPTIONS OR "CANCEL"', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST907A, MESSAGE ID-
TEXT='IST907A SNAPSHOT MODE TPRINT? ENTER Y OR N', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST908A, MESSAGE ID-
TEXT='IST908A OFFLINE MODE TPRINT? ENTER Y OR N', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST909A, MESSAGE ID-
TEXT='IST909A INVALID INTERVAL SPECIFIED - REENTER LINE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
USSMSG MSG=IST910I, MESSAGE ID-
TEXT='IST910I TRACE PRINT UTILITY ENDED',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST911I, MESSAGE ID-
TEXT='IST911I TPRINT CANCELLED - INVALID SYS004 ASSIGNMENT',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST912I, MESSAGE ID-
TEXT='IST912I TPRINT CANCELLED - VTAM TERMINATION IN PROGRESS',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST913I, MESSAGE ID-
TEXT='IST913I TPRINT CANCELLED - ALREADY IN PROGRESS',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST914I, MESSAGE ID-
TEXT='IST914I TPRINT EDITING OFFLINE FILE ON SYS004',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST915I, MESSAGE ID-
TEXT='IST915I TRACE RECORDING SUSPENDED FOR ONLINE TPRINT',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST916I, MESSAGE ID-
TEXT='IST916I SNAPSHOT TPRINT PROCEEDING - NO FILE PROVIDED',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST917I, MESSAGE ID-
TEXT='IST917I RECORDS MISSING ON TRACE FILE',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST919A, MESSAGE ID-
TEXT='IST919A INVALID CLEAR OPTION - REENTER LINE',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST919I, MESSAGE ID-
TEXT='IST919I NODE % NO LONGER HAS CONTROLLING LU %(%2)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST920I, MESSAGE ID-
TEXT='IST920I %(%2) %(%3) BUFF SIZE %(%4) EXP INCREMENT %(%5)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST921I, MESSAGE ID-
TEXT='IST921I TIMES EXP % EXP/CONT THRESH %(%2- %(%3)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NBLKSUP MESSAGE PROCESSING OPTIONS
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables 1193
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST965I, MESSAGE ID-
TEXT='IST965I AUTO DUMP/LOAD: %',
ROUTCDE=2, ROUTING CODE-
DESC=3, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST966I, MESSAGE ID-
TEXT='IST966I USER=VCNS',
ROUTCDE=(2,8), ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST967I, MESSAGE ID-
TEXT='IST967I % FAILED FOR %(%2); RC %(%3) RS %(%4)',
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST968I, MESSAGE ID-
TEXT='IST968I INTERFACE INITIALIZATION FAILED-REASON %',
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST970I, MESSAGE ID-
TEXT='IST970I LU-LU VERIFICATION ERROR % FOR %(%2)',
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST971I, MESSAGE ID-
TEXT='IST971I ADJ LINK STATION % USING LINE %(%2) IN %(%3)',
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST972I, MESSAGE ID-
TEXT='IST972I % TRACE FOR %(%2) TERMINATED %(%3)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST973I, MESSAGE ID-
TEXT='IST973I USRVAR % CHANGED FROM %(%2) TO %(%3)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST974I, MESSAGE ID-
TEXT='IST974I TPRINT CANCELLED - INSUFFICIENT STORAGE',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST976I, MESSAGE ID-
TEXT='IST976I ENTRY % DEFINED BUT NO %(%2) DEFINED FOR %(%3)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST977I, MESSAGE ID-
TEXT='IST977I MDLTAB=% ASLTAB=%(%2)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST979I, MESSAGE ID-
TEXT='IST979I BUILD FAILED FOR TABLE %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST981I, MESSAGE ID-
TEXT='IST981I VTAM PRIVATE: CURRENT = %, MAXIMUM USED = %(%2)',
ROUTCDE=2, ROUTING CODE-
Appendix A. IBM-supplied tables

1195
Appendix A. IBM-supplied tables

1197
USSMSG MSG=IST1024I, MESSAGE ID-
  TEXT='IST1024I I/O ERROR ON READ FOR CUA=%-BYTE COUNT MI-
  SMATCH', ROUTCDE=4, ROUTING CODE-
  DESC=2, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1025I, MESSAGE ID-
  TEXT='IST1025I CPS DEFINITION TABLE NAME=%', ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1026I, MESSAGE ID-
  TEXT='IST1026I SIGNAL=% DELAY=%(2) RETRY LIMIT=%(3)', ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1027I, MESSAGE ID-
  TEXT='IST1027I CPS DEFINITION TABLE NOT LOADED', ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1028I, MESSAGE ID-
  TEXT='IST1028I %', ROUTCDE=8, ROUTING CODE-
  DESC=4, DESCRIPTOR CODE-
  SUPP=SER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1029I, MESSAGE ID-
  TEXT='IST1029I % %(2) %(3) %(4) %(5) %(6)', ROUTCDE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1030I, MESSAGE ID-
  TEXT='IST1030I USERVAR EXIT IS %', ROUTCDE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1031I, MESSAGE ID-
  TEXT='IST1031I % COMMAND FAILED - %(2): EXIT FAILURE, CO-
  DE %(3)', ROUTCDE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1032I, MESSAGE ID-
  TEXT='IST1032I % BUFFER SIZE TOO SMALL-SIZE MUST BE AT L-
  EAST %(2)', ROUTCDE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=SER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1033I, MESSAGE ID-
  TEXT='IST1033I % ALREADY DEFINED FOR APPC SESSIONS OF %(-
  2)', ROUTCDE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1034I, MESSAGE ID-
  TEXT='IST1034I % ALREADY USING %(2) FOR APPC SESSIONS', ROUTCDE=8, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1035I, MESSAGE ID-
  TEXT='IST1035I ERROR WHILE % NETWORK %(2) %(3) WGN %(4)-
  ', ROUTCDE=8, ROUTING CODE-
  DESC=4, DESCRIPTOR CODE-
  SUPP=WARN, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1036I, MESSAGE ID-
  TEXT='IST1036I NODE ABEND-UNUSABLE RESOURCE IS %', ROUTCDE=8, ROUTING CODE-
Appendix A. IBM-supplied tables
USSMSG MSG=IST1050I, MESSAGE ID-
TEXT='IST1050I MAXIMUM COMPRESSION LEVEL - INPUT = %, OUTPUT = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1051I, MESSAGE ID-
TEXT='IST1051I EVENT CODE = %',
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1052I, MESSAGE ID-
TEXT='IST1052I SYNTAX ERROR AFTER % ALL FURTHER OPTIONS IGNORED',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1053I, MESSAGE ID-
TEXT='IST1053I VALUE FOR % MUST BE ''YES'' OR ''NO'',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1054I, MESSAGE ID-
TEXT='IST1054I VALUE FOR % MUST BE BETWEEN % AND %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1055I, MESSAGE ID-
TEXT='IST1055I % IS ALSO A REAL RESOURCE',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1056I, MESSAGE ID-
TEXT='IST1056I LUGROUP MAJOR NODE = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1057I, MESSAGE ID-
TEXT='IST1057I MODEL NAME = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1058I, MESSAGE ID-
TEXT='IST1058I MODEL LU GROUP = %, LUSEED = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1059I, MESSAGE ID-
TEXT='IST1059I LUGROUP MAJOR NODE = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1060I, MESSAGE ID-
TEXT='IST1060I EVENT ID = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1061I, MESSAGE ID-
TEXT='IST1061I FAILURE OCCURRED ON % AT %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1062I, MESSAGE ID-
TEXT='IST1062I EVENT ID = %',
ROUTCDE=2, ROUTING CODE-
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1149I, MESSAGE ID-
TEXT='IST1149I VARY % PROCESSING FOR NODE %(2) COMPLETE', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1150I, MESSAGE ID-
TEXT='IST1150I % CHANGED: %(2) TO %(3)', -
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1151I, MESSAGE ID-
TEXT='IST1151I USERVAR % DEFINED: VALUE = %(2)', -
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1152I, MESSAGE ID-
TEXT='IST1152I % CONTROLLING LU %(2) REMOVED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1153I, MESSAGE ID-
TEXT='IST1153I % (2) SESSION %(3) Buf Use %(4)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1154I, MESSAGE ID-
TEXT='IST1154I % (2) %(3)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1155I, MESSAGE ID-
TEXT='IST1155I % VARY NOLOGON - %(2) FAILED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1156I, MESSAGE ID-
TEXT='IST1156I USERVAR % IN %(2) HAS VALUE %(3)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1157I, MESSAGE ID-
TEXT='IST1157I DUPLICATE REGISTRATION % (2)', -
ROUTCDE=2, ROUTING CODE-
DESC=3, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1158I, MESSAGE ID-
TEXT='IST1158I MODIFY TOPO COMMAND FAILED, ID = %(2)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1159I, MESSAGE ID-
TEXT='IST1159I HOST NODE DATABASE ENTRY CANNOT BE DELETED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1160I, MESSAGE ID-
TEXT='IST1160I TYPE=MUST BE SPECIFIED FOR LOCAL TG- OR ADJACENT NODE', -
IST1183I % EXIT RETURNED A CODE OF %(2) %(3) %(4)

IST1184I CPNAME=%- NETSRVR = %(2)

IST1185I NAME=%- DIRECTORY ENTRY = %(2) %(3)

IST1186I DIRECTORY ENTRY = % %(2)

IST1187I % NOT VALID-APPN NOT SUPPORTED BY %(2)

IST1188I VTAM % STARTED AT %(2) ON %(3)

IST1189I % SESSION DEACTIVATION FAILURE FOR %(2)

IST1190I OPEN FAILED FOR % ABEND = %(2) RC = %(3)

IST1191I I/O ERROR ON % %(2)

IST1192I CLOSE FAILED FOR % ABEND = %(2) RC = %(3)

IST1193I DUPLICATE RESOURCE IS %

IST1194I APPN CONNECTION FOR % INACTIVE - TG = %(-2)
Appendix A. IBM-supplied tables
USSMSG MSG=IST1209I, MESSAGE ID-
TEXT='IST1209I PROGRAM-TO-PROGRAM INTERFACE MODULE STORAGE SHORTAGE',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1211I, MESSAGE ID-
TEXT='IST1211I I/O ERROR % (2) (3) (4)',
ROUTCODE=(2,8,10), ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1212I, MESSAGE ID-
TEXT='IST1212I % = (2) STATUS = (3)',
ROUTCODE=(2,8), ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1213I, MESSAGE ID-
TEXT='IST1213I % LU-LU VERIFY ERROR (2)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1214I, MESSAGE ID-
TEXT='IST1214I FFST %',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1215I, MESSAGE ID-
TEXT='IST1215I ERROR IN START LIST % - (2)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1216A, MESSAGE ID-
TEXT='IST1216A ENTER 1 TO CONTINUE - 2 TO REENTER LIST - 3 TO TERMINATE VTAM',
ROUTCODE=1, ROUTING CODE-
DESC=2, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1217A, MESSAGE ID-
TEXT='IST1217A RESPONSE NOT VALID: REENTER 1, 2, OR 3',
ROUTCODE=1, ROUTING CODE-
DESC=2, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1218I, MESSAGE ID-
TEXT='IST1218I ACB ERROR FIELD = %',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1219I, MESSAGE ID-
TEXT='IST1219I RTNCD = %, FDB2 = (2)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1220I, MESSAGE ID-
TEXT='IST1220I SSCPNAME NETID CURRENT STATE ROUTING STATUS',
ROUTCODE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1221I, MESSAGE ID-
TEXT='IST1221I % DEV = (2) STATUS = (3) STATE = (4)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1222I, MESSAGE ID-

1212 z/OS V2R1.0 Communications Server: SNA Resource Definition Reference
USSMSG MSG=IST1247I, TEXT='IST1247I ALL ATTEMPTS TO ESTABLISH A SESSION WERE UNSUCCESSFUL', ROUTCDE=2, DESC=5, SUPP=SER, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1248I, TEXT='IST1248I DEACTIVATE LOCAL LINK BEFORE DELETING', ROUTCDE=2, DESC=5, SUPP=WARN, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1249I, TEXT='IST1249I SYNTAX ERROR AT RECORD % IN MEMBER %', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1250I, TEXT='IST1250I NAME LEVEL MODULE STATUS', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1251I, TEXT='IST1251I % %(2) %(3) %(4)', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1252I, TEXT='IST1252I DEFINED NETWORK NODE SERVER LIST, NAME = %', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1253I, TEXT='IST1253I % %(2) %(3)', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1254I, TEXT='IST1254I SERVER LIST PROCESSED ORDER = %', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1255I, TEXT='IST1255I OTHER NETWORK NODES ALLOWED AS SERVERS', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1256I, TEXT='IST1256I CURRENT NETWORK NODE SERVER', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1257I, TEXT='IST1257I SEQUENCE NOT VALID, STATEMENT IGNORED, SKIPPING TO EOF', ROUTCDE=2, DESC=5, SUPP=WARN, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1258I, TEXT='IST1258I % IS NOT VALID FOR %', ROUTCDE=2, DESC=5, SUPP=WARN, OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1259I, TEXT='IST1259I VBUILD TYPE=% IS ONLY VALID FOR %', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1260I, MESSAGE ID-
TEXT='IST1260I % TRUNCATED - INSUFFICIENT STORAGE',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1261I, MESSAGE ID-
TEXT='IST1261I ABEND OCCURRED DURING LINK DEFINITION',
ROUTCODE=2, ROUTING CODE-
DESC=3, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1262I, MESSAGE ID-
TEXT='IST1262I MODULE % LOAD FAILED - %(2)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1263I, MESSAGE ID-
TEXT='IST1263I % FOR %(2) FORCED COLD, %(3)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1264I, MESSAGE ID-
TEXT='IST1264I % FOR %(2) FAILED DURING DEFINITION',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1265I, MESSAGE ID-
TEXT='IST1265I % FOR %(2) FAILED - %(3)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1266I, MESSAGE ID-
TEXT='IST1266I % DEACTIVATION %(2) FAILED: %(3)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1267I, MESSAGE ID-
TEXT='IST1267I % FOR %(2) FAILED',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1268I, MESSAGE ID-
TEXT='IST1268I % FOR %(2) SCHEDULED BY %(3)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1269I, MESSAGE ID-
TEXT='IST1269I % FOR %(2) AFFECTS NEW SESSIONS ONLY',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1270I, MESSAGE ID-
TEXT='IST1270I % FOR %(2) FAILED - %(3)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS-
MSGLVL=START ISSUE OLD MESSAGE
USSMSG MSG=IST1271I, MESSAGE ID-
TEXT='IST1271I % FOR %(2) SCHEDULED BY %(3)',
ROUTCODE=2, ROUTING CODE-
Appendix A. IBM-supplied tables

1219
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables 1223
USSMSG MSG=IST1377I, MESSAGE ID-
TEXT='IST1377I LIST ENTRIES % %(2) %(3)', -
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1378I, MESSAGE ID-
TEXT='IST1378I % FAILED FOR %(2) - GENERIC RESOURCE NAME- EXISTS',
ROUTCODE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1380I, MESSAGE ID-
TEXT='IST1380I DISCONNECTING FROM STRUCTURE %',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1381I, MESSAGE ID-
TEXT='IST1381I REBUILD STARTED FOR STRUCTURE %',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1382I, MESSAGE ID-
TEXT='IST1382I REBUILD HAS BEEN STOPPED FOR STRUCTURE %',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1383I, MESSAGE ID-
TEXT='IST1383I REBUILD COMPLETE FOR STRUCTURE %',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1385I, MESSAGE ID-
TEXT='IST1385I ADJCLUST INFORMATION WAS IGNORED DUE TO insufficient STORAGE',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1386I, MESSAGE ID-
TEXT='IST1386I DYNAMIC ALLOCATION FAILED FOR % CODE = % (2) REASON = %%(3)',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1391I, MESSAGE ID-
TEXT='IST1391I DELAYED DISCONNECT OF % FAILED DUE TO ABEND',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1392I, MESSAGE ID-
TEXT='IST1392I DISCNTIM = % DEFINED AT %(2) FOR DISCONNECT',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1393I, MESSAGE ID-
TEXT='IST1393I GENERIC RESOURCE NAME RESOLUTION EXIT IS - %',
ROUTCODE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1394I, MESSAGE ID-
TEXT='IST1394I CPNAME = % STATION ID = %%(2)',
ROUTCODE=6, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1395I, MESSAGE ID-
Appendix A. IBM-supplied tables
TEXT='IST1395I FLDTAB = % FILTER = %d(2)',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=NEVER, SUPPRESSION CODE=
 OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1396I, MESSAGE ID=
 TEXT='IST1396I DISK I/O INITIALIZATION FAILED FOR CMIP SERVICES',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=SER, SUPPRESSION CODE=
 OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1397I, MESSAGE ID=
 TEXT='IST1397I INITIALIZATION FAILED FOR CMIP SERVICES',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=SER, SUPPRESSION CODE=
 OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1398I, MESSAGE ID=
 TEXT='IST1398I ALL ATTEMPTS TO RESTART CMIP SERVICES WERE UNSUCCESSFUL',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=SER, SUPPRESSION CODE=
 OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1399I, MESSAGE ID=
 TEXT='IST1399I ATTEMPTING TO RESTART CMIP SERVICES',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=SER, SUPPRESSION CODE=
 OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1400I, MESSAGE ID=
 TEXT='IST1400I DGTIMER = % EXTIMER = %d(2)',
 ROUTCDE=8, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=NEVER, SUPPRESSION CODE=
 OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1401I, MESSAGE ID=
 TEXT='IST1401I RESOURCE NOT FOUND-RETRY IN % SEC(S) OR % REQUEST(S)',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=NEVER, SUPPRESSION CODE=
 OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1402I, MESSAGE ID=
 TEXT='IST1402I SRTIMER = % SRCOUNT = %d(2)',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=NEVER, SUPPRESSION CODE=
 OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1403I, MESSAGE ID=
 TEXT='IST1403I MODIFY QUERY REPLY FROM %',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=NEVER, SUPPRESSION CODE=
 OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1404I, MESSAGE ID=
 TEXT='IST1404I % %d(2)',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=NEVER, SUPPRESSION CODE=
 OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1405I, MESSAGE ID=
 TEXT='IST1405I %',
 ROUTCDE=2, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=NEVER, SUPPRESSION CODE=
 OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1406I, MESSAGE ID=
 TEXT='IST1406I CONTIMER = % IATIMER = %d(2)',
 ROUTCDE=8, ROUTING CODE=
 DESC=5, DESCRIPTOR CODE=
 SUPP=NEVER, SUPPRESSION CODE=
 OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1407I, MESSAGE ID=
 TEXT='IST1407I % IS A MULTIPLE INSTANCE OF EXIT %d(2)',
 ROUTCDE=8, ROUTING CODE=
 DESC=4, DESCRIPTOR CODE=
 SUPP=NEVER, SUPPRESSION CODE=
USSMSG MSG=IST1408I, MESSAGE ID-
TEXT='IST1408I MODIFY TGP NOT APPLICABLE FOR % %(2)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1409I, MESSAGE ID-
TEXT='IST1409I % ASSOC = %(2) ETYPE = %(3)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1410I, MESSAGE ID-
TEXT='IST1410I QUERY % %(2)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1411I, MESSAGE ID-
TEXT='IST1411I INOP GENERATED FOR %',
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1412I, MESSAGE ID-
TEXT='IST1412I % %(2) - RETURN CODE %(3)',
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1413I, MESSAGE ID-
TEXT='IST1413I % - REDIAL ATTEMPTED FOR %(2)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1414I, MESSAGE ID-
TEXT='IST1414I % - REDIAL NOT ATTEMPTED FOR %(2)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1415I, MESSAGE ID-
TEXT='IST1415I % CONFLICTS WITH A GENERIC RESOURCE NAME',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1416I, MESSAGE ID-
TEXT='IST1416I ID = % FAILED - RECOVERY IN PROGRESS',
ROUTCDE=(2,8), ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1417I, MESSAGE ID-
TEXT='IST1417I NETID NAME STATUS TYPE -'
MALNODE'
ROUTCDE=2, ROUTING CODE-
DESC=3, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1418I, MESSAGE ID-
TEXT='IST1418I % %(2) %(3) %(4) %(5)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1419I, MESSAGE ID-
TEXT='IST1419I DUPLICATE SESSION INFORMATION REPORTED FOR %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1420I, MESSAGE ID-
TEXT='IST1420I UNABLE TO ASSOCIATE THE FOLLOWING SESSION-%(5) WITH %',

### USSMSG MSG=ISTI4611, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>NOBLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4611 % %(2) %(3) %(4)
```

### USSMSG MSG=ISTI4621, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>NOBLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4621 ECHO IS %
```

### USSMSG MSG=ISTI4631, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4631 ALLOCATION DURATION: % MILLISECONDS
```

### USSMSG MSG=ISTI4641, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4641 PROGRAM STARTUP AND VERSION EXCHANGE: % MILLISECONDS
```

### USSMSG MSG=ISTI4651, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4651 DURATION DATA SENT DATA RATE
```

### USSMSG MSG=ISTI4661, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4661 (MILLISECONDS) (BYTES) (KBYTE/SECOND) (MBIT/SECOND)
```

### USSMSG MSG=ISTI4671, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4671 % %(2) %(3) %(4)
```

### USSMSG MSG=ISTI4681, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4681 TOTALS: % %(2) %(3) %(4)
```

### USSMSG MSG=ISTI4691, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4691 DURATION STATISTICS:
```

### USSMSG MSG=ISTI4701, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4701 MINIMUM = % AVERAGE = % (2) MAXIMUM = % (3)
```

### USSMSG MSG=ISTI4711, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4711 SESSION UNAVAILABLE FOR APIING
```

### USSMSG MSG=ISTI4721, MESSAGE ID-
<table>
<thead>
<tr>
<th>ROUTCDE</th>
<th>DESC</th>
<th>SUPP</th>
<th>OPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>NEVER</td>
<td>BLKSUP</td>
</tr>
</tbody>
</table>

#### Text:
```
ISTI4721 APIING % ERROR
```

---

Appendix A. IBM-supplied tables
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1473I,  MESSAGE ID-
TEXT='IST1473I SENSE = %',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1474I,  MESSAGE ID-
TEXT='IST1474I TP CONCURRENT INSTANCE LIMIT = %',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1475I,  MESSAGE ID-
TEXT='IST1475I EXIT % IN USE: REENTER COMMAND TO FORCE',
ROUTCDE=2,  ROUTING CODE-
DESC=3,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1476I,  MESSAGE ID-
TEXT='IST1476I TCID X''%'' - REMOTE TCID X''%(2)'',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1477I,  MESSAGE ID-
TEXT='IST1477I ALLOWED DATA FLOW RATE = %',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1478I,  MESSAGE ID-
TEXT='IST1478I NUMBER OF UNACKNOWLEDGED BUFFERS = %',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1479I,  MESSAGE ID-
TEXT='IST1479I RTP CONNECTION STATE=% - MNPS = %',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1480I,  MESSAGE ID-
TEXT='IST1480I RTP END TO END ROUTE - RSCV PATH',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1481I,  MESSAGE ID-
TEXT='IST1481I DESTINATION CP % - NCE X''%(2)'',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1482I,  MESSAGE ID-
TEXT='IST1482I HPR = % - OVERRIDE = %',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=BLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1483I,  MESSAGE ID-
TEXT='IST1483I DTEAD = %',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=NOBLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1484I,  MESSAGE ID-
TEXT='IST1484I DLCADDR SUBFIELDS FOR %',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
OPT=NOBLKSUP  MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1485I,  MESSAGE ID-
TEXT='IST1485I DLCADDR SUBFIELDS FOR %',
ROUTCDE=2,  ROUTING CODE-
DESC=5,  DESCRIPTOR CODE-
SUPP=NEVER,  SUPPRESSION CODE-
USSMSG MSG=IST1501I, MESSAGE ID-
TEXT='IST1501I XCF TOKEN = %',
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1502I, MESSAGE ID-
TEXT='IST1502I ADJACENT CP = %',
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1503I, MESSAGE ID-
TEXT='IST1503I XCF TOKEN = %, STATUS = %(2)',
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1504I, MESSAGE ID-
TEXT='IST1504I XCF CONNECTION WITH % IS INOPERATIVE',
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1505I, MESSAGE ID-
TEXT='IST1505I TYPE = %, TOKEN = %(2)',
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1506I, MESSAGE ID-
TEXT='IST1506I % FAILED FOR %(2) - MEMBER LEAVING GROUP',
ROUTCDE=(2,8), ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1507I, MESSAGE ID-
TEXT='IST1507I VR-BASED TG NOT SUPPORTED',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1508I, MESSAGE ID-
TEXT='IST1508I CP-CP SESSIONS ON VR-BASED TG NOT SUPPORTED',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1509I, MESSAGE ID-
TEXT='IST1509I % UNKNOWN BUT ACCEPTED -- PREVIOUS VALUE WAS %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1510I, MESSAGE ID-
TEXT='IST1510I LLERP= % - RECEIVED = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1511I, MESSAGE ID-
TEXT='IST1511I MAXIMUM NETWORK LAYE PACKET SIZE = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1512I, MESSAGE ID-
TEXT='IST1512I % FAILED - CODE X''%(2)'' - CUA %(3)',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1513I, MESSAGE ID-
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables

1241
USSMSG MSG=IST1602I, MESSAGE ID-
  TEXT='IST1602I RU ERROR: EXTRA CV X''%''', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1603I, MESSAGE ID-
  TEXT='IST1603I RU ERROR: INVALID POSITIVE RESPONSE', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1604I, MESSAGE ID-
  TEXT='IST1604I RU ERROR: LENGTH, FORMAT, OR TYPE NOT VAL-
  ID', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1605I, MESSAGE ID-
  TEXT='IST1605I RU ERROR: MISSING CV X''0B''', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1606I, MESSAGE ID-
  TEXT='IST1606I DIAL RETRY FAILED', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1607I, MESSAGE ID-
  TEXT='IST1607I RU ERROR: RESPONSE TOO LONG', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1608I, MESSAGE ID-
  TEXT='IST1608I RU ERROR: RESPONSE TOO SHORT', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1609I, MESSAGE ID-
  TEXT='IST1609I CV X''0B'' INDICATES ADJACENT LINK STATION-
  N NOT SUPPORTED', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1610I, MESSAGE ID-
  TEXT='IST1610I CORRELATOR MISMATCH - LOAD=NO', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1611I, MESSAGE ID-
  TEXT='IST1611I CORRELATOR MISMATCH - NCP ACQUIRED BEFORE-
  ACTIVATION', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1612I, MESSAGE ID-
  TEXT='IST1612I LOAD MODULE MISMATCH - EXPECTED % FOUND %-
(2)', 
  ROUTCDE=2, ROUTING CODE=
  DESC=5, DESCRIPTOR CODE=
  SUPP=SER, SUPPRESSION CODE=
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1613I, MESSAGE ID-
  TEXT='IST1613I TYPE = % ATTN = %\%(2)', 
  ROUTCDE=2, ROUTING CODE=
  DESC=4, DESCRIPTOR CODE=
  SUPP=NEVER, SUPPRESSION CODE=
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1614I, MESSAGE ID-
  TEXT='IST1614I READ510 = % PACKET = %\%(2) BYT = %\%(3)', 
  ROUTCDE=2, ROUTING CODE=
Appendix A. IBM-supplied tables
CONGEST STALL SESS',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1696I, MESSAGE ID-
TEXT='IST1696I % (2) (3) (4) (5) (6)'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1697I, MESSAGE ID-
TEXT='IST1697I RTP PACING ALGORITHM = ARB RESPONSIVE MODE'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1698I, MESSAGE ID-
TEXT='IST1698I Probe % attempted - FFST not available'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1700I, MESSAGE ID-
TEXT='IST1700I % conflicts with % (2) % (3)'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1701I, MESSAGE ID-
TEXT='IST1701I CP name locate size'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1702I, MESSAGE ID-
TEXT='IST1702I % % (2)'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1703I, MESSAGE ID-
TEXT='IST1703I desired locate size = % last locate size - % (2)'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1704I, MESSAGE ID-
TEXT='IST1704I % % from adjacent sscp table'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1705I, MESSAGE ID-
TEXT='IST1705I % % from start option'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1706I, MESSAGE ID-
TEXT='IST1706I partner name generic resource member attributes'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1707I, MESSAGE ID-
TEXT='IST1707I % % (2) % (3) % (4)'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1708I, MESSAGE ID-
TEXT='IST1708I % failed - generic resources not available'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
Appendix A. IBM-supplied tables
TEXT='IST1734I DEVICE % INITIATED SLOWDOWN TERMINATION', -
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=NORM, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1736I, MESSAGE ID-
TEXT='IST1736I PU NAME', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1737I, MESSAGE ID-
TEXT='IST1737I %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1738I, MESSAGE ID-
TEXT='IST1738I ANR LABEL TP NUMBER', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1739I, MESSAGE ID-
TEXT='IST1739I % % (2) % (3)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1740I, MESSAGE ID-
TEXT='IST1740I IBMTGPS IN % IS EMPTY - PROCESSING CONTINUES', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1743I, MESSAGE ID-
TEXT='IST1743I IBMTGPS IN % NOT FOUND - PROCESSING CONTINUES', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1744I, MESSAGE ID-
TEXT='IST1744I IBMTGPS IN % IN ERROR - PROCESSING CONTINUES', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1745I, MESSAGE ID-
TEXT='IST1745I % REJECTED BECAUSE VARYWLD = % (2)', -
ROUTCDE=5, ROUTING CODE-
DESC=2, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1746I, MESSAGE ID-
TEXT='IST1746I \% COMMAND PROCESSING COMPLETE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1747I, MESSAGE ID-
TEXT='IST1747I SUMMARY OF STATE INFORMATION;', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1748I, MESSAGE ID-
TEXT='IST1748I % \% (2) \% (3)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1749I, MESSAGE ID-
TEXT='IST1749I % REJECTED - VARYWLD = NO FOR APPL % (2)', -
ROUTCDE=5, ROUTING CODE-
DESC=2, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7501, MESSAGE ID-
TEXT='IST1750I PCITHRSH = % PCITHRSH = %(2)', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7511, MESSAGE ID-
TEXT='IST1751I PCUNPRD = % PCUNPRD = %(2)', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7521, MESSAGE ID-
TEXT='IST1752I RPROCDEF = % RPROCDEF = %(2)', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7531, MESSAGE ID-
TEXT='IST1753I SBALMAX = % SBALMAX = %(2)', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7541, MESSAGE ID-
TEXT='IST1754I QDPTHMAX = % QDPTHMAX = %(2)', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7551, MESSAGE ID-
TEXT='IST1755I RTP FROM THE DIRECTION OF THE DLUR', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7561, MESSAGE ID-
TEXT='IST1756I % SUBCHANNEL %(2) VARY OFFLINE NOT ALLOWED', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7571, MESSAGE ID-
TEXT='IST1757I % PARAMETER EXTRANEOUS FOR DISPLAY ID COMMAND', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7581, MESSAGE ID-
TEXT='IST1758I RSCV TOWARDS DLUR SAVED AT SESSION ACTIVATION', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7591, MESSAGE ID-
TEXT='IST1759I RTP FROM THE DIRECTION OF THE DLUR', ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=ISTI7601, MESSAGE ID-
TEXT='IST1760I % ACT FAILED, TCPNAME OR IPADDR START OPT-
Appendix A. IBM-supplied tables 1253
USSMSG MSG=IST1790I, MESSAGE ID-
TEXT='IST1790I % (2) (3) (4)', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1791I, MESSAGE ID-
TEXT='IST1791I HPR ROUTE TEST PACKET NOT RETURNED BY NODE %', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1792I, MESSAGE ID-
TEXT='IST1792I TOTAL RTP TRAVERSAL TIME % MILLISECONDS', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1793I, MESSAGE ID-
TEXT='IST1793I HPR ROUTE TEST NOT INITIATED - RTP PU NOT IN PROPER STATE', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1794I, MESSAGE ID-
TEXT='IST1794I HPR ROUTE TEST NOT INITIATED - TEST ALREADY IN PROGRESS', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1795I, MESSAGE ID-
TEXT='IST1795I HPR ROUTE TEST NOT INITIATED - INSUFFICIENT STORAGE', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1796I, MESSAGE ID-
TEXT='IST1796I SYSTEM-MANAGED DUPLEXING REBUILD IS IN PROGRESS', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1797I, MESSAGE ID-
TEXT='IST1797I STRUCTURE TYPE = %', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1798I, MESSAGE ID-
TEXT='IST1798I TOPOLOGY DATASET RETRIEVED WAS CREATED ON % (2)', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1799I, MESSAGE ID-
TEXT='IST1799I NO ROUTE AVAILABLE TO DESTINATION IP ADDRESS %', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=SER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1800I, MESSAGE ID-
TEXT='IST1800I TRLE = % ** CONGESTED **', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1801I, MESSAGE ID-
TEXT='IST1801I UNITS OF WORK FOR NCB AT ADDRESS X''%', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=NEVER, SUPPRESSION CODE=OPT-BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1802I, MESSAGE ID-
TEXT='IST1802I % CURRENT = %(2), AVERAGE = %(3), MAXIMUM = %(4)', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=NEVER, SUPPRESSION CODE=OPT-BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1803I, MESSAGE ID-
TEXT='IST1803I % PARAMETER VALUE NOT VALID - DEFAULT %(2) USED', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=INFO, SUPPRESSION CODE=OPT-BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1804I, MESSAGE ID-
TEXT='IST1804I % PARAMETER NOT VALID - IGNORED', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=INFO, SUPPRESSION CODE=OPT-BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1805I, MESSAGE ID-
TEXT='IST1805I ONLY LOCAL TOPOLOGY INFORMATION IS AVAILABLE', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=INFO, SUPPRESSION CODE=OPT-NOBLSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1806I, MESSAGE ID-
TEXT='IST1806I NO DIAL-IN LINE FOUND FOR HPR/IP SWITCHED-CONNECTION', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=NEVER, SUPPRESSION CODE=OPT-BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1807I, MESSAGE ID-
TEXT='IST1807I NO DIAL-IN LINE FOR HPR/IP VIRTUAL NODE %', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=NEVER, SUPPRESSION CODE=OPT-BLKSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1809I, MESSAGE ID-
TEXT='IST1809I HPR ROUTE TEST NOT INITIATED - INSUFFICIENT PATH INFORMATION', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=INFO, SUPPRESSION CODE=OPT-NOBLSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1810I, MESSAGE ID-
TEXT='IST1810I PKTIQDO = % PKTIQD = %(2)', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=NEVER, SUPPRESSION CODE=OPT-NOBLSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1811I, MESSAGE ID-
TEXT='IST1811I BYTIQDO = % BYTIQD = %(2)', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=NEVER, SUPPRESSION CODE=OPT-NOBLSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1816I, MESSAGE ID-
TEXT='IST1816I PARAMETER 1 FOR DISCNTIM MUST BE NUMERIC - OR IMMEDIATE', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=NEVER, SUPPRESSION CODE=OPT-NOBLSUP, MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1817I, MESSAGE ID-
TEXT='IST1817I PATH SWITCH REASON: RTP CONNECTION UNAVAILABLE', ROUTCODE=2, ROUTING CODE=5, DESCRIPTOR CODE=5, SUPP=NEVER, SUPPRESSION CODE=OPT-NOBLSUP, MESSAGE PROCESSING OPTIONS

Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables

**IST1855I**
NUMBER OF SESSIONS USING RTP = %

**IST1856I**
LAST PATH SWITCH OCCURRENCE WAS ON % AT %

**IST1857I**
BACKPRESSURE REASON COUNTS:

**IST1858I**
PATHSWITCH SEND QUEUE MAX STORAGE FAILU-
RE STALLED PIPE:

**IST1859I**
% %(2) %(3) - %(4)

**IST1860I**
NUMBER OF NLPS SENT=%- OVERFLOW = %(2)-

**IST1861I**
NUMBER OF NLPS RECEIVED=%- OVERFLOW = -

**IST1862I**
ARB MAXIMUM SEND RATE = % %(2)

**IST1863I**
SUBAREA INDEX E-
LEMENT:

**IST1864I**
X''%'' (%(2)) X''%(3)'' (%(4)) X''-
'%'(5)'' (%(6))'.'

**IST1865I**
GLOBAL INOPDUMP = %

**IST1866I**
TRLE = % INOPDUMP = %(2)'}
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables 1267
TEXT='IST1943I DIRECTED SEARCH TO A SERVED END NODE', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1944I, MESSAGE ID-
TEXT='IST1944I DIRECTED SEARCH TO A NETWORK NODE', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1945I, MESSAGE ID-
TEXT='IST1945I DIRECTED SEARCH TO A BORDER NODE', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1946I, MESSAGE ID-
TEXT='IST1946I LOCAL SUBAREA SEARCH', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1947I, MESSAGE ID-
TEXT='IST1947I BROADCAST SEARCH TO SERVED END NODES', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1948I, MESSAGE ID-
TEXT='IST1948I DIRECTED SEARCHES TO BORDER NODES', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1949I, MESSAGE ID-
TEXT='IST1949I DIRECTED SEARCH TO A CENTRAL DIRECTORY SERVER', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1950I, MESSAGE ID-
TEXT='IST1950I DIRECTED SEARCHES TO ALTERNATE CENTRAL DIRECTORY SERVERS', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1951I, MESSAGE ID-
TEXT='IST1951I BROADCAST SEARCH TO NETWORK NODES', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1952I, MESSAGE ID-
TEXT='IST1952I DIRECTED SEARCHES TO INTERCHANGE NODES', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1953I, MESSAGE ID-
TEXT='IST1953I % - SENSE ½(2) FROM ¾(3)', -
ROUTCDE=8, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1954I, MESSAGE ID-
TEXT='IST1954I TRL MAJOR NODE = ¾', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST1955I, MESSAGE ID-
TEXT='IST1955I STALL DETECTED FOR RTP ½ TO ¾(2)', -
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP=WARN, SUPPRESSION CODE-
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS
TEXT='IST1969I MAXIMUM ACTUAL DATA FLOW RATE = % %(2)', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1970I, MESSAGE ID=TEXT='IST1970I RATE REDUCTIONS DUE TO RETRANSMISSIONS = - %', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1971I, MESSAGE ID=TEXT='IST1971I TIMER INFORMATION:', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1972I, MESSAGE ID=TEXT='IST1972I SHORT REQUEST TIMER = % MILLISECONDS', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1973I, MESSAGE ID=TEXT='IST1973I OUTBOUND TRANSMISSION INFORMATION:', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1974I, MESSAGE ID=TEXT='IST1974I NUMBER OF NLPS SENT= % ( %(2) )', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1975I, MESSAGE ID=TEXT='IST1975I TOTAL BYTES SENT=%( %(2) )', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1976I, MESSAGE ID=TEXT='IST1976I BYTES RETRANSMITTED=%( %(2) )', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1977I, MESSAGE ID=TEXT='IST1977I MAXIMUM NUMBER OF NLPS ON WAITING-FOR-ACK-QUEUE = %', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1978I, MESSAGE ID=TEXT='IST1978I WAITING-FOR-ACK QUEUE MAX REACHED ON % AT-%(2)', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1979I, MESSAGE ID=TEXT='IST1979I INBOUND TRANSMISSION INFORMATION:', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1980I, MESSAGE ID=TEXT='IST1980I SEQUENCE NUMBER = % (X''%(2)'' )', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1981I, MESSAGE ID=TEXT='IST1981I TOTAL BYTES RECEIVED=%( %(2) )', ROUTCDE=2, ROUTING CODE= DESC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19821, MESSAGE ID-
  TEXT='IST1982I NUMBER OF NLPS ON INBOUND WORK QUEUE = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19831, MESSAGE ID-
  TEXT='IST1983I MAXIMUM NUMBER OF NLPS ON INBOUND WORK QUEUE = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19841, MESSAGE ID-
  TEXT='IST1984I PATH SWITCH INFORMATION:',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19851, MESSAGE ID-
  TEXT='IST1985I PATH SWITCHES INITIATED FROM REMOTE RTP = - %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19861, MESSAGE ID-
  TEXT='IST1986I PATH SWITCHES INITIATED FROM LOCAL RTP = - %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19871, MESSAGE ID-
  TEXT='IST1987I PATH SWITCHES DUE TO LOCAL FAILURE = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19881, MESSAGE ID-
  TEXT='IST1988I PATH SWITCHES DUE TO LOCAL PSRETRY = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19891, MESSAGE ID-
  TEXT='IST1989I NO MATCHING AUTOLOGIN REQUESTS',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19901, MESSAGE ID-
  TEXT='IST1990I PENDING AUTOLOGIN REQUESTS FOR:',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19911, MESSAGE ID-
  TEXT='IST1991I % - WAITING FOR PLU NOTIFICATION',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19921, MESSAGE ID-
  TEXT='IST1992I % - WAITING FOR AUTOTI TIMER',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19931, MESSAGE ID-
  TEXT='IST1993I % - WAITING FOR CDRM ACTIVATION',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST19941, MESSAGE ID-
  TEXT='IST1994I % - WAITING FOR CDRM OR CPCP ACTIVATION',
  ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1995I, MESSAGE ID-
TEXT='IST1995I % - WAITING FOR AUTOCAP NODE ACTIVATION', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1996I, MESSAGE ID-
TEXT='IST1996I OR AUTOTI TIMER', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1997I, MESSAGE ID-
TEXT='IST1997I % (2) % (3)' -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1998I, MESSAGE ID-
TEXT='IST1998I % - WAITING FOR AUTOTI/AUTORTRY START OPTION', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST1999I, MESSAGE ID-
TEXT='IST1999I MODIFICATION OR SLU REACTIVATION', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2000I, MESSAGE ID-
TEXT='IST2000I ENTERPRISE EXTENDER GENERAL INFORMATION', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2001I, MESSAGE ID-
TEXT='IST2001I ENTERPRISE EXTENDER CONNECTION INFORMATION', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2002I, MESSAGE ID-
TEXT='IST2002I ENTERPRISE EXTENDER XCA MAJOR NODE NAME == -
% ', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2003I, MESSAGE ID-
TEXT='IST2003I LIVETIME = % SRQTIME = % (2) SRQRETRY = % (3)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2004I, MESSAGE ID-
TEXT='IST2004I XCAP = % XCAP = % (2) XCAP = % (3)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2005I, MESSAGE ID-
TEXT='IST2005I IPRESOLV = % ', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2006I, MESSAGE ID-
TEXT='IST2006I PORT PRIORITY = SIGNAL NETWORK HIGH MEDIUM LOW', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2007I, MESSAGE ID-
TEXT='IST2007I IP PORT NUMBER = % (2) % (3) - % (4) % (5),'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2008I, MESSAGE ID-
TEXT='IST2008I IPTOS VALUE = % (2) - % (3) % (4) % (5),'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2009I, MESSAGE ID-
TEXT='IST2009I RTP PIPES = % LU-LU SESSIONS = - % (2),'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2010I, MESSAGE ID-
TEXT='IST2010I INOPS DUE TO SRQRETRY EXPIRATION = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2011I, MESSAGE ID-
TEXT='IST2011I AVAILABLE LINES FOR THIS EE VRN = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2012I, MESSAGE ID-
TEXT='IST2012I ACTIVE CONNECTIONS USING THIS EE-VRN = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2013I, MESSAGE ID-
TEXT='IST2013I AVAILABLE LINES FOR PREDEFINED EE CONNECT-
IONS = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2014I, MESSAGE ID-
TEXT='IST2014I ACTIVE PREDEFINED EE CONNECTIONS = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2015I, MESSAGE ID-
TEXT='IST2015I ACTIVE LOCAL VRN EE CONNECTIONS = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2016I, MESSAGE ID-
TEXT='IST2016I ACTIVE GLOBAL VRN EE CONNECTIONS = %',
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2017I, MESSAGE ID-
TEXT='IST2017I TOTAL RTP PIPES = % LU-LU SESSIONS = - % (2),'
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

Appendix A. IBM-supplied tables 1273
Appendix A. IBM-supplied tables
USSMSG MSG=IST20571, MESSAGE ID- 
TEXT='IST20571 UNREACHABLE PARTNER INFORMATION:', - 
ROUTCDE=2, ROUTING CODE- 
DESC=5, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20581, MESSAGE ID- 
TEXT='IST20581 UNREACHABLE PARTNER LIMIT EXCEEDED FOR VR-N %', - 
ROUTCDE=2, ROUTING CODE- 
DESC=3, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20591, MESSAGE ID- 
TEXT='IST20591 NUMBER OF NLPS RECEIVED = % \((\%\% )^{2}\)', - 
ROUTCDE=2, ROUTING CODE- 
DESC=5, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20611, MESSAGE ID- 
TEXT='IST20611 NO FORCED TAKEOVER REQUESTS ARE ACCEPTABLE', - 
ROUTCDE=2, ROUTING CODE- 
DESC=4, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20621, MESSAGE ID- 
TEXT='IST20621 \% FORCED TAKEOVER REQUESTS ARE ACCEPTABLE', - 
ROUTCDE=2, ROUTING CODE- 
DESC=4, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20631, MESSAGE ID- 
TEXT='IST20631 ALL FORCED TAKEOVER REQUESTS ARE ACCEPTABLE', - 
ROUTCDE=2, ROUTING CODE- 
DESC=4, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20641, MESSAGE ID- 
TEXT='IST20641 PLU TO SLU RU SIZE = % SLU TO PLU RU SIZE = %', - 
ROUTCDE=2, ROUTING CODE- 
DESC=5, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20651, MESSAGE ID- 
TEXT='IST20651 ENTERPRISE EXTENDER CONNECTION REXMIT INFORMATION', - 
ROUTCDE=2, ROUTING CODE- 
DESC=5, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20661, MESSAGE ID- 
TEXT='IST20661 ENTERPRISE EXTENDER CONNECTION SRQ RETRY INFORMATION', - 
ROUTCDE=2, ROUTING CODE- 
DESC=5, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20671, MESSAGE ID- 
TEXT='IST20671 EEDIAG DISPLAY ISSUED ON % AT \%(\%\% )^{2}\)', - 
ROUTCDE=2, ROUTING CODE- 
DESC=5, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20681, MESSAGE ID- 
TEXT='IST20681 NLP RETRANSMIT RATE = \%(\%\% )^{2}\)', - 
ROUTCDE=2, ROUTING CODE- 
DESC=5, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST20691, MESSAGE ID- 
TEXT='IST20691 REXMIT COUNTERS LAST CLEARED ON % AT \%(\%\% )^{2}\)', - 
ROUTCDE=2, ROUTING CODE- 
DESC=5, DESCRIPTOR CODE- 
SUPP=NEVER, SUPPRESSION CODE- 
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

Appendix A. IBM-supplied tables 1277
USSMSG MSG=IST2070I, MESSAGE ID-
  TEXT=IST2070I SRQRETRY COUNTERS LAST CLEARED ON % AT %(-2),
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2071I, MESSAGE ID-
  TEXT=IST2071I ALL DIAGNOSTIC COUNTERS CLEARED FOR % EE-
  CONNECTIONS',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2072I, MESSAGE ID-
  TEXT=IST2072I REXMIT COUNTERS CLEARED FOR % EE CONNECTIONS',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2073I, MESSAGE ID-
  TEXT=IST2073I SRQRETRY COUNTERS CLEARED FOR % EE CONNECTIONS',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2074I, MESSAGE ID-
  TEXT=IST2074I SUCCESSFUL SRQRETRY ATTEMPT = % OCCURRENCES = %2',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2075I, MESSAGE ID-
  TEXT=IST2075I DISPLAY RTPS SUMMARY INFORMATION',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2076I, MESSAGE ID-
  TEXT=IST2076I TOTAL MATCHING PIPES = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2077I, MESSAGE ID-
  TEXT=IST2077I CPSVCMG PIPES = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2078I, MESSAGE ID-
  TEXT=IST2078I RSETUP PIPES = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2079I, MESSAGE ID-
  TEXT=IST2079I LU-LU PIPES = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2080I, MESSAGE ID-
  TEXT=IST2080I PATH SWITCHING PIPES = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2081I, MESSAGE ID-
  TEXT=IST2081I CONGESTED PIPES = %',
  ROUTCDE=2, ROUTING CODE-
  DESC=5, DESCRIPTOR CODE-
  SUPP=NEVER, SUPPRESSION CODE-
  OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2082I, MESSAGE ID-
TEXT='IST2082I STALLED PIPES = %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2083I, MESSAGE ID-
TEXT='IST2083I SESSIONS = %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2084I, MESSAGE ID-
TEXT='IST2084I % OF %(2) MATCHING RTP PIPES DISPLAYED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2085I, MESSAGE ID-
TEXT='IST2085I NUMBER OF NLPS ON OUTBOUND WORK QUEUE = %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2086I, MESSAGE ID-
TEXT='IST2086I MAXIMUM NUMBER OF NLPS ON OUTBOUND WORK QUEUE = %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2087I, MESSAGE ID-
TEXT='IST2087I OUTBOUND WORK QUEUE MAX REACHED ON % AT %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2088I, MESSAGE ID-
TEXT='IST2088I CDRSCS DEFINED USING THIS MODEL:', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2089I, MESSAGE ID-
TEXT='IST2089I NO CDRSCS DEFINED USING THIS MODEL', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2090I, MESSAGE ID-
TEXT='IST2090I TRACE INITIATED FOR % CLONE CDRSCS', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2091I, MESSAGE ID-
TEXT='IST2091I TRACE ENDED FOR % CLONE CDRSCS', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2092I, MESSAGE ID-
TEXT='IST2092I NO CLONE CDRSCS EXIST FOR SCOPE %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2093I, MESSAGE ID-
TEXT='IST2093I AUTOLOGIN SEARCH INITIATED FOR % APPLICATIONS', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=INFO, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2094I, MESSAGE ID-
TEXT='IST2094I MODEL CDRSC WAS NOT DELETED - CLONE CDRSCS EXIST', -

Appendix A. IBM-supplied tables 1279
USSMSG MSG=IST2107I, MESSAGE ID-
TEXT='IST2107I LOCAL GR ACB IS OPEN', -
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-SER, SUPPRESSION CODE-
OPT-NOBLSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2108I, MESSAGE ID-
TEXT='IST2108I LOCAL APPLICATION OWNED GR AFFINITY EXISTS', -
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-SER, SUPPRESSION CODE-
OPT-NOBLSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2109I, MESSAGE ID-
TEXT='IST2109I NON-LOCAL GR INSTANCE SELECTABLE', -
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-SER, SUPPRESSION CODE-
OPT-NOBLSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2110I, MESSAGE ID-
TEXT='IST2110I NON-LOCAL APPLICATION OWNED GR AFFINITY EXISTS', -
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-SER, SUPPRESSION CODE-
OPT-NOBLSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2111I, MESSAGE ID-
TEXT='IST2111I NON-LOCAL GR SESSION ACTIVE', -
ROUTCDE=8, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-SER, SUPPRESSION CODE-
OPT-NOBLSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2113I, MESSAGE ID-
TEXT='IST2113I % SESSION DEACTIVATED - EXPFTRM THRESHOLD EXCEEDED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2114I, MESSAGE ID-
TEXT='IST2114I LIVETIME: INITIAL = %, MAXIMUM = %, CURRENT = %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2116I, MESSAGE ID-
TEXT='IST2116I STALL DETECTED FOR XCF TRLE % TO %', -
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP-WARN, SUPPRESSION CODE-
OPT-BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2117I, MESSAGE ID-
TEXT='IST2117I STALL CONTINUES FOR XCF TRLE % TO %', -
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP-WARN, SUPPRESSION CODE-
OPT-BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2118I, MESSAGE ID-
TEXT='IST2118I STALL ALLEVIATED FOR XCF TRLE % TO %', -
ROUTCDE=2, ROUTING CODE-
DESC=4, DESCRIPTOR CODE-
SUPP-WARN, SUPPRESSION CODE-
OPT-BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2119I, MESSAGE ID-
TEXT='IST2119I ENTERPRISE EXTENDER DISPLAY CORRELATOR:', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2120I, MESSAGE ID-
TEXT='IST2120I HOSTNAME RESOLUTION IN PROGRESS', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2121I, MESSAGE ID-
IST2121I HOSTNAME RESOLUTION COMPLETE.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2122I, MESSAGE ID=
TEXT='IST2122I EE DISPLAY REJECTED - MAXHNR LIMIT EXCEEDED'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2123I, MESSAGE ID=
TEXT='IST2123I DIAL FAILED FOR EE PU % - EXISTING CONNECTION FOUND'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2124I, MESSAGE ID=
TEXT='IST2124I REMOTE IPADDR OR HOSTNAME IS ALSO REQUIRED'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2125I, MESSAGE ID=
TEXT='IST2125I LOCAL IPADDR OR HOSTNAME IS ALSO REQUIRED'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2126I, MESSAGE ID=
TEXT='IST2126I CONNECTIVITY TEST IN PROGRESS'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2127I, MESSAGE ID=
TEXT='IST2127I CONNECTIVITY TEST FAILED - NO DIAL OUT LINE AVAILABLE FOR EE'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2128I, MESSAGE ID=
TEXT='IST2128I CONNECTIVITY TEST FAILED - EE NOT AVAILABLE FOR LOCAL IPADDR'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2129I, MESSAGE ID=
TEXT='IST2129I CONNECTIVITY TEST FAILED - MISMATCH OF IP ADDRESS FAMILY'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2130I, MESSAGE ID=
TEXT='IST2130I ENTERPRISE EXTENDER CONNECTIVITY TEST INFORMATION'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2131I, MESSAGE ID=
TEXT='IST2131I EEDIAG DISPLAY COMPLETED ON % AT %'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP, MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2132I, MESSAGE ID=
TEXT='IST2132I LDLC PROBE VERSIONS: VTAM = % PARTNER = %'.

ROUTE2=2, ROUTING CODE=
DESC5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2133I, MESSAGE ID-
TEXT='IST2133I INTFNAME: % INTFTYPE: %(2)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2134I, MESSAGE ID-
TEXT='IST2134I CONNECTIVITY SUCCESSFUL -
PORT: %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2135I, MESSAGE ID-
TEXT='IST2135I CONNECTIVITY UNSUCCESSFUL SENSE: % -
PORT: %(2)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2136I, MESSAGE ID-
TEXT='IST2136I CONNECTIVITY TEST ENDED - MAXIMUM TIME - LIMIT EXCEEDED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2137I, MESSAGE ID-
TEXT='IST2137I % %(2) %(3) RTT: %(4)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2138I, MESSAGE ID-
TEXT='IST2138I % %(2) %(3) RTT: %(4)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2139I, MESSAGE ID-
TEXT='IST2139I CONNECTIVITY TEST RESULTS DISPLAYED FOR %-OF %(2) ROUTES', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2140I, MESSAGE ID-
TEXT='IST2140I CONNECTIVITY TEST FAILED - LINE % IS INACTIVE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2141I, MESSAGE ID-
TEXT='IST2141I CONNECTIVITY TEST FAILED - SWITCHED PU IS NOT ENABLED FOR EE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DEScriptor CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2142I, MESSAGE ID-
TEXT='IST2142I CONNECTIVITY TEST REJECTED - SAME TEST ALREADY IN PROGRESS', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2143I, MESSAGE ID-
TEXT='IST2143I IN PROGRESS ENTERPRISE EXTENDER DISPLAY CORRELATOR: %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2144I, MESSAGE ID-
TEXT='IST2144I CONNECTIVITY TEST REJECTED - MAXEETST LIMIT EXCEEDED', -
ROUTCDE=2, ROUTING CODE-
USSMSG MSG=IST2145I, MESSAGE ID-
TEXT='IST2145I PENDING ENTERPRISE EXTENDER DISPLAY COMMANDS', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2146I, MESSAGE ID-
TEXT='IST2146I EE DISPLAY ISSUED ON % AT % (2)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2147I, MESSAGE ID-
TEXT='IST2147I CORRELATOR: % LINE: % (2) STATUS: % (3)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2148I, MESSAGE ID-
TEXT='IST2148I EE CONNECTIVITY TEST REACHES MAXTIME ON % AT % (2)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2149I, MESSAGE ID-
TEXT='IST2149I % OF % (2) CORRELATORS DISPLAYED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2150I, MESSAGE ID-
TEXT='IST2150I VIRTUAL NODE % - UNREACHABLE PARTNER(S)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2151I, MESSAGE ID-
TEXT='IST2151I PARTNER LIMIT EXCEEDED - UNUSABLE UNTIL COUNT IS BELOW %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2152I, MESSAGE ID-
TEXT='IST2152I MODIFY TOPO COMMAND FAILED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2153I, MESSAGE ID-
TEXT='IST2153I ORIG, VRN, OR DEST OPERAND REQUIRED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2154I, MESSAGE ID-
TEXT='IST2154I ROUTE SELECTION TRACE IS ACTIVE - BFRNUM = %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2155I, MESSAGE ID-
TEXT='IST2155I ROUTE SELECTION TRACE IS INACTIVE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2156I, MESSAGE ID-
TEXT='IST2156I STORAGE ALLOCATED TO ROUTE SELECTION TRACE = %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
USSMSG MSG=IST21821, MESSAGE ID-
TEXT='IST21821 UNRCHTM = %', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP
USSMSG MSG=IST21831, MESSAGE ID-
TEXT='IST21831 QDIOSYNC = % - SYNCID = %(2) - SAVED = %%(3)', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP
USSMSG MSG=IST21841, MESSAGE ID-
TEXT='IST21841 QDIOSYNC = % - SYNCID = %(2) - SAVED = %%(3)', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP
USSMSG MSG=IST21851, MESSAGE ID-
TEXT='IST21851 FRINVCTO = % FRINVCT = %(2)', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP
USSMSG MSG=IST21861, MESSAGE ID-
TEXT='IST21861 THIS PATH WILL NOT BE SELECTED FOR UNRCHT-IM SECONDS', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP
USSMSG MSG=IST21871, MESSAGE ID-
TEXT='IST21871 XCF SEND FAILURE ON TRLE % MESSAGE TYPE: %%(2)', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP
USSMSG MSG=IST21881, MESSAGE ID-
TEXT='IST21881 LDLC TIMER OPERANDS ON GROUP % IGNORED', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP
USSMSG MSG=IST21891, MESSAGE ID-
TEXT='IST21891 LDLC TIMER OPERANDS ALREADY SET BY GROUP - %', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP
USSMSG MSG=IST21901, MESSAGE ID-
TEXT='IST21901 DEVICEID PARAMETER FOR OSAENTA TRACE COMMAND = %', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP
USSMSG MSG=IST21911, MESSAGE ID-
TEXT='IST21911 HPR PATH SWITCH SUMMARY FROM % AT %(2)', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=BLKSUP
USSMSG MSG=IST21921, MESSAGE ID-
TEXT='IST21921 STARTED = %', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP
USSMSG MSG=IST21931, MESSAGE ID-
TEXT='IST21931 TGINOP = % SRQTIMER = %(2) PSRET- hydration = %%(3)', ROUTCDE=2, DESC=5, SUPP=NEVER, OPT=NOBLKSUP
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22001I, MESSAGE ID-
TEXT='IST22001I % PATH SWITCH EVENTS FOR %2 CPS WITHIN
%3 NETIDS', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22002I, MESSAGE ID-
TEXT='IST22002I % TABLE FOR %2', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22003I, MESSAGE ID-
TEXT='IST22003I % = %2 FROM START OPTION', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22004I, MESSAGE ID-
TEXT='IST22004I % = %2 FROM ADJCLUST TABLE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22101I, MESSAGE ID-
TEXT='IST22101I GR PREFERENCE TABLE ENTRY = %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22111I, MESSAGE ID-
TEXT='IST22111I ACK QUEUE MAX', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22121I, MESSAGE ID-
TEXT='IST22121I %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22131I, MESSAGE ID-
TEXT='IST22131I LAST BACKPRESSURE APPLIED ON % AT %2', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22141I, MESSAGE ID-
TEXT='IST22141I BACKPRESSURE REASON: PATHSWITCH', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22151I, MESSAGE ID-
TEXT='IST22151I BACKPRESSURE REASON: SEND QUEUE MAXIMUM
REACHED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22161I, MESSAGE ID-
TEXT='IST22161I BACKPRESSURE REASON: STORAGE FAILURE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22171I, MESSAGE ID-
TEXT='IST22171I BACKPRESSURE REASON: STALLED PIPE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST22181I, MESSAGE ID-
TEXT='IST22181I BACKPRESSURE REASON: WAITING-FOR-ACK QUEU
E MAXIMUM REACHED', -
ROUTCDE=2, ROUTING CODE-
IST2219I % ACTIVATION WAITING FOR MINIMUM NUMBER OF DEVICES

IST2220I % ACTIVATION RESUMING - ONLINE DEVICES DETECTED

IST2221I EXPLICITBINDPORTRANGE - START: % END: %

IST2222I QDIOSYNC CAPTURE INITIATED FOR TRL= %

IST2223I QDIOSYNC STATE ARMED FOR TRL= % AT TIME OF NOP

IST2224I ENTERPRISE EXTENDER ROUTING POLICY INFORMATION

IST2225I PORT ROUTE TABLE ROUTING RULE

IST2226I % % %

IST2227I CONNECTIVITY NOT TESTED - ROUTE NOT APPLICABLE PORT: %

IST2229I REFIFO TIMER = % MILLISECONDS

IST2230I MAXIMUM NUMBER OF NLPS ON OUT-OF-SEQUENCE QUEUE = %

IST2231I CURRENT HPR CLOCK RATE = %
Appendix A. IBM-supplied tables 1291
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables

TEXT='IST22801 ½ ½(2) ½(3) ½(4) ½(5)', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22801', MESSAGE ID=
TEXT='IST22801 LAST TDU SENT = ½(2)', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22802', MESSAGE ID=
TEXT='IST22802 TDU COUNTS:', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22803', MESSAGE ID=
TEXT='IST22803 NO TDUDIAG RSN UPDATES EXIST', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22804', MESSAGE ID=
TEXT='IST22804 FOR THE FOLLOWING NODES AND TGS', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22805', MESSAGE ID=
TEXT='IST22805 TDUS SENT BETWEEN ½(2) - ½(3) ½(4)', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22806', MESSAGE ID=
TEXT='IST22806 TDUS RECEIVED:', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22807', MESSAGE ID=
TEXT='IST22807 TDUS SENT:', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22808', MESSAGE ID=
TEXT='IST22808 CP NAME RSN DESTINATION CP-TGN SENT REC', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22809', MESSAGE ID=
TEXT='IST22809 RESOURCE SEQUENCE NUMBERS UPDATED BY THIS-NODE', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP

USSMSG MSG='IST22901', MESSAGE ID=
TEXT='IST22901 TDUDIAG START OPTION = ½', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP

USSMSG MSG='IST22902', MESSAGE ID=
TEXT='IST22902 UPDATED = ½', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP

USSMSG MSG='IST22903', MESSAGE ID=
TEXT='IST22903 CP NAME RSN DESTINATION CP-TGN UPDATED', ROUTING CODE=
DESC=5, MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
USSMSG MSG=IST2293I, MESSAGE ID-
TEXT='IST2293I % %(2) %(3) %(4) %(5)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2294I, MESSAGE ID-
TEXT='IST2294I TDUDIAG RSN UPDATES:', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2295I, MESSAGE ID-
TEXT='IST2295I TIME HEX RSN HEX', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2296I, MESSAGE ID-
TEXT='IST2296I CP NAME UPDATED BEFORE AFTER REASON', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2297I, MESSAGE ID-
TEXT='IST2297I TDUDIAG RSN THRESHOLD REACHED FOR NODE: ID = -', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2299I, MESSAGE ID-
TEXT='IST2299I TDUDIAG RSN THRESHOLD REACHED FOR TG: TGN = -', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2300I, MESSAGE ID-
TEXT='IST2300I RECEIVED FROM: -', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2301I, MESSAGE ID-
TEXT='IST2301I % OF %(2) TOPOLOGY RESOURCES DISPLAYED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2302I, MESSAGE ID-
TEXT='IST2302I MODEL % IS THE BEST ACTIVE MATCH FOR %(2)-', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2303I, MESSAGE ID-
TEXT='IST2303I THERE IS NO ACTIVE MODEL MATCH FOR %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2304I, MESSAGE ID-
TEXT='IST2304I % ALREADY EXISTS, TYPE = %(2)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2305I, MESSAGE ID-
TEXT='IST2305I NUMBER OF DISCARDED INBOUND READ BUFFERS = %', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2306I, MESSAGE ID= TEXT='IST2306I TDU DIAGNOSTIC INFORMATION FOR NODE: ID = %', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2307I, MESSAGE ID= TEXT='IST2307I THIS NODE DOES NOT SUPPORT UNKNOWN TOPOLOGY VECTORS', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2308I, MESSAGE ID= TEXT='IST2308I THAT HAVE SAVED TDU DIAG RSN UPDATES', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2309I, MESSAGE ID= TEXT='IST2309I ACCELERATED ROUTING ENABLED', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2310I, MESSAGE ID= TEXT='IST2310I ACCELERATED ROUTING DISABLED', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2311I, MESSAGE ID= TEXT='IST2311I TDU DIAGNOSTIC INFORMATION FOR TG: TGN = %', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2312I, MESSAGE ID= TEXT='IST2312I CURRENT RSN = (2)', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2313I, MESSAGE ID= TEXT='IST2313I TOTAL RSN UPDATES BY LOCAL HOST NODE = %', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2314I, MESSAGE ID= TEXT='IST2314I % OF (2) RSN UPDATES DISPLAYED', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2315I, MESSAGE ID= TEXT='IST2315I LAST TDU SENT = NONE', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2316I, MESSAGE ID= TEXT='IST2316I EARLYINO = (2) EARLYINT = %', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2317I, MESSAGE ID= TEXT='IST2317I ULPRETUO = (2) ULPRETU = %', ROUTCDE=2, ROUTING CODE= DEC=5, DESCRIPTOR CODE= SUPP=NEVER, SUPPRESSION CODE= OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
SUPP=NEVER, SUPPRESSION CODE=
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST23181, MESSAGE ID=
TEXT='IST2318I UNABLE TO OPEN % - NETWORK ADDRESS LIMIT -
REACHED',
ROUTCDE=2, ROUTING CODE=
DESC=5, DESCRIPTOR CODE=
SUPP=WARN, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST23191, MESSAGE ID=
TEXT='IST2319I IQD NETWORK ID = %',
ROUTCDE=2, ROUTING CODE=
DESC=7, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2320I, MESSAGE ID=
TEXT='IST2320I WTOR % FROM CONSOLE % (2) DELETED DUE TO VTAM HALT',
ROUTCDE=2, ROUTING CODE=
DESC=7, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2321I, MESSAGE ID=
TEXT='IST2321I TDUDIAG VALUE MUST BE NUMERIC, 'ALWAYS' , OR 'NEVER',
ROUTCDE=2, ROUTING CODE=
DESC=5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2322I, MESSAGE ID=
TEXT='IST2322I WTOR % FROM POA % (2) DELETED DUE TO VTAM HALT',
ROUTCDE=2, ROUTING CODE=
DESC=7, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2323E, MESSAGE ID=
TEXT='IST2323E EE HEALTH VERIFICATION FAILED FOR ONE OR MORE CONNECTIONS',
ROUTCDE=2, ROUTING CODE=
DESC=11, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2324I, MESSAGE ID=
TEXT='IST2324I EE HEALTH VERIFICATION FAILED ON % AT %',
ROUTCDE=2, ROUTING CODE=
DESC=5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2325I, MESSAGE ID=
TEXT='IST2325I LINE % PU % (3) ON % (3) AT % (4)',
ROUTCDE=2, ROUTING CODE=
DESC=5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2326I, MESSAGE ID=
TEXT='IST2326I EE HEALTH VERIFICATION TOTAL CONNECTION FAILURES = %',
ROUTCDE=2, ROUTING CODE=
DESC=5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2327I, MESSAGE ID=
TEXT='IST2327I EE HEALTH VERIFICATION OPTION - EEVERIFY = % MINUTES',
ROUTCDE=2, ROUTING CODE=
DESC=5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2328I, MESSAGE ID=
TEXT='IST2328I EE HEALTH VERIFICATION FAILED ON % AT %',
ROUTCDE=2, ROUTING CODE=
DESC=5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2329I, MESSAGE ID=
TEXT='IST2329I EE HEALTH VERIFICATION SUCCESSFUL ON % AT %',
ROUTCDE=2, ROUTING CODE=
DESC=5, DESCRIPTOR CODE=
SUPP=NEVER, SUPPRESSION CODE=
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables
USSMSG MSG=IST2393I, MESSAGE ID-
TEXT='IST2393I PFID % ALLOCATION FAILURE - PFID IS NOT O-
NLINE', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2394I, MESSAGE ID-
TEXT='IST2394I PFID % IS ALREADY ACTIVE ON %(2) USING TR-
CLE %(3)', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2395I, MESSAGE ID-
TEXT='IST2395I RTP PACING ALGORITHM = ARB BASE MODE', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2396I, MESSAGE ID-
TEXT='IST2396I RNIC STATISTICS FOR %', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2397I, MESSAGE ID-
TEXT='IST2397I DESCRIPTION OVERFL-
OW COUNT', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2398I, MESSAGE ID-
TEXT='IST2398I % %(2) %(3)', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2399I, MESSAGE ID-
TEXT='IST2399I MESSAGE TRIGGER: RNICTRLE = %', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2400I, MESSAGE ID-
TEXT='IST2400I NO ADAPTER DIAGNOSTICS PRODUCED FOR %: %(-
2)', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2401I, MESSAGE ID-
TEXT='IST2401I DEVSTATS REJECTED FOR TRLE % - DEVICE NOT-
ACTIVE', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2402I, MESSAGE ID-
TEXT='IST2402I DEVSTATS FAILED FOR TRLE % - DEVICE NOT O-
PERATIONAL', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2403I, MESSAGE ID-
TEXT='IST2403I 64-BIT STORAGE TYPE CURRENT -
MAXIMUM LIMIT', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2404I, MESSAGE ID-
TEXT='IST2404I HVCOMMON % %(2) %(3)', -
ROUTCD=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP=NEVER, SUPPRESSION CODE-
OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IST2405I, MESSAGE ID-
   TEXT='IST2405I TRACE HVCOMMON % %(2) %(3)',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2406I, MESSAGE ID-
   TEXT='IST2406I SMC-R LINK FAILURE ON TRLE % CODE = %(2)',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2407I, MESSAGE ID-
   TEXT='IST2407I LOCAL LINK ID = % REMOTE LINK ID = %(2)',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2408I, MESSAGE ID-
   TEXT='IST2408I LOCAL MAC = % REMOTE MAC = %(2)',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2409I, MESSAGE ID-
   TEXT='IST2409I % GID = %(2)',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2410I, MESSAGE ID-
   TEXT='IST2410I LOCAL QP = % REMOTE QP = %(2)',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2411I, MESSAGE ID-
   TEXT='IST2411I VLAN = %',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2412I, MESSAGE ID-
   TEXT='IST2412I FIXED HVCOMMON % %(2) %(3)',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2413I, MESSAGE ID-
   TEXT='IST2413I FIXED PRIVATE % %(2) %(3)',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2414I, MESSAGE ID-
   TEXT='IST2414I TOTAL FIXED % %(2) %(3)',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IST2415I, MESSAGE ID-
   TEXT='IST2415I % PROCESSED',
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS

USSMSG MSG=IUT5000I, MESSAGE ID-
   TEXT='IUT5000I % STILL ACTIVE: VTAM TERMINATION WAITING - FOR %(2)',
   ROUTCDE=2, ROUTING CODE-
   DESC=4, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IUT5001I, MESSAGE ID-
TEXT='IUT5001I VTAM MPC CONNECTION MANAGER PROCESSING TERMINATED', - ROUTCODE=2, ROUTING CODE- DESC=6, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IUT5002I, MESSAGE ID-
TEXT='IUT5002I TASK FOR ULPID % USING TRLE %(2) TERMINATING', - ROUTCODE=2, ROUTING CODE- DESC=4, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5501I, MESSAGE ID-
TEXT='IVT5501I CSM PARMLIB MEMBER % NOT FOUND - DEFAULT - VALUES USED', - ROUTCODE=2, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5502I, MESSAGE ID-
TEXT='IVT5502I READ ERROR ON PARMLIB MEMBER % - DEFAULT - VALUES USED', - ROUTCODE=2, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5503I, MESSAGE ID-
TEXT='IVT5503I CSM PARMLIB PARAMETER NOT VALID - %', - ROUTCODE=2, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5504I, MESSAGE ID-
TEXT='IVT5504I ABEND % OCCURRED IN %(2) - SDUMP HAS BEEN ISSUED', - ROUTCODE=2, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5505I, MESSAGE ID-
TEXT='IVT5505I CSM SDUMP FAILED WITH RETURN CODE % REASON X''%(2)''', - ROUTCODE=2, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5506I, MESSAGE ID-
TEXT='IVT5506I % STORAGE ALLOCATION FAILED IN CSM', - ROUTCODE=2, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5507I, MESSAGE ID-
TEXT='IVT5507I CSM PARMLIB INFORMATION FOUND IN MEMBER %', - ROUTCODE=2, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5508I, MESSAGE ID-
TEXT='IVT5508I DISPLAY ACCEPTED', - ROUTCODE=6, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5510I, MESSAGE ID-
TEXT='IVT5510I MODIFY ACCEPTED', - ROUTCODE=6, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE- OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5511I, MESSAGE ID-
TEXT='IVT5511I CSM COMMAND SYNTAX NOT VALID', - ROUTCODE=2, ROUTING CODE- DESC=5, DESCRIPTOR CODE- SUPP=NEVER, SUPPRESSION CODE-
Appendix A. IBM-supplied tables
Appendix A. IBM-supplied tables

1309
USSMSG MSG=IVT5570I, MESSAGE ID-
TEXT='IVT5570I CSM MONITOR IS DYNAMIC AND CURRENTLY NOT -
ACTIVE', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5571I, MESSAGE ID-
TEXT='IVT5571I MONITOR PARAMETER IS NOT VALID WITH %', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5572I, MESSAGE ID-
TEXT='IVT5572I PROCESSING DISPLAY CSMUSE COMMAND - OWNER-
ID NOT SPECIFIED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5573I, MESSAGE ID-
TEXT='IVT5573I PROCESSING DISPLAY CSMUSE COMMAND - OWNER-
ID SPECIFIED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5574I, MESSAGE ID-
TEXT='IVT5574I PROCESSING DISPLAY CSMUSE COMMAND - POOL -
SPECIFIED', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5575I, MESSAGE ID-
TEXT='IVT5575I USAGE SUMMARY - % POOL TOTAL (ALL USERS) -
= %(2)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5576I, MESSAGE ID-
TEXT='IVT5576I AMOUNT MONITOR ID OWNERID JOBNAME', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5577I, MESSAGE ID-
TEXT='IVT5577I % %(2) %(3) %(4)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5578I, MESSAGE ID-
TEXT='IVT5578I DISPLAY TOTAL FOR % POOL (%(2) USERS) -
= %(3)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5579I, MESSAGE ID-
TEXT='IVT5579I BUFFER USE FOR % : USECNT USERDATA M-
ONITOR HISTORY', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5580I, MESSAGE ID-
TEXT='IVT5580I % %(2) %(3)', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
OPT-NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5581I, MESSAGE ID-
TEXT='IVT5581I POOL NAME NOT VALID', -
ROUTCDE=2, ROUTING CODE-
DESC=5, DESCRIPTOR CODE-
SUPP-NEVER, SUPPRESSION CODE-
USSMSG MSG=IVT5594I, MESSAGE ID-
   TEXT='IVT5594I % MAXIMUM USED = %2 SINCE IPL', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5595I, MESSAGE ID-
   TEXT='IVT5595I HVCOMM MAXIMUM = %', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5596I, MESSAGE ID-
   TEXT='IVT5596I HVCOMM PARAMETER VALUE NOT VALID', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5598I, MESSAGE ID-
   TEXT='IVT5598I CSM HVCOMM STORAGE LIMIT EXCEEDED', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5599I, MESSAGE ID-
   TEXT='IVT5599I END', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5600I, MESSAGE ID-
   TEXT='IVT5600I PROBE % ATTEMPTED - FFST NOT AVAILABLE', -
   ROUTCDE=5, ROUTING CODE-
   DESC=2, DESCRIPTOR CODE-
   SUPP=WARN, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5603I, MESSAGE ID-
   TEXT='IVT5603I % HVCOMM STORAGE ALLOCATION FAILED IN CSM-', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=BLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5604I, MESSAGE ID-
   TEXT='IVT5604I HVCOMM MAXIMUM = % HVCOMM CURRENT = %2-', -
   ROUTCDE=2, ROUTING CODE-
   DESC=5, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5605I, MESSAGE ID-
   TEXT='IVT5605I CSM HVCOMM STORAGE AT CRITICAL LEVEL', -
   ROUTCDE=2, ROUTING CODE-
   DESC=6, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5606I, MESSAGE ID-
   TEXT='IVT5606I CSM HVCOMM STORAGE SHORTAGE RELIEVED', -
   ROUTCDE=2, ROUTING CODE-
   DESC=6, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSMSG MSG=IVT5607I, MESSAGE ID-
   TEXT='IVT5607I CSM HVCOMM STORAGE AT CONstrained LEVEL', -
   ROUTCDE=2, ROUTING CODE-
   DESC=6, DESCRIPTOR CODE-
   SUPP=NEVER, SUPPRESSION CODE-
   OPT=NOBLKSUP MESSAGE PROCESSING OPTIONS
USSEND
* END OF ISTINCN0 USS DEFINITION TABLE END
APPN Class of Service (CoS) definitions (COSAPPN)

Three sets of IBM-supplied APPN Class of Service definitions are available:

- COSAPPN
- ISTACST2
- ISTACST3

COSAPPN

<table>
<thead>
<tr>
<th>CPSVCMG</th>
<th>APPNCOS</th>
<th>PRIORITY=NETWORK</th>
<th>transmission priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINEROW</td>
<td>WEIGHT=30,</td>
<td>line row weight</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>NUMBER=1,</td>
<td>line row number</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>U Parm1=(0,255),</td>
<td>user defined char 1</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>U Parm2=(0,255),</td>
<td>user defined char 2</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>U Parm3=(0,255),</td>
<td>user defined char 3</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>CAPACITY=(4M,MAXIMUM),</td>
<td>line speed</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>COSTTIME=(0,0),</td>
<td>cost per connect time</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>COSTBYTE=(0,0),</td>
<td>cost per byte transmitted</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>P DELAY=(MINIMUM,NEGLIGIB),</td>
<td>propagation delay</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>SECURITY=(UNSECURE,MAXIMUM)</td>
<td>security level for TG</td>
<td>*</td>
</tr>
</tbody>
</table>

| NODEROW | NUMBER=1, | node row number | * |
|         | WEIGHT=5, | node row weight | * |
|         | CONGEST=(LOW,LOW), | congestion | * |
|         | ROUTERES=(0,0,31), | route addition resistance | * |
| LINEROW | WEIGHT=60, | line row weight | * |
|         | NUMBER=2, | line row number | * |
|         | U Parm1=(0,255), | user defined char 1 | * |
|         | U Parm2=(0,255), | user defined char 2 | * |
|         | U Parm3=(0,255), | user defined char 3 | * |
|         | CAPACITY=(56000,MAXIMUM), | line speed | * |
|         | COSTTIME=(0,0), | cost per connect time | * |
|         | COSTBYTE=(0,0), | cost per byte transmitted | * |
|         | P DELAY=(MINIMUM,TERRESTRI), | propagation delay | * |
|         | SECURITY=(UNSECURE,MAXIMUM) | security level for TG | * |

| NODEROW | NUMBER=2, | node row number | * |
|         | WEIGHT=10, | node row weight | * |
|         | CONGEST=(LOW,LOW), | congestion | * |
|         | ROUTERES=(0,0,63), | route addition resistance | * |
| LINEROW | WEIGHT=90, | line row weight | * |
|         | NUMBER=3, | line row number | * |
|         | U Parm1=(0,255), | user defined char 1 | * |
|         | U Parm2=(0,255), | user defined char 2 | * |
|         | U Parm3=(0,255), | user defined char 3 | * |
|         | CAPACITY=(19200,MAXIMUM), | line speed | * |
|         | COSTTIME=(0,0), | cost per connect time | * |
|         | COSTBYTE=(0,0), | cost per byte transmitted | * |
|         | P DELAY=(MINIMUM,TERRESTRI), | propagation delay | * |
|         | SECURITY=(UNSECURE,MAXIMUM) | security level for TG | * |

| NODEROW | NUMBER=3, | node row number | * |
|         | WEIGHT=20, | node row weight | * |
|         | CONGEST=(LOW,LOW), | congestion | * |
|         | ROUTERES=(0,0,95), | route addition resistance | * |
| LINEROW | WEIGHT=120, | line row weight | * |
|         | NUMBER=4, | line row number | * |
|         | U Parm1=(0,255), | user defined char 1 | * |
|         | U Parm2=(0,255), | user defined char 2 | * |
|         | U Parm3=(0,255), | user defined char 3 | * |
|         | CAPACITY=(9600,MAXIMUM), | line speed | * |
|         | COSTTIME=(0,0), | cost per connect time | * |
|         | COSTBYTE=(0,0), | cost per byte transmitted | * |
|         | P DELAY=(MINIMUM,TERRESTRI), | propagation delay | * |
|         | SECURITY=(UNSECURE,MAXIMUM) | security level for TG | * |

<p>| NODEROW | NUMBER=4, | node row number | * |
|         | WEIGHT=40, | node row weight | * |
|         | CONGEST=(LOW,LOW), | congestion | * |</p>
<table>
<thead>
<tr>
<th>NODEROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGEST</th>
<th>ROUTESES</th>
<th>LINEROW WEIGHT</th>
<th>NUMBER</th>
<th>UPARM1</th>
<th>UPARM2</th>
<th>UPARM3</th>
<th>CAPACITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>LOW,LOW</td>
<td>(0,31)</td>
<td>60</td>
<td>2</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(56000,MAX)</td>
<td>(0,0)</td>
<td>(0,0)</td>
<td>(MINIMUM,TERRESTR)</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>LOW,LOW</td>
<td>(0,63)</td>
<td>90</td>
<td>3</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(19200,MAX)</td>
<td>(0,0)</td>
<td>(0,0)</td>
<td>(MINIMUM,TERRESTR)</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>LOW,LOW</td>
<td>(0,95)</td>
<td>120</td>
<td>4</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(9600,MAX)</td>
<td>(0,0)</td>
<td>(0,0)</td>
<td>(MINIMUM,TERRESTR)</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>LOW,LOW</td>
<td>(0,127)</td>
<td>150</td>
<td>5</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(19200,MAX)</td>
<td>(0,0)</td>
<td>(0,0)</td>
<td>(MINIMUM,PACKET)</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>5</td>
<td>60</td>
<td>LOW,LOW</td>
<td>(0,159)</td>
<td>180</td>
<td>6</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(9600,MAX)</td>
<td>(0,0)</td>
<td>(0,0)</td>
<td>(MINIMUM,PACKET)</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
</tbody>
</table>

PDELAY = (MINIMUM, NEGLIGIB), propagation delay
SECURITY = (UNSECURE, MAXIMUM), security level for TG
WEIGHT = 5, node row weight
CONGEST = (LOW, LOW), congestion
ROUTERSES = (0,31), route addition resistance
LINEROW WEIGHT = 60, line row weight
NUMBER = 2, line row number
UPARM1 = (0,255), user defined char 1
UPARM2 = (0,255), user defined char 2
UPARM3 = (0,255), user defined char 3
CAPACITY = (56000, MAXIMUM), line speed
COSTTIME = (0,0), cost per connect time
COSTBYTE = (0,0), cost per byte transmitted
PDELAY = (MINIMUM, TERRESTR), propagation delay
SECURITY = (UNSECURE, MAXIMUM), security level for TG
NODEROW NUMBER = 1, node row number
WEIGHT = 5, node row weight
CONGEST = (LOW, LOW), congestion
ROUTERSES = (0,31), route addition resistance
LINEROW WEIGHT = 60, line row weight
NUMBER = 2, line row number
UPARM1 = (0,255), user defined char 1
UPARM2 = (0,255), user defined char 2
UPARM3 = (0,255), user defined char 3
CAPACITY = (56000, MAXIMUM), line speed
COSTTIME = (0,0), cost per connect time
COSTBYTE = (0,0), cost per byte transmitted
PDELAY = (MINIMUM, TERRESTR), propagation delay
SECURITY = (UNSECURE, MAXIMUM), security level for TG
NODEROW NUMBER = 2, node row number
WEIGHT = 10, node row weight
CONGEST = (LOW, LOW), congestion
ROUTERSES = (0,63), route addition resistance
LINEROW WEIGHT = 90, line row weight
NUMBER = 3, line row number
UPARM1 = (0,255), user defined char 1
UPARM2 = (0,255), user defined char 2
UPARM3 = (0,255), user defined char 3
CAPACITY = (19200, MAXIMUM), line speed
COSTTIME = (0,0), cost per connect time
COSTBYTE = (0,0), cost per byte transmitted
PDELAY = (MINIMUM, TERRESTR), propagation delay
SECURITY = (UNSECURE, MAXIMUM), security level for TG
NODEROW NUMBER = 3, node row number
WEIGHT = 20, node row weight
CONGEST = (LOW, LOW), congestion
ROUTERSES = (0,95), route addition resistance
LINEROW WEIGHT = 120, line row weight
NUMBER = 4, line row number
UPARM1 = (0,255), user defined char 1
UPARM2 = (0,255), user defined char 2
UPARM3 = (0,255), user defined char 3
CAPACITY = (9600, MAXIMUM), line speed
COSTTIME = (0,0), cost per connect time
COSTBYTE = (0,0), cost per byte transmitted
PDELAY = (MINIMUM, TERRESTR), propagation delay
SECURITY = (UNSECURE, MAXIMUM), security level for TG
NODEROW NUMBER = 4, node row number
WEIGHT = 40, node row weight
CONGEST = (LOW, LOW), congestion
ROUTERSES = (0,127), route addition resistance
LINEROW WEIGHT = 150, line row weight
NUMBER = 5, line row number
UPARM1 = (0,255), user defined char 1
UPARM2 = (0,255), user defined char 2
UPARM3 = (0,255), user defined char 3
CAPACITY = (19200, MAXIMUM), line speed
COSTTIME = (0,0), cost per connect time
COSTBYTE = (0,0), cost per byte transmitted
PDELAY = (MINIMUM, PACKET), propagation delay
SECURITY = (UNSECURE, MAXIMUM), security level for TG
NODEROW NUMBER = 5, node row number
WEIGHT = 60, node row weight
CONGEST = (LOW, LOW), congestion
ROUTERSES = (0,159), route addition resistance
LINEROW WEIGHT = 180, line row weight
NUMBER = 6, line row number
UPARM1 = (0,255), user defined char 1
UPARM2 = (0,255), user defined char 2
UPARM3 = (0,255), user defined char 3
CAPACITY=(9600, MAXIMUM), line speed
COSTTIME=(0, 128), cost per connect time
COSTBYTE=(0, 128), cost per byte transmitted
PDELAY=(MINIMUM, PACKET), propagation delay
SECURITY=(UNSECURE, MAXIMUM)

NODEROW NUMBER=6, node row number
WEIGHT=80, node row weight
CONGEST=(LOW, LOW), congestion
ROUTERES=(0, 191), route addition resistance

LINEROW WEIGHT=210, line row weight
NUMBER=7, line row number
UPARM1=(0, 255), user defined char 1
UPARM2=(0, 255), user defined char 2
UPARM3=(0, 255), user defined char 3
CAPACITY=(48000, MAXIMUM), line speed
COSTTIME=(0, 196), cost per connect time
COSTBYTE=(0, 196), cost per byte transmitted
PDELAY=(MINIMUM, MAXIMUM), propagation delay
SECURITY=(UNSECURE, MAXIMUM)

NODEROW NUMBER=7, node row number
WEIGHT=120, node row weight
CONGEST=(LOW, HIGH), congestion
ROUTERES=(0, 223), route addition resistance

LINEROW WEIGHT=240, line row weight
NUMBER=8, line row number
UPARM1=(0, 255), user defined char 1
UPARM2=(0, 255), user defined char 2
UPARM3=(0, 255), user defined char 3
CAPACITY=(MINIMUM, MAXIMUM), line speed
COSTTIME=(0, 255), cost per connect time
COSTBYTE=(0, 255), cost per byte transmitted
PDELAY=(MINIMUM, MAXIMUM), propagation delay
SECURITY=(UNSECURE, MAXIMUM)

NODEROW NUMBER=8, node row number
WEIGHT=160, node row weight
CONGEST=(LOW, HIGH), congestion
ROUTERES=(0, 255), route addition resistance

* #CONNECT APPNOS PRIORITY=MEDIUM transmission priority
LINEROW WEIGHT=30, line row weight
NUMBER=1, line row number
UPARM1=(0, 255), user defined char 1
UPARM2=(0, 255), user defined char 2
UPARM3=(0, 255), user defined char 3
CAPACITY=(4M, MAXIMUM), line speed
COSTTIME=(0, 0), cost per connect time
COSTBYTE=(0, 0), cost per byte transmitted
PDELAY=(MINIMUM, NEGLIGIB), propagation delay
SECURITY=(UNSECURE, MAXIMUM)

NODEROW NUMBER=1, node row number
WEIGHT=5, node row weight
CONGEST=(LOW, LOW), congestion
ROUTERES=(0, 31), route addition resistance

LINEROW WEIGHT=60, line row weight
NUMBER=2, line row number
UPARM1=(0, 255), user defined char 1
UPARM2=(0, 255), user defined char 2
UPARM3=(0, 255), user defined char 3
CAPACITY=(56000, MAXIMUM), line speed
COSTTIME=(0, 0), cost per connect time
COSTBYTE=(0, 0), cost per byte transmitted
PDELAY=(MINIMUM, TERRESTR), propagation delay
SECURITY=(UNSECURE, MAXIMUM)

NODEROW NUMBER=2, node row number
WEIGHT=10, node row weight
CONGEST=(LOW, LOW), congestion
ROUTERES=(0, 63), route addition resistance
<table>
<thead>
<tr>
<th>Linerow number</th>
<th>Line row weight</th>
<th>Line number</th>
<th>User defined char 1</th>
<th>User defined char 2</th>
<th>User defined char 3</th>
<th>Line speed</th>
<th>Cost per connect time</th>
<th>Cost per byte transmitted</th>
<th>Propagation delay</th>
<th>Security level for TG</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>90</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>120</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>150</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>180</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>210</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Appendix A. IBM-supplied tables
<table>
<thead>
<tr>
<th>NODEROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGEST</th>
<th>ROUTERES</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>40</td>
<td>LOW,LOW</td>
<td>(0,127)</td>
<td>0</td>
<td>0</td>
<td>MINIMUM</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>5</td>
<td>60</td>
<td>LOW,LOW</td>
<td>(0,159)</td>
<td>0</td>
<td>0</td>
<td>MINIMUM</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>6</td>
<td>80</td>
<td>LOW,LOW</td>
<td>(0,191)</td>
<td>0</td>
<td>0</td>
<td>MINIMUM</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>7</td>
<td>120</td>
<td>LOW,HIGH</td>
<td>(0,223)</td>
<td>0</td>
<td>0</td>
<td>MINIMUM</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>LOW,HIGH</td>
<td>(0,255)</td>
<td>0</td>
<td>0</td>
<td>MINIMUM</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NODEROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGEST</th>
<th>ROUTERES</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>60</td>
<td>LOW,LOW</td>
<td>(0,159)</td>
<td>0</td>
<td>0</td>
<td>MINIMUM</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>6</td>
<td>80</td>
<td>LOW,LOW</td>
<td>(0,191)</td>
<td>0</td>
<td>0</td>
<td>MINIMUM</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>7</td>
<td>120</td>
<td>LOW,HIGH</td>
<td>(0,223)</td>
<td>0</td>
<td>0</td>
<td>MINIMUM</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>LOW,HIGH</td>
<td>(0,255)</td>
<td>0</td>
<td>0</td>
<td>MINIMUM</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
</tbody>
</table>

* #INTERSC APPNCS PRIORITY=HIGH transmission priority
  LINEROW WEIGHT=30, line row weight *
CONGEST=(LOW,LOW), congestion
ROUTERES=(0,159), route addition resistance
LINEROW WEIGHT=180, line row weight
NUMBER=6, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(9600,MAXIMUM), line speed
COSTTIME=(0,0), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted
PDELAY=(MINIMUM,PACKET), propagation delay
SECURITY=(PUBLIC,MAXIMUM) security level for TG
NODEROW NUMBER=6, node row number
WEIGHT=80, node row weight
CONGEST=(LOW,LOW), congestion
ROUTERES=(0,191), route addition resistance
LINEROW WEIGHT=210, line row weight
NUMBER=7, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(9600,MAXIMUM), line speed
COSTTIME=(0,196), cost per connect time
COSTBYTE=(0,196), cost per byte transmitted
PDELAY=(MINIMUM,MAXIMUM), propagation delay
SECURITY=(PUBLIC,MAXIMUM) security level for TG
NODEROW NUMBER=7, node row number
WEIGHT=120, node row weight
CONGEST=(LOW,HIGH), congestion
ROUTERES=(0,223), route addition resistance
LINEROW WEIGHT=240, line row weight
NUMBER=8, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(MINIMUM,MAXIMUM), line speed
COSTTIME=(0,255), cost per connect time
COSTBYTE=(0,255), cost per byte transmitted
PDELAY=(MINIMUM,MAXIMUM), propagation delay
SECURITY=(PUBLIC,MAXIMUM) security level for TG
NODEROW NUMBER=8, node row number
WEIGHT=160, node row weight
CONGEST=(LOW,HIGH), congestion
ROUTERES=(0,255), route addition resistance

#BATCH APPNCS PRIORITY=LOW transmission priority
LINEROW WEIGHT=30, line row weight
NUMBER=1, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(56000,MAXIMUM), line speed
COSTTIME=(0,0), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted
PDELAY=(MINIMUM,MAXIMUM), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=1, node row number
WEIGHT=5, node row weight
CONGEST=(LOW,LOW), congestion
ROUTERES=(0,31), route addition resistance
LINEROW WEIGHT=60, line row weight
NUMBER=2, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(19200,MAXIMUM), line speed
COSTTIME=(0,0), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=2, node row number *
WEIGHT=10, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,63) route addition resistance
LINEROW WEIGHT=90, line row weight *
NUMBER=3, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(19200,MAXIMUM), line speed *
COSTTIME=0, cost per connect time *
COSTBYTE=(0,128), cost per byte transmitted *
PDELAY=0, propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=3, node row number *
WEIGHT=20, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,95) route addition resistance
LINEROW WEIGHT=120, line row weight *
NUMBER=4, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(19200,MAXIMUM), line speed *
COSTTIME=0, cost per connect time *
COSTBYTE=0, cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=4, node row number *
WEIGHT=40, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,127) route addition resistance
LINEROW WEIGHT=150, line row weight *
NUMBER=5, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(9600,MAXIMUM), line speed *
COSTTIME=0, cost per connect time *
COSTBYTE=0, cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=5, node row number *
WEIGHT=60, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,159) ROUTE ADDITION RESISTANCE
LINEROW WEIGHT=180, line row weight *
NUMBER=6, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(9600,MAXIMUM), line speed *
COSTTIME=0, cost per connect time *
COSTBYTE=0, cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=6, node row number *
WEIGHT=80, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,191) route addition resistance
LINEROW WEIGHT=210, line row weight *
NUMBER=7, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
<table>
<thead>
<tr>
<th>NODEROW NUMBER</th>
<th>NODE ROW NUMBER</th>
<th>NODE ROW WEIGHT</th>
<th>CONGESTION</th>
<th>ROUTERES</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>120</td>
<td>7</td>
<td>LOW, HIGH</td>
<td>0,223</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>8</td>
<td>LOW, HIGH</td>
<td>0,255</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>1</td>
<td>LOW, LOW</td>
<td>0,31</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>2</td>
<td>LOW, LOW</td>
<td>0,63</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>3</td>
<td>LOW, LOW</td>
<td>0,63</td>
</tr>
</tbody>
</table>

*UPARM3=(0,255), user defined char 3*
*CAPACITY=(4800,MAXIMUM), line speed*
*COSTTIME=(0,196), cost per connect time*
*COSTBYTE=(0,196), cost per byte transmitted*
*PDELAY=(MINIMUM,MAXIMUM), propagation delay*
*SECURITY=(UNSECURE,MAXIMUM), security level for TG*

#BATCHSC APPNCOS PRIORITY=LOW transmission priority

<table>
<thead>
<tr>
<th>LINEROW WEIGHT</th>
<th>LINE ROW NUMBER</th>
<th>LINE ROW WEIGHT</th>
<th>CONGESTION</th>
<th>ROUTERES</th>
</tr>
</thead>
<tbody>
<tr>
<td>240</td>
<td>8</td>
<td>240</td>
<td>LOW, HIGH</td>
<td>0,255</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>1</td>
<td>30</td>
<td>LOW, LOW</td>
<td>0,31</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>60</td>
<td>LOW, LOW</td>
<td>0,63</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>3</td>
<td>90</td>
<td>LOW, LOW</td>
<td>0,63</td>
</tr>
</tbody>
</table>

*UPARM1=(0,255), user defined char 1*
*UPARM2=(0,255), user defined char 2*
*UPARM3=(0,255), user defined char 3*
*CAPACITY=(MINIMUM,MAXIMUM), line speed*
*COSTTIME=(0,255), cost per connect time*
*COSTBYTE=(0,255), cost per byte transmitted*
*PDELAY=(MINIMUM,MAXIMUM), propagation delay*
*SECURITY=(UNSECURE,MAXIMUM), security level for TG*

<table>
<thead>
<tr>
<th>NODEROW NUMBER</th>
<th>NODE ROW NUMBER</th>
<th>NODE ROW WEIGHT</th>
<th>CONGESTION</th>
<th>ROUTERES</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>160</td>
<td>8</td>
<td>LOW, HIGH</td>
<td>0,255</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>2</td>
<td>LOW, LOW</td>
<td>0,63</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>3</td>
<td>LOW, LOW</td>
<td>0,63</td>
</tr>
</tbody>
</table>

*CAPACITY=(19200,MAXIMUM), line speed*
*COSTTIME=(0,128), cost per connect time*
*COSTBYTE=(0,128), cost per byte transmitted*
*PDELAY=(MINIMUM,MAXIMUM), propagation delay*
*SECURITY=(PUBLIC,MAXIMUM), security level for TG*
<table>
<thead>
<tr>
<th>NODEROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGEST</th>
<th>ROUTERES</th>
<th>LINEROW WEIGHT</th>
<th>NUMBER</th>
<th>UPARM1</th>
<th>UPARM2</th>
<th>UPARM3</th>
<th>CAPACITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
<th>SECURITY</th>
<th>NODEROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGEST</th>
<th>ROUTERES</th>
<th>LINEROW WEIGHT</th>
<th>NUMBER</th>
<th>UPARM1</th>
<th>UPARM2</th>
<th>UPARM3</th>
<th>CAPACITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
<th>SECURITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>40</td>
<td>LOW,LOW</td>
<td>(0,127)</td>
<td>120</td>
<td>4</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(9600,MAXIMUM)</td>
<td>(0,128)</td>
<td>(0,128)</td>
<td>(MINIMUM,MAXIMUM)</td>
<td>(PUBLIC,MAXIMUM)</td>
<td>4</td>
<td>60</td>
<td>LOW,LOW</td>
<td>(0,127)</td>
<td>150</td>
<td>5</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(9600,MAXIMUM)</td>
<td>(0,128)</td>
<td>(0,128)</td>
<td>(MINIMUM,MAXIMUM)</td>
<td>(PUBLIC,MAXIMUM)</td>
<td>5</td>
<td>60</td>
</tr>
<tr>
<td>Node Row Number</td>
<td>Node Weight</td>
<td>Congestion</td>
<td>Route Addition Resistance</td>
<td>Line Row Weight</td>
<td>Line Row Number</td>
<td>User Defined Char 1</td>
<td>User Defined Char 2</td>
<td>User Defined Char 3</td>
<td>Line Speed</td>
<td>Cost Per Connect Time</td>
<td>Cost Per Byte Transmitted</td>
<td>Propagation Delay</td>
<td>Security Level for TG</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------</td>
<td>------------</td>
<td>--------------------------</td>
<td>-----------------</td>
<td>----------------</td>
<td>-------------------</td>
<td>-------------------</td>
<td>-------------------</td>
<td>-----------</td>
<td>---------------------</td>
<td>--------------------------</td>
<td>----------------</td>
<td>-------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>LOW, HIGH</td>
<td>(0,255)</td>
<td>8</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UNSECURE, MAXIMUM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>LOW, LOW</td>
<td>(0,31)</td>
<td>30</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SECURITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>LOW, LOW</td>
<td>(0,63)</td>
<td>60</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SECURITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>LOW, LOW</td>
<td>(0,95)</td>
<td>90</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SECURITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>LOW, LOW</td>
<td>(0,127)</td>
<td>120</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SECURITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**ISTACST2**

<table>
<thead>
<tr>
<th>CPSVCMG</th>
<th>APPNCOS</th>
<th>PRIORITY=NETWORK</th>
<th>Transmission Priority</th>
<th>Line Row Weight</th>
<th>Line Row Number</th>
<th>User Defined Char 1</th>
<th>User Defined Char 2</th>
<th>User Defined Char 3</th>
<th>Line Speed</th>
<th>Cost Per Connect Time</th>
<th>Cost Per Byte Transmitted</th>
<th>Propagation Delay</th>
<th>Security Level for TG</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UNSECURE, MAXIMUM</td>
</tr>
</tbody>
</table>

**Appendix A. IBM-supplied tables**

1325
SNASVCMG APPNCOS PRIORITY=NETWORK transmission priority
LINEROW WEIGHT=30, line row weight
NUMBER=1, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(4M,MAXIMUM), line speed
COSTTIME=(0,0), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted
PDELAY=(MINIMUM,NEGLIGIB), propagation delay
SECURITY=(UNSECURE,MAXIMUM), security level for TG
NODEROW NUMBER=1, node row number
WEIGHT=5, node row weight

*
<table>
<thead>
<tr>
<th>NODE ROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGESTION</th>
<th>ROUTERES</th>
<th>LINEROW WEIGHT</th>
<th>NUMBER</th>
<th>UPARM1</th>
<th>UPARM2</th>
<th>UPARM3</th>
<th>CAPACITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDDELAY</th>
<th>SECURITY</th>
<th>NOTES</th>
</tr>
</thead>
</table>
| 2              | 10        | LOW,LOW      | (0,31)     | 60             | 2      | 0,255 | 0,255 | 0,255 | 56000,MAXIMUM | (0,0)   | (0,0)   | (MINIMUM,TERRESTR) | (UNSECURE,MAXIMUM) | *
| 3              | 20        | LOW,LOW      | (0,63)     | 90             | 3      | 0,255 | 0,255 | 0,255 | 19200,MAXIMUM | (0,0)   | (0,0)   | (MINIMUM,TERRESTR) | (UNSECURE,MAXIMUM) | *
| 4              | 40        | LOW,LOW      | (0,95)     | 120            | 4      | 0,255 | 0,255 | 0,255 | 9600,MAXIMUM   | (0,0)   | (0,0)   | (MINIMUM,TERRESTR) | (UNSECURE,MAXIMUM) | *
| 5              | 60        | LOW,LOW      | (0,127)    | 150            | 5      | 0,255 | 0,255 | 0,255 | 19200,MAXIMUM | (0,0)   | (0,0)   | (MINIMUM,PACKET) | (UNSECURE,MAXIMUM) | *
| 6              | 80        | LOW,LOW      | (0,159)    | 180            | 6      | 0,255 | 0,255 | 0,255 | 9600,MAXIMUM   | (0,0)   | (0,0)   | (MINIMUM,PACKET) | (UNSECURE,MAXIMUM) | *

Appendix A. IBM-supplied tables

1327
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=6, node row number
WEIGHT=80, node row weight
CONGEST=(LOW,LOW), congestion
ROUTERES=(0,191), route addition resistance
LINEROW WEIGHT=210, line row weight
NUMBER=7, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(4800,MAXIMUM), LINE SPEED
COSTTIME=(0,196), cost per connect time
COSTBYTE=(0,196), cost per byte transmitted
PDELAY=(MINIMUM,MAXIMUM), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=7, node row number
WEIGHT=120, node row weight
CONGEST=(LOW,HIGH), congestion
ROUTERES=(0,223), route addition resistance
LINEROW WEIGHT=240, line row weight
NUMBER=8, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(MINIMUM,MAXIMUM), line speed
COSTTIME=(0,255), cost per connect time
COSTBYTE=(0,255), cost per byte transmitted
PDELAY=(MINIMUM,MAXIMUM), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=8, node row number
WEIGHT=160, node row weight
CONGEST=(LOW,HIGH), congestion
ROUTERES=(0,255), route addition resistance
* #CONNECT APPNCOS PRIORITY=MEDIUM,NUMBER=12 transmission priority A1R
LINEROW WEIGHT=20, line row weight
NUMBER=1, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(100M,MAXIMUM), line speed
COSTTIME=(0,64), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted
PDELAY=(MINIMUM,NEGLIGIB), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=1, node row number
WEIGHT=5, node row weight
CONGEST=(LOW,LOW), congestion
ROUTERES=(0,31), route addition resistance
LINEROW WEIGHT=40, line row weight
NUMBER=2, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(100M,MAXIMUM), line speed
COSTTIME=(0,64), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted
PDELAY=(MINIMUM,NEGLIGIB), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=2, node row number
WEIGHT=10, node row weight
CONGEST=(LOW,LOW), congestion
ROUTERES=(0,63), route addition resistance
LINEROW WEIGHT=60, line row weight
NUMBER=3, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3 *
CAPACITY=(4M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,NEGLIGIB), propagation delay *
SECURITY=(UNSECURE,MAXIMUM), security level for TG
NODEROW NUMBER=3, node row number *
WEIGHT=20, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,95), route addition resistance
LINEROW WEIGHT=80, line row weight *
NUMBER=4, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(10M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM), security level for TG
NODEROW NUMBER=4, node row number *
WEIGHT=40, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,127), route addition resistance
LINEROW WEIGHT=100, line row weight *
NUMBER=5, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(56000,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM), security level for TG
NODEROW NUMBER=5, node row number *
WEIGHT=60, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,159), route addition resistance
LINEROW WEIGHT=120, line row weight *
NUMBER=6, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(29K,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM), security level for TG
NODEROW NUMBER=6, node row number *
WEIGHT=80, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,191), route addition resistance
LINEROW WEIGHT=140, line row weight *
NUMBER=7, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(100M,MAXIMUM), line speed *
COSTTIME=(0,128), cost per connect time *
COSTBYTE=(0,128), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM), security level for TG
NODEROW NUMBER=7, node row number *
WEIGHT=120, node row weight *
CONGEST=(LOW,HIGH), congestion *
ROUTERES=(0,223), route addition resistance
LINEROW WEIGHT=160, line row weight *
<table>
<thead>
<tr>
<th>NODEROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGEST</th>
<th>ROUTERES</th>
<th>LNEROW WEIGHT</th>
<th>NUMBER</th>
<th>UPARM1</th>
<th>UPARM2</th>
<th>UPARM3</th>
<th>CAPACITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>160</td>
<td>LOW</td>
<td>(0,255)</td>
<td>180</td>
<td>9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(100M,MAX)</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>200</td>
<td>LOW</td>
<td>(0,255)</td>
<td>200</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(56000,MAX)</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>220</td>
<td>LOW</td>
<td>(0,255)</td>
<td>220</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(9600,MAX)</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>240</td>
<td>LOW</td>
<td>(0,255)</td>
<td>240</td>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(MINIMUM,MAX)</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>260</td>
<td>LOW</td>
<td>(0,255)</td>
<td>260</td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(MINIMUM,MAX)</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MDN1</td>
<td>UNSECURE</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

* #INTER APPNCS PRIORITY=HIGH,NUMBER=12 transmission priority
  LNEROW WEIGHT=20, line row weight*
  NUMBER=1, line row number*
  UPARM1=(0,255), user defined char 1*
  UPARM2=(0,255), user defined char 2*
  UPARM3=(0,255), user defined char 3*
  CAPACITY=(100M,MAX), line speed*
  COSTTIME=(0,64), cost per connect time*
  COSTBYTE=(0,0), cost per byte transmitted*
  PDELAY=(MINIMUM,NEGLIGIB), propagation delay*
  SECURITY=(UNSECURE,MAXIMUM), security level for TG*
NODEROW NUMBER=1, node row number*
  WEIGHT=5, node row weight*
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>40</td>
<td>(LOW, LOW)</td>
<td>(0, 31)</td>
<td>40</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(UNSECURE, MAXIMUM)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>60</td>
<td>(LOW, LOW)</td>
<td>(0, 63)</td>
<td>60</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(UNSECURE, MAXIMUM)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>80</td>
<td>(LOW, LOW)</td>
<td>(0, 95)</td>
<td>80</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(UNSECURE, MAXIMUM)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>100</td>
<td>(LOW, LOW)</td>
<td>(0, 127)</td>
<td>100</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(UNSECURE, MAXIMUM)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>120</td>
<td>(LOW, LOW)</td>
<td>(0, 159)</td>
<td>120</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(UNSECURE, MAXIMUM)</td>
<td></td>
</tr>
<tr>
<td>Node Row Number</td>
<td>Node Row Weight</td>
<td>Congestion</td>
<td>Route Addition Resistance</td>
<td>Line Row Weight</td>
<td>Line Row Number</td>
<td>User Defined Char 1</td>
<td>User Defined Char 2</td>
<td>User Defined Char 3</td>
<td>Line Speed</td>
<td>Cost Per Connect Time</td>
<td>Cost Per Byte Transmitted</td>
<td>Propagation Delay</td>
<td>Security Level</td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>------------</td>
<td>---------------------------</td>
<td>----------------</td>
<td>----------------</td>
<td>-------------------</td>
<td>-------------------</td>
<td>-------------------</td>
<td>------------</td>
<td>----------------------</td>
<td>--------------------------</td>
<td>-----------------</td>
<td>------------------</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>80</td>
<td>Low, Low</td>
<td>(0,191)</td>
<td>140</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Unsecure, Maximum)</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>120</td>
<td>Low, High</td>
<td>(0,223)</td>
<td>160</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Unsecure, Maximum)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>Low, High</td>
<td>(0,255)</td>
<td>180</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Unsecure, Maximum)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>200</td>
<td>Low, High</td>
<td>(0,255)</td>
<td>220</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Unsecure, Maximum)</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>240</td>
<td>Low, High</td>
<td>(0,255)</td>
<td>240</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Unsecure, Maximum)</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>280</td>
<td>Low, High</td>
<td>(0,255)</td>
<td>280</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Unsecure, Maximum)</td>
<td></td>
</tr>
<tr>
<td>UPARM3</td>
<td>CAPACITY</td>
<td>COSTTIME</td>
<td>COSTBYTE</td>
<td>PDELAY</td>
<td>SECURITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>----------</td>
<td>----------</td>
<td>----------</td>
<td>--------</td>
<td>----------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(0,255)</td>
<td>(MINIMUM,MAXIMUM)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(MINIMUM,MAXIMUM)</td>
<td>(UNSECURE,MAXIMUM)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* user defined char 3
* line speed
* cost per connect time
* cost per byte transmitted
* propagation delay

* security level for TG

#INTERSC APPNCOS  PRIORITY=HIGH,NUMBER=12
LINEROW  WEIGHT=20,  
NUMBER=1,  
UPARM1=(0,255),  
UPARM2=(0,255),  
UPARM3=(0,255),  
CAPACITY=(100M,MAXIMUM),  
COSTTIME=(0,64),  
COSTBYTE=(0,0),  
PDELAY=(MINIMUM,NEGLIGIB),  
SECURITY=(PUBLIC,MAXIMUM)

* transmission priority
* line row weight
* line row number
* user defined char 1
* user defined char 2
* user defined char 3
* line speed
* cost per connect time
* cost per byte transmitted
* propagation delay

* security level for TG

NODEROW  NUMBER=1,  
WEIGHT=5,  
CONGEST=(LOW,LOW),  
ROUTERES=(0,31)

LINEROW  WEIGHT=40,  
NUMBER=2,  
UPARM1=(0,255),  
UPARM2=(0,255),  
UPARM3=(0,255),  
CAPACITY=(10M,MAXIMUM),  
COSTTIME=(0,64),  
COSTBYTE=(0,0),  
PDELAY=(MINIMUM,NEGLIGIB),  
SECURITY=(PUBLIC,MAXIMUM)

* node row number
* node row weight
* congestion
* route addition resistance

LINEROW  WEIGHT=60,  
NUMBER=3,  
UPARM1=(0,255),  
UPARM2=(0,255),  
UPARM3=(0,255),  
CAPACITY=(1M,MAXIMUM),  
COSTTIME=(0,64),  
COSTBYTE=(0,0),  
PDELAY=(MINIMUM,NEGLIGIB),  
SECURITY=(PUBLIC,MAXIMUM)

* node row number
* node row weight
* congestion
* route addition resistance

LINEROW  WEIGHT=80,  
NUMBER=4,  
UPARM1=(0,255),  
UPARM2=(0,255),  
UPARM3=(0,255),  
CAPACITY=(100M,MAXIMUM),  
COSTTIME=(0,64),  
COSTBYTE=(0,0),  
PDELAY=(MINIMUM,TERRESTR),  
SECURITY=(PUBLIC,MAXIMUM)

* node row number
* node row weight
* congestion
* route addition resistance

Appendix A. IBM-supplied tables  1333
<table>
<thead>
<tr>
<th>NODEROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGEST</th>
<th>ROUTERES</th>
<th>LINEWEIGHT</th>
<th>NUMBER</th>
<th>UPARM1</th>
<th>UPARM2</th>
<th>UPARM3</th>
<th>CAPACITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>60</td>
<td>LOW,LOW</td>
<td>0,159</td>
<td>120</td>
<td>6</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(56000,MAXIMUM)</td>
<td>(0,128)</td>
<td>(0,128)</td>
<td>(MINIMUM,TERRESTR)</td>
<td>(PUBLIC,MAXIMUM)</td>
</tr>
<tr>
<td>6</td>
<td>80</td>
<td>LOW,LOW</td>
<td>0,191</td>
<td>140</td>
<td>7</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(1M,MAXIMUM)</td>
<td>(0,128)</td>
<td>(0,128)</td>
<td>(MINIMUM,TERRESTR)</td>
<td>(PUBLIC,MAXIMUM)</td>
</tr>
<tr>
<td>7</td>
<td>120</td>
<td>LOW,HIGH</td>
<td>0,223</td>
<td>160</td>
<td>8</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(29K,MAXIMUM)</td>
<td>(0,64)</td>
<td>(0,0)</td>
<td>(MINIMUM,TERRESTR)</td>
<td>(PUBLIC,MAXIMUM)</td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>LOW,HIGH</td>
<td>0,255</td>
<td>180</td>
<td>9</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(29K,MAXIMUM)</td>
<td>(0,64)</td>
<td>(0,0)</td>
<td>(MINIMUM,TERRESTR)</td>
<td>(PUBLIC,MAXIMUM)</td>
</tr>
<tr>
<td>9</td>
<td>180</td>
<td>LOW,HIGH</td>
<td>0,255</td>
<td>200</td>
<td>10</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(29K,MAXIMUM)</td>
<td>(0,64)</td>
<td>(0,0)</td>
<td>(MINIMUM,TERRESTR)</td>
<td>(PUBLIC,MAXIMUM)</td>
</tr>
<tr>
<td>Line</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>----------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td><strong>UPARM3=(0,255),</strong> user defined char 3 *</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td><strong>CAPACITY=(29K,MAXIMUM),</strong> line speed *</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td><strong>COSTTIME=(0,128),</strong> cost per connect time *</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td><strong>COSTBYTE=(0,128),</strong> cost per byte transmitted *</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td><strong>PDELAY=(MINIMUM,PACKET),</strong> propagation delay *</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| 6     | **SECURITY=(PUBLIC,MAXIMUM),** security level for TG *
| 7     | **LINEROW WEIGHT=220,** line row weight *         |
| 8     | **NUMBER=11,** line row number *                  |
| 9     | **UPARM1=(0,255),** user defined char 1 *         |
| 10    | **UPARM2=(0,255),** user defined char 2 *         |
| 11    | **UPARM3=(0,255),** user defined char 3 *         |
| 12    | **CAPACITY=(9600,MAXIMUM),** line speed *        |
| 13    | **COSTTIME=(0,64),** cost per connect time *      |
| 14    | **COSTBYTE=(0,0),** cost per byte transmitted *  |
| 15    | **PDELAY=(MINIMUM,PACKET),** propagation delay * |
| 16    | **SECURITY=(PUBLIC,MAXIMUM),** security level for TG *
| 17    | **LINEROW WEIGHT=240,** line row weight *         |
| 18    | **NUMBER=12,** line row number *                  |
| 19    | **UPARM1=(0,255),** user defined char 1 *         |
| 20    | **UPARM2=(0,255),** user defined char 2 *         |
| 21    | **UPARM3=(0,255),** user defined char 3 *         |
| 22    | **CAPACITY=(MINIMUM,MAXIMUM),** line speed *     |
| 23    | **COSTTIME=(0,255),** cost per connect time *     |
| 24    | **COSTBYTE=(0,255),** cost per byte transmitted *|
| 25    | **PDELAY=(MINIMUM,MAXIMUM),** propagation delay *|
| 26    | **SECURITY=(PUBLIC,MAXIMUM),** security level for TG *
| 27    | **#BATCH APPNCOS PRIORITY=LOW,NUMBER=12** transmission priority |
| 28    | **LINEROW WEIGHT=20,** line row weight *          |
| 29    | **NUMBER=1,** line row number *                   |
| 30    | **UPARM1=(0,255),** user defined char 1 *         |
| 31    | **UPARM2=(0,255),** user defined char 2 *         |
| 32    | **UPARM3=(0,255),** user defined char 3 *         |
| 33    | **CAPACITY=(100M,MAXIMUM),** line speed *        |
| 34    | **COSTTIME=(0,0),** cost per connect time *       |
| 35    | **COSTBYTE=(0,0),** cost per byte transmitted *  |
| 36    | **PDELAY=(MINIMUM,MAXIMUM),** propagation delay *|
| 37    | **SECURITY=(PUBLIC,MAXIMUM),** security level for TG *
| 38    | **NODEROW NUMBER=1,** node row number *           |
| 39    | **WEIGHT=5,** node row weight *                   |
| 40    | **CONGEST=(LOW,LOW),** congestion *              |
| 41    | **ROUTERES=(0,31),** route addition resistance * |
| 42    | **LINEROW WEIGHT=40,** line row weight *          |
| 43    | **NUMBER=2,** line row number *                   |
| 44    | **UPARM1=(0,255),** user defined char 1 *         |
| 45    | **UPARM2=(0,255),** user defined char 2 *         |
| 46    | **UPARM3=(0,255),** user defined char 3 *         |
| 47    | **CAPACITY=(10M,MAXIMUM),** line speed *         |
| 48    | **COSTTIME=(0,0),** cost per connect time *       |
| 49    | **COSTBYTE=(0,0),** cost per byte transmitted *  |
| 50    | **PDELAY=(MINIMUM,MAXIMUM),** propagation delay *|
| 51    | **SECURITY=(PUBLIC,MAXIMUM),** security level for TG *
| 52    | **NODEROW NUMBER=2,** node row number *           |
| 53    | **WEIGHT=10,** node row weight *                  |
| 54    | **CONGEST=(LOW,LOW),** congestion *              |
| 55    | **ROUTERES=(0,63),** route addition resistance * |
| 56    | **LINEROW WEIGHT=60,** line row weight *          |
| 57    | **NUMBER=3,** line row number *                   |
| 58    | **UPARM1=(0,255),** user defined char 1 *         |
| 59    | **UPARM2=(0,255),** user defined char 2 *         |
| 60    | **UPARM3=(0,255),** user defined char 3 *         |
| 61    | **CAPACITY=(10M,MAXIMUM),** line speed *         |
| 62    | **COSTTIME=(0,64),** cost per connect time *      |
| 63    | **COSTBYTE=(0,0),** cost per byte transmitted *  |
| 64    | **PDELAY=(MINIMUM,MAXIMUM),** propagation delay *|
| 65    | **SECURITY=(PUBLIC,MAXIMUM),** security level for TG *
<p>| 66    | <strong>NODEROW NUMBER=3,</strong> node row number *           |</p>
<table>
<thead>
<tr>
<th>Node Row Number</th>
<th>Line Row Weight</th>
<th>Line Row Number</th>
<th>User Defined Char 1</th>
<th>User Defined Char 2</th>
<th>User Defined Char 3</th>
<th>Line Speed</th>
<th>Cost per Connect Time</th>
<th>Cost per Byte Transmitted</th>
<th>Propagation Delay</th>
<th>Security Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>180</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UNSECURE</td>
</tr>
<tr>
<td>10</td>
<td>200</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UNSECURE</td>
</tr>
<tr>
<td>11</td>
<td>220</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UNSECURE</td>
</tr>
<tr>
<td>12</td>
<td>240</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UNSECURE</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LOW</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PUBLIC</td>
</tr>
</tbody>
</table>

Appendix A: IBM-supplied tables

1337
<table>
<thead>
<tr>
<th>Noderow Number</th>
<th>Node Row Weight</th>
<th>Congestion</th>
<th>Route Addition Resistance</th>
<th>Line Row Weight</th>
<th>Line Row Number</th>
<th>User Defined Char 1</th>
<th>User Defined Char 2</th>
<th>User Defined Char 3</th>
<th>Line Speed</th>
<th>Cost Per Connect Time</th>
<th>Cost Per Byte Transmitted</th>
<th>Propagation Delay</th>
<th>Security Level for TG</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>120</td>
<td>LOW, HIGH</td>
<td>(0,223)</td>
<td>160</td>
<td>8</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>100M, MAXIMUM</td>
<td>(0,64)</td>
<td>(0,128)</td>
<td>(MINIMUM, MAXIMUM)</td>
<td>(PUBLIC, MAXIMUM)</td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>LOW, HIGH</td>
<td>(0,255)</td>
<td>180</td>
<td>9</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>56000, MAXIMUM</td>
<td>(0,128)</td>
<td>(0,128)</td>
<td>(MINIMUM, MAXIMUM)</td>
<td>(PUBLIC, MAXIMUM)</td>
</tr>
<tr>
<td>9</td>
<td>180</td>
<td>LOW, HIGH</td>
<td>(0,255)</td>
<td>200</td>
<td>10</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>29K, MAXIMUM</td>
<td>(0,128)</td>
<td>(0,128)</td>
<td>(MINIMUM, MAXIMUM)</td>
<td>(PUBLIC, MAXIMUM)</td>
</tr>
<tr>
<td>10</td>
<td>200</td>
<td>LOW, HIGH</td>
<td>(0,255)</td>
<td>220</td>
<td>11</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>9600, MAXIMUM</td>
<td>(0,64)</td>
<td>(0,0)</td>
<td>(MINIMUM, MAXIMUM)</td>
<td>(PUBLIC, MAXIMUM)</td>
</tr>
<tr>
<td>11</td>
<td>220</td>
<td>LOW, HIGH</td>
<td>(0,255)</td>
<td>240</td>
<td>12</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>MINIMUM, MAXIMUM</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(MINIMUM, MAXIMUM)</td>
<td>(PUBLIC, MAXIMUM)</td>
</tr>
</tbody>
</table>
ISTACST3

CPSVC MG APPNC OS PRIORITY=NETWORK, NUMBER=12 transmission priority
LINEROW WEIGHT=20, line row weight *
NUMBER=1, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(10G,MAXIMUM), line speed *
COSTTIME=(0,0), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,NEGLIGIB), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=1, node row number *
WEIGHT=5, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,31) route addition resistance

LINEROW WEIGHT=40, line row weight *
NUMBER=2, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1G,MAXIMUM), line speed *
COSTTIME=(0,0), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,NEGLIGIB), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=2, node row number *
WEIGHT=10, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,63) route addition resistance

LINEROW WEIGHT=60, line row weight *
NUMBER=3, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(100M,MAXIMUM), line speed *
COSTTIME=(0,0), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,NEGLIGIB), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=3, node row number *
WEIGHT=20, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,95) route addition resistance

LINEROW WEIGHT=80, line row weight *
NUMBER=4, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1G,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=4, node row number *
WEIGHT=40, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,127) route addition resistance

LINEROW WEIGHT=100, line row weight *
NUMBER=5, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(100M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
<table>
<thead>
<tr>
<th>NODEROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGEST</th>
<th>ROUTERES</th>
<th>LINEROW WEIGHT</th>
<th>NUMBER</th>
<th>UPARM1</th>
<th>UPARM2</th>
<th>UPARM3</th>
<th>CAPACITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
<th>NODEROW NUMBER</th>
<th>WEIGHT</th>
<th>CONGEST</th>
<th>ROUTERES</th>
<th>LINEROW WEIGHT</th>
<th>NUMBER</th>
<th>UPARM1</th>
<th>UPARM2</th>
<th>UPARM3</th>
<th>CAPACITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>60</td>
<td>LOW,LOW</td>
<td>(0,159)</td>
<td>120</td>
<td>6</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(25M,MAX)</td>
<td>(0,64)</td>
<td>(0,0)</td>
<td>(MINIMUM,TERRESTR),</td>
<td>(UNSECURE,MAXIMUM)</td>
<td>6</td>
<td>80</td>
<td>LOW,LOW</td>
<td>(0,191)</td>
<td>140</td>
<td>7</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(10G,MAX)</td>
<td>(0,128)</td>
<td>(0,128)</td>
<td>(MINIMUM,TERRESTR),</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>6</td>
<td>120</td>
<td>LOW,LOW</td>
<td>(0,223)</td>
<td>160</td>
<td>8</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(1M,MAX)</td>
<td>(0,64)</td>
<td>(0,0)</td>
<td>(MINIMUM,TERRESTR),</td>
<td>(UNSECURE,MAXIMUM)</td>
<td>7</td>
<td>160</td>
<td>LOW,LOW</td>
<td>(0,255)</td>
<td>180</td>
<td>9</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(16M,MAX)</td>
<td>(0,64)</td>
<td>(0,0)</td>
<td>(MINIMUM,PACKET),</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>LOW,LOW</td>
<td>(0,255)</td>
<td>200</td>
<td>10</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(1M,MAX)</td>
<td>(0,64)</td>
<td>(0,0)</td>
<td>(MINIMUM,PACKET),</td>
<td>(UNSECURE,MAXIMUM)</td>
<td>9</td>
<td>160</td>
<td>LOW,LOW</td>
<td>(0,255)</td>
<td>220</td>
<td>11</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(0,255)</td>
<td>(16M,MAX)</td>
<td>(0,64)</td>
<td>(0,0)</td>
<td>(MINIMUM,PACKET),</td>
<td>(UNSECURE,MAXIMUM)</td>
</tr>
</tbody>
</table>
SECURITY=(UNSECURE,MAXIMUM) security level for TG
LINEROW WEIGHT=220, line row weight
NUMBER=11, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(64K,MAXIMUM), line speed
COSTTIME=(0,196), cost per connect time
COSTBYTE=(0,196), cost per byte transmitted
PDELAY=(MINIMUM,MAXIMUM), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG
LINEROW WEIGHT=240, line row weight
NUMBER=12, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(MINIMUM,MAXIMUM), line speed
COSTTIME=(0,255), cost per connect time
COSTBYTE=(0,255), cost per byte transmitted
PDELAY=(MINIMUM,MAXIMUM), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG

SNASVC Speaker MSG APPNCOS PRIORITY=NETWORK,NUMBER=12 transmission priority
LINEROW WEIGHT=20, line row weight
NUMBER=1, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(10G,MAXIMUM), line speed
COSTTIME=(0,0), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted
PDELAY=(MINIMUM,NEGLIGIB), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=1, node row number
WEIGHT=5, node row weight
CONGEST=(LOW,LOW), congestion
ROUTERES=(0,31), route addition resistance
LINEROW WEIGHT=40, line row weight
NUMBER=2, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(1G,MAXIMUM), line speed
COSTTIME=(0,0), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted
PDELAY=(MINIMUM,NEGLIGIB), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=2, node row number
WEIGHT=10, node row weight
CONGEST=(LOW,LOW), congestion
ROUTERES=(0,63), route addition resistance
LINEROW WEIGHT=60, line row weight
NUMBER=3, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(100M,MAXIMUM), line speed
COSTTIME=(0,0), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted
PDELAY=(MINIMUM,NEGLIGIB), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG
NODEROW NUMBER=3, node row number
WEIGHT=20, node row weight
CONGEST=(LOW,LOW), congestion
ROUTERES=(0,95), route addition resistance
LINEROW WEIGHT=80, line row weight
NUMBER=4, line row number
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1G,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=4, node row number *
WEIGHT=40, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,127) route addition resistance
LINEROW WEIGHT=100, line row weight *
NUMBER=5, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(100M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=5, node row number *
WEIGHT=60, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,159) route addition resistance
LINEROW WEIGHT=120, line row weight *
NUMBER=6, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(25M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=6, node row number *
WEIGHT=80, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,191) route addition resistance
LINEROW WEIGHT=140, line row weight *
NUMBER=7, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(10G,MAXIMUM), line speed *
COSTTIME=(0,128), cost per connect time *
COSTBYTE=(0,128), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=7, node row number *
WEIGHT=120, node row weight *
CONGEST=(LOW,HIGH), congestion *
ROUTERES=(0,223) route addition resistance
LINEROW WEIGHT=160, line row weight *
NUMBER=8, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1M,MAXIMUM), line speed *
COSTTIME=(0,128), cost per connect time *
COSTBYTE=(0,128), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=8, node row number *
WEIGHT=160, node row weight *
CONGEST=(LOW,HIGH), congestion *

Appendix A. IBM-supplied tables 1343
<table>
<thead>
<tr>
<th>ROUTERES=(0,255)</th>
<th>route addition resistance</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINEROW WEIGHT=180,</td>
<td>line row weight</td>
</tr>
<tr>
<td>NUMBER=9,</td>
<td>line row number</td>
</tr>
<tr>
<td>UPARM1=(0,255),</td>
<td>user defined char 1</td>
</tr>
<tr>
<td>UPARM2=(0,255),</td>
<td>user defined char 2</td>
</tr>
<tr>
<td>UPARM3=(0,255),</td>
<td>user defined char 3</td>
</tr>
<tr>
<td>CAPACITY=(16M,MAXIMUM),</td>
<td>line speed</td>
</tr>
<tr>
<td>COSTTIME=(0,64),</td>
<td>cost per connect time</td>
</tr>
<tr>
<td>COSTBYTE=(0,0),</td>
<td>cost per byte transmitted</td>
</tr>
<tr>
<td>PDELAY=(MINIMUM,PACKET),</td>
<td>propagation delay</td>
</tr>
<tr>
<td>SECURITY=(UNSECURE,MAXIMUM)</td>
<td>security level for TG</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROUTERES=(0,255)</th>
<th>route addition resistance</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINEROW WEIGHT=200,</td>
<td>line row weight</td>
</tr>
<tr>
<td>NUMBER=10,</td>
<td>line row number</td>
</tr>
<tr>
<td>UPARM1=(0,255),</td>
<td>user defined char 1</td>
</tr>
<tr>
<td>UPARM2=(0,255),</td>
<td>user defined char 2</td>
</tr>
<tr>
<td>UPARM3=(0,255),</td>
<td>user defined char 3</td>
</tr>
<tr>
<td>CAPACITY=(1M,MAXIMUM),</td>
<td>line speed</td>
</tr>
<tr>
<td>COSTTIME=(0,128),</td>
<td>cost per connect time</td>
</tr>
<tr>
<td>COSTBYTE=(0,128),</td>
<td>cost per byte transmitted</td>
</tr>
<tr>
<td>PDELAY=(MINIMUM,PACKET),</td>
<td>propagation delay</td>
</tr>
<tr>
<td>SECURITY=(UNSECURE,MAXIMUM)</td>
<td>security level for TG</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROUTERES=(0,255)</th>
<th>route addition resistance</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINEROW WEIGHT=220,</td>
<td>line row weight</td>
</tr>
<tr>
<td>NUMBER=11,</td>
<td>line row number</td>
</tr>
<tr>
<td>UPARM1=(0,255),</td>
<td>user defined char 1</td>
</tr>
<tr>
<td>UPARM2=(0,255),</td>
<td>user defined char 2</td>
</tr>
<tr>
<td>UPARM3=(0,255),</td>
<td>user defined char 3</td>
</tr>
<tr>
<td>CAPACITY=(64K,MAXIMUM),</td>
<td>line speed</td>
</tr>
<tr>
<td>COSTTIME=(0,196),</td>
<td>cost per connect time</td>
</tr>
<tr>
<td>COSTBYTE=(0,196),</td>
<td>cost per byte transmitted</td>
</tr>
<tr>
<td>PDELAY=(MINIMUM,MAXIMUM),</td>
<td>propagation delay</td>
</tr>
<tr>
<td>SECURITY=(UNSECURE,MAXIMUM)</td>
<td>security level for TG</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROUTERES=(0,255)</th>
<th>route addition resistance</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINEROW WEIGHT=240,</td>
<td>line row weight</td>
</tr>
<tr>
<td>NUMBER=12,</td>
<td>line row number</td>
</tr>
<tr>
<td>UPARM1=(0,255),</td>
<td>user defined char 1</td>
</tr>
<tr>
<td>UPARM2=(0,255),</td>
<td>user defined char 2</td>
</tr>
<tr>
<td>UPARM3=(0,255),</td>
<td>user defined char 3</td>
</tr>
<tr>
<td>CAPACITY=(MINIMUM,MAXIMUM),</td>
<td>line speed</td>
</tr>
<tr>
<td>COSTTIME=(0,255),</td>
<td>cost per connect time</td>
</tr>
<tr>
<td>COSTBYTE=(0,255),</td>
<td>cost per byte transmitted</td>
</tr>
<tr>
<td>PDELAY=(MINIMUM,MAXIMUM),</td>
<td>propagation delay</td>
</tr>
<tr>
<td>SECURITY=(UNSECURE,MAXIMUM)</td>
<td>security level for TG</td>
</tr>
</tbody>
</table>

* #CONNECT APPNCOS PRIORITY=MEDIUM,NUMBER=12 transmission priority

<table>
<thead>
<tr>
<th>ROUTERES=(0,255)</th>
<th>route addition resistance</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINEROW WEIGHT=20,</td>
<td>line row weight</td>
</tr>
<tr>
<td>NUMBER=1,</td>
<td>line row number</td>
</tr>
<tr>
<td>UPARM1=(0,255),</td>
<td>user defined char 1</td>
</tr>
<tr>
<td>UPARM2=(0,255),</td>
<td>user defined char 2</td>
</tr>
<tr>
<td>UPARM3=(0,255),</td>
<td>user defined char 3</td>
</tr>
<tr>
<td>CAPACITY=(1G,MAXIMUM),</td>
<td>line speed</td>
</tr>
<tr>
<td>COSTTIME=(0,64),</td>
<td>cost per connect time</td>
</tr>
<tr>
<td>COSTBYTE=(0,0),</td>
<td>cost per byte transmitted</td>
</tr>
<tr>
<td>PDELAY=(MINIMUM,NEGligIB),</td>
<td>propagation delay</td>
</tr>
<tr>
<td>SECURITY=(UNSECURE,MAXIMUM)</td>
<td>security level for TG</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROUTERES=(0,31)</th>
<th>route addition resistance</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINEROW WEIGHT=40,</td>
<td>line row weight</td>
</tr>
<tr>
<td>NUMBER=2,</td>
<td>line row number</td>
</tr>
<tr>
<td>UPARM1=(0,255),</td>
<td>user defined char 1</td>
</tr>
<tr>
<td>UPARM2=(0,255),</td>
<td>user defined char 2</td>
</tr>
<tr>
<td>UPARM3=(0,255),</td>
<td>user defined char 3</td>
</tr>
<tr>
<td>CAPACITY=(1G,MAXIMUM),</td>
<td>line speed</td>
</tr>
<tr>
<td>COSTTIME=(0,64),</td>
<td>cost per connect time</td>
</tr>
<tr>
<td>COSTBYTE=(0,0),</td>
<td>cost per byte transmitted</td>
</tr>
<tr>
<td>PDELAY=(MINIMUM,NEGligIB),</td>
<td>propagation delay</td>
</tr>
<tr>
<td>SECURITY=(UNSECURE,MAXIMUM)</td>
<td>security level for TG</td>
</tr>
</tbody>
</table>
COSTBYTE=(0,128), cost per byte transmitted
PDELAY=(MINIMUM,TERRESTR), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=7, node row number
WEIGHT=120, node row weight
CONGEST=(LOW,HIGH), congestion
ROUTERES=(0,223) route addition resistance

LINEROW WEIGHT=160, line row weight
NUMBER=8, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(1G,MAXIMUM), line speed
COSTTIME=(0,128), cost per connect time
COSTBYTE=(0,128), cost per byte transmitted
PDELAY=(MINIMUM,TERRESTR), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=8, node row number
WEIGHT=160, node row weight
CONGEST=(LOW,HIGH), congestion
ROUTERES=(0,255) route addition resistance

LINEROW WEIGHT=180, line row weight
NUMBER=9, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(16M,MAXIMUM), line speed
COSTTIME=(0,64), cost per connect time
COSTBYTE=(0,128), cost per byte transmitted
PDELAY=(MINIMUM,PACKET), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG

LINEROW WEIGHT=200, line row weight
NUMBER=10, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(1M,MAXIMUM), line speed
COSTTIME=(0,128), cost per connect time
COSTBYTE=(0,128), cost per byte transmitted
PDELAY=(MINIMUM,PACKET), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG

LINEROW WEIGHT=220, line row weight
NUMBER=11, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(64K,MAXIMUM), line speed
COSTTIME=(0,64), cost per connect time
COSTBYTE=(0,0), cost per byte transmitted
PDELAY=(MINIMUM,PACKET), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG

LINEROW WEIGHT=240, line row weight
NUMBER=12, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2
UPARM3=(0,255), user defined char 3
CAPACITY=(MINIMUM,MAXIMUM), line speed
COSTTIME=(0,255), cost per connect time
COSTBYTE=(0,255), cost per byte transmitted
PDELAY=(MINIMUM,MAXIMUM), propagation delay
SECURITY=(UNSECURE,MAXIMUM) security level for TG

* #INTER APPNCS PRIORITY=HIGH,NUMBER=12 transmission priority
LINEROW WEIGHT=20, line row weight
NUMBER=1, line row number
UPARM1=(0,255), user defined char 1
UPARM2=(0,255), user defined char 2

1346 z/OS V2R1.0 Communications Server: SNA Resource Definition Reference
<table>
<thead>
<tr>
<th>Node Row Number</th>
<th>Node Row Weight</th>
<th>Congestion</th>
<th>Route Addition Resistance</th>
<th>Line Row Weight</th>
<th>Line Row Number</th>
<th>User Defined Char 1</th>
<th>User Defined Char 2</th>
<th>User Defined Char 3</th>
<th>Line Capacity</th>
<th>Line Speed</th>
<th>Cost Per Connect Time</th>
<th>Cost Per Byte Transmitted</th>
<th>Propagation Delay</th>
<th>Security Level for TG</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>Low, Low</td>
<td>(0.31)</td>
<td>40</td>
<td>2</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>10G, Maximum</td>
<td>10G</td>
<td>0, 64</td>
<td>0, 0</td>
<td>Minimum, Negligible</td>
<td>Unsecure, Maximum</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>Low, Low</td>
<td>(0.63)</td>
<td>60</td>
<td>3</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>1G, Maximum</td>
<td>1G</td>
<td>0, 64</td>
<td>0, 0</td>
<td>Minimum, Negligible</td>
<td>Unsecure, Maximum</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>Low, Low</td>
<td>(0.99)</td>
<td>80</td>
<td>4</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>100M, Maximum</td>
<td>100M</td>
<td>0, 64</td>
<td>0, 0</td>
<td>Minimum, Terrestrial</td>
<td>Unsecure, Maximum</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>Low, Low</td>
<td>(1.27)</td>
<td>100</td>
<td>5</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>1G, Maximum</td>
<td>1G</td>
<td>0, 64</td>
<td>0, 0</td>
<td>Minimum, Terrestrial</td>
<td>Unsecure, Maximum</td>
</tr>
<tr>
<td>5</td>
<td>60</td>
<td>Low, Low</td>
<td>(1.59)</td>
<td>120</td>
<td>6</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>100M, Maximum</td>
<td>100M</td>
<td>0, 64</td>
<td>0, 0</td>
<td>Minimum, Terrestrial</td>
<td>Unsecure, Maximum</td>
</tr>
</tbody>
</table>
NUMBER=6, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(25M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=6, node row number *
WEIGHT=80, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,191) route addition resistance *
LINEROW WEIGHT=140, line row weight *
NUMBER=7, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(100M,MAXIMUM), line speed *
COSTTIME=(0,128), cost per connect time *
COSTBYTE=(0,128), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=7, node row number *
WEIGHT=120, node row weight *
CONGEST=(LOW,HIGH), congestion *
ROUTERES=(0,223) route addition resistance *
LINEROW WEIGHT=160, line row weight *
NUMBER=8, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(16M,MAXIMUM), line speed *
COSTTIME=(0,128), cost per connect time *
COSTBYTE=(0,128), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=8, node row number *
WEIGHT=160, node row weight *
CONGEST=(LOW,HIGH), congestion *
ROUTERES=(0,255) route addition resistance *
LINEROW WEIGHT=180, line row weight *
NUMBER=9, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTR), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=9, node row number *
WEIGHT=200, node row weight *
CONGEST=(LOW,HIGH), congestion *
ROUTERES=(0,255) route addition resistance *
LINEROW WEIGHT=200, line row weight *
NUMBER=10, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1M,MAXIMUM), line speed *
COSTTIME=(0,128), cost per connect time *
COSTBYTE=(0,128), cost per byte transmitted *
PDELAY=(MINIMUM,PACKET), propagation delay *
SECURITY=(UNSECURE,MAXIMUM) security level for TG

NODEROW NUMBER=10, node row number *
WEIGHT=220, node row weight *
CONGEST=(LOW,HIGH), congestion *
ROUTERES=(0,255) route addition resistance *
LINEROW WEIGHT=220, line row weight *
NUMBER=11, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(64K,MAXIMUM), line speed
<table>
<thead>
<tr>
<th>Line</th>
<th>Weight</th>
<th>Number</th>
<th>User Defined Char 1</th>
<th>User Defined Char 2</th>
<th>User Defined Char 3</th>
<th>Capacity</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Propagation Delay</th>
<th>Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>40</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>60</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>80</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Appendix A. IBM-supplied tables*
SECURITY=(PUBLIC,MAXIMUM) security level for TG
NODEROW NUMBER=4, node row number *
  WEIGHT=40, node row weight *
  CONGEST=(LOW, LOW), congestion *
  ROUTERES=(0, 127), route addition resistance *
LINEROW WEIGHT=100, line row weight *
  NUMBER=5, line row number *
  UPARM1=(0, 255), user defined char 1 *
  UPARM2=(0, 255), user defined char 2 *
  UPARM3=(0, 255), user defined char 3 *
  CAPACITY=(100M, MAXIMUM), line speed *
  COSTTIME=(0, 64), cost per connect time *
  COSTBYTE=(0, 0), cost per byte transmitted *
  PDELAY=(MINIMUM, TERRESTR), propagation delay *
SECURITY=(PUBLIC, MAXIMUM) security level for TG
NODEROW NUMBER=5, node row number *
  WEIGHT=60, node row weight *
  CONGEST=(LOW, LOW), congestion *
  ROUTERES=(0, 159), route addition resistance *
LINEROW WEIGHT=120, line row weight *
  NUMBER=6, line row number *
  UPARM1=(0, 255), user defined char 1 *
  UPARM2=(0, 255), user defined char 2 *
  UPARM3=(0, 255), user defined char 3 *
  CAPACITY=(25M, MAXIMUM), line speed *
  COSTTIME=(0, 64), cost per connect time *
  COSTBYTE=(0, 0), cost per byte transmitted *
  PDELAY=(MINIMUM, TERRESTR), propagation delay *
SECURITY=(PUBLIC, MAXIMUM) security level for TG
NODEROW NUMBER=6, node row number *
  WEIGHT=80, node row weight *
  CONGEST=(LOW, LOW), congestion *
  ROUTERES=(0, 191), route addition resistance *
LINEROW WEIGHT=140, line row weight *
  NUMBER=7, line row number *
  UPARM1=(0, 255), user defined char 1 *
  UPARM2=(0, 255), user defined char 2 *
  UPARM3=(0, 255), user defined char 3 *
  CAPACITY=(100M, MAXIMUM), line speed *
  COSTTIME=(0, 128), cost per connect time *
  COSTBYTE=(0, 128), cost per byte transmitted *
  PDELAY=(MINIMUM, TERRESTR), propagation delay *
SECURITY=(PUBLIC, MAXIMUM) security level for TG
NODEROW NUMBER=7, node row number *
  WEIGHT=120, node row weight *
  CONGEST=(LOW, HIGH), congestion *
  ROUTERES=(0, 223), route addition resistance *
LINEROW WEIGHT=160, line row weight *
  NUMBER=8, line row number *
  UPARM1=(0, 255), user defined char 1 *
  UPARM2=(0, 255), user defined char 2 *
  UPARM3=(0, 255), user defined char 3 *
  CAPACITY=(16M, MAXIMUM), line speed *
  COSTTIME=(0, 128), cost per connect time *
  COSTBYTE=(0, 128), cost per byte transmitted *
  PDELAY=(MINIMUM, TERRESTR), propagation delay *
SECURITY=(PUBLIC, MAXIMUM) security level for TG
NODEROW NUMBER=8, node row number *
  WEIGHT=160, node row weight *
  CONGEST=(LOW, HIGH), congestion *
  ROUTERES=(0, 255), route addition resistance *
LINEROW WEIGHT=180, line row weight *
  NUMBER=9, line row number *
  UPARM1=(0, 255), user defined char 1 *
  UPARM2=(0, 255), user defined char 2 *
  UPARM3=(0, 255), user defined char 3 *
  CAPACITY=(1M, MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,TERRESTRF), propagation delay *
SECURITY=(PUBLIC,MAXIMUM), security level for TG
LINEROW WEIGHT=200, line row weight *
NUMBER=10, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1M,MAXIMUM), line speed *
COSTTIME=(0,128), cost per connect time *
COSTBYTE=(0,128), cost per byte transmitted *
PDELAY=(MINIMUM,PACKET), propagation delay *
SECURITY=(PUBLIC,MAXIMUM), security level for TG
LINEROW WEIGHT=220, line row weight *
NUMBER=11, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(64K,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,PACKET), propagation delay *
SECURITY=(PUBLIC,MAXIMUM), security level for TG
LINEROW WEIGHT=240, line row weight *
NUMBER=12, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(10G,MAXIMUM), line speed *
COSTTIME=(0,255), cost per connect time *
COSTBYTE=(0,255), cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(UNSECURE,MAXIMUM), security level for TG
* #BATCH APPNCOS PRIORITY=LOW,NUMBER=12 transmission priority
LINEROW WEIGHT=20, line row weight *
NUMBER=1, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1G,MAXIMUM), line speed *
COSTTIME=(0,0), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(UNSECURE,MAXIMUM), security level for TG
NODEROW NUMBER=1, node row number *
WEIGHT=5, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,31), route addition resistance
LINEROW WEIGHT=40, line row weight *
NUMBER=2, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1G,MAXIMUM), line speed *
COSTTIME=(0,0), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(UNSECURE,MAXIMUM), security level for TG
NODEROW NUMBER=2, node row number *
WEIGHT=10, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,63), route addition resistance
LINEROW WEIGHT=60, line row weight *
NUMBER=3, line row number *
UPARM1=(0,255), user defined char 1 *
<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LINEROW WEIGHT=160, line row weight</td>
</tr>
<tr>
<td>2</td>
<td>NUMBER=8, line row number</td>
</tr>
<tr>
<td>3</td>
<td>UPARM1=(0,255), user defined char 1</td>
</tr>
<tr>
<td>4</td>
<td>UPARM2=(0,255), user defined char 2</td>
</tr>
<tr>
<td>5</td>
<td>UPARM3=(0,255), user defined char 3</td>
</tr>
<tr>
<td>6</td>
<td>CAPACITY=(1G,MAXIMUM), line speed</td>
</tr>
<tr>
<td>7</td>
<td>COSTTIME=(0,64), cost per connect time</td>
</tr>
<tr>
<td>8</td>
<td>COSTBYTE=(0,128), cost per byte transmitted</td>
</tr>
<tr>
<td>9</td>
<td>PDELAY=(MINIMUM,MAXIMUM), propagation delay</td>
</tr>
<tr>
<td>10</td>
<td>SECURITY=(UNSECURE,MAXIMUM), security level for TG</td>
</tr>
<tr>
<td>11</td>
<td>NODEROW NUMBER=8, node row number</td>
</tr>
<tr>
<td>12</td>
<td>WEIGHT=160, node row weight</td>
</tr>
<tr>
<td>13</td>
<td>CONGEST=(LOW,HIGH), congestion</td>
</tr>
<tr>
<td>14</td>
<td>ROUTERES=(0,255), route addition resistance</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>LINEROW WEIGHT=180, line row weight</td>
</tr>
<tr>
<td>18</td>
<td>NUMBER=9, line row number</td>
</tr>
<tr>
<td>19</td>
<td>UPARM1=(0,255), user defined char 1</td>
</tr>
<tr>
<td>20</td>
<td>UPARM2=(0,255), user defined char 2</td>
</tr>
<tr>
<td>21</td>
<td>UPARM3=(0,255), user defined char 3</td>
</tr>
<tr>
<td>22</td>
<td>CAPACITY=(16M,MAXIMUM), line speed</td>
</tr>
<tr>
<td>23</td>
<td>COSTTIME=(0,128), cost per connect time</td>
</tr>
<tr>
<td>24</td>
<td>COSTBYTE=(0,128), cost per byte transmitted</td>
</tr>
<tr>
<td>25</td>
<td>PDELAY=(MINIMUM,MAXIMUM), propagation delay</td>
</tr>
<tr>
<td>26</td>
<td>SECURITY=(UNSECURE,MAXIMUM), security level for TG</td>
</tr>
<tr>
<td>27</td>
<td>NODEROW NUMBER=10, node row number</td>
</tr>
<tr>
<td>28</td>
<td>WEIGHT=200, node row weight</td>
</tr>
<tr>
<td>29</td>
<td>CONGEST=(LOW, HIGH), congestion</td>
</tr>
<tr>
<td>30</td>
<td>ROUTERES=(0,255), route addition resistance</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>LINEROW WEIGHT=220, line row weight</td>
</tr>
<tr>
<td>34</td>
<td>NUMBER=11, line row number</td>
</tr>
<tr>
<td>35</td>
<td>UPARM1=(0,255), user defined char 1</td>
</tr>
<tr>
<td>36</td>
<td>UPARM2=(0,255), user defined char 2</td>
</tr>
<tr>
<td>37</td>
<td>UPARM3=(0,255), user defined char 3</td>
</tr>
<tr>
<td>38</td>
<td>CAPACITY=(64K,MAXIMUM), line speed</td>
</tr>
<tr>
<td>39</td>
<td>COSTTIME=(0,128), cost per connect time</td>
</tr>
<tr>
<td>40</td>
<td>COSTBYTE=(0,128), cost per byte transmitted</td>
</tr>
<tr>
<td>41</td>
<td>PDELAY=(MINIMUM,MAXIMUM), propagation delay</td>
</tr>
<tr>
<td>42</td>
<td>SECURITY=(UNSECURE,MAXIMUM), security level for TG</td>
</tr>
<tr>
<td>43</td>
<td>NODEROW NUMBER=12, node row number</td>
</tr>
<tr>
<td>44</td>
<td>WEIGHT=240, node row weight</td>
</tr>
<tr>
<td>45</td>
<td>CONGEST=(LOW, HIGH), congestion</td>
</tr>
<tr>
<td>46</td>
<td>ROUTERES=(0,255), route addition resistance</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>49</td>
<td>LINEROW WEIGHT=260, line row weight</td>
</tr>
<tr>
<td>50</td>
<td>NUMBER=13, line row number</td>
</tr>
<tr>
<td>51</td>
<td>UPARM1=(0,255), user defined char 1</td>
</tr>
<tr>
<td>52</td>
<td>UPARM2=(0,255), user defined char 2</td>
</tr>
<tr>
<td>53</td>
<td>UPARM3=(0,255), user defined char 3</td>
</tr>
<tr>
<td>54</td>
<td>CAPACITY=(MINIMUM,MAXIMUM), line speed</td>
</tr>
<tr>
<td>55</td>
<td>COSTTIME=(0,255), cost per connect time</td>
</tr>
<tr>
<td>56</td>
<td>COSTBYTE=(0,255), cost per byte transmitted</td>
</tr>
<tr>
<td>57</td>
<td>PDELAY=(MINIMUM,MAXIMUM), propagation delay</td>
</tr>
<tr>
<td>58</td>
<td>SECURITY=(UNSECURE,MAXIMUM), security level for TG</td>
</tr>
</tbody>
</table>

* #BATCHSC APPNCOS PRIORITY=LOW,NUMBER=12 transmission priority

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>59</td>
<td>LINEROW WEIGHT=160, line row weight</td>
</tr>
<tr>
<td>60</td>
<td>NUMBER=1, line row number</td>
</tr>
<tr>
<td>61</td>
<td>UPARM1=(0,255), user defined char 1</td>
</tr>
<tr>
<td>62</td>
<td>UPARM2=(0,255), user defined char 2</td>
</tr>
<tr>
<td>63</td>
<td>UPARM3=(0,255), user defined char 3</td>
</tr>
<tr>
<td>64</td>
<td>CAPACITY=(1G,MAXIMUM), line speed</td>
</tr>
<tr>
<td>65</td>
<td>COSTTIME=(0,0), cost per connect time</td>
</tr>
<tr>
<td>66</td>
<td>COSTBYTE=(0,0), cost per byte transmitted</td>
</tr>
<tr>
<td>67</td>
<td>PDELAY=(MINIMUM,MAXIMUM), propagation delay</td>
</tr>
<tr>
<td>68</td>
<td>SECURITY=(PUBLIC,MAXIMUM), security level for TG</td>
</tr>
<tr>
<td>69</td>
<td>NODEROW NUMBER=1, node row number</td>
</tr>
</tbody>
</table>
WEIGHT=5, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,31), route addition resistance

LINEROW WEIGHT=40, line row weight *
NUMBER=2, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1G,MAXIMUM), line speed *
COSTTIME=(0,0), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(PUBLIC,MAXIMUM) security level for TG

NODEROW NUMBER=2, node row number *
WEIGHT=10, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,63), route addition resistance

LINEROW WEIGHT=60, line row weight *
NUMBER=3, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1G,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(PUBLIC,MAXIMUM) security level for TG

NODEROW NUMBER=3, node row number *
WEIGHT=20, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,95), route addition resistance

LINEROW WEIGHT=80, line row weight *
NUMBER=4, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(100M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(PUBLIC,MAXIMUM) security level for TG

NODEROW NUMBER=4, node row number *
WEIGHT=40, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,127), route addition resistance

LINEROW WEIGHT=100, line row weight *
NUMBER=5, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(25M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
PDELAY=(MINIMUM,MAXIMUM), propagation delay *
SECURITY=(PUBLIC,MAXIMUM) security level for TG

NODEROW NUMBER=5, node row number *
WEIGHT=60, node row weight *
CONGEST=(LOW,LOW), congestion *
ROUTERES=(0,159), route addition resistance

LINEROW WEIGHT=120, line row weight *
NUMBER=6, line row number *
UPARM1=(0,255), user defined char 1 *
UPARM2=(0,255), user defined char 2 *
UPARM3=(0,255), user defined char 3 *
CAPACITY=(1M,MAXIMUM), line speed *
COSTTIME=(0,64), cost per connect time *
COSTBYTE=(0,0), cost per byte transmitted *
<table>
<thead>
<tr>
<th>NODE ROW NUMBER</th>
<th>NODE WEIGHT</th>
<th>CONGESTION</th>
<th>ROUTERES</th>
<th>LINE ROW WEIGHT</th>
<th>LINE NUMBER</th>
<th>UPARM1</th>
<th>UPARM2</th>
<th>UPARM3</th>
<th>CAPACITY</th>
<th>COSTTIME</th>
<th>COSTBYTE</th>
<th>PDELAY</th>
<th>SECURITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>80</td>
<td>(LOW, LOW)</td>
<td>(0, 191)</td>
<td>140</td>
<td>7</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>(10G, MAXIMUM)</td>
<td>(0, 64)</td>
<td>(0, 128)</td>
<td>MINIMUM, MAXIMUM</td>
<td>PUBLIC, MAXIMUM</td>
</tr>
<tr>
<td>7</td>
<td>120</td>
<td>(LOW, HIGH)</td>
<td>(0, 223)</td>
<td>160</td>
<td>8</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>(1G, MAXIMUM)</td>
<td>(0, 64)</td>
<td>(0, 128)</td>
<td>MINIMUM, MAXIMUM</td>
<td>PUBLIC, MAXIMUM</td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>(LOW, HIGH)</td>
<td>(0, 255)</td>
<td>180</td>
<td>9</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>(16M, MAXIMUM)</td>
<td>(0, 128)</td>
<td>(0, 128)</td>
<td>MINIMUM, MAXIMUM</td>
<td>PUBLIC, MAXIMUM</td>
</tr>
<tr>
<td>9</td>
<td>200</td>
<td>(LOW, HIGH)</td>
<td>(0, 240)</td>
<td>200</td>
<td>10</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>(1M, MAXIMUM)</td>
<td>(0, 128)</td>
<td>(0, 128)</td>
<td>MINIMUM, MAXIMUM</td>
<td>PUBLIC, MAXIMUM</td>
</tr>
<tr>
<td>10</td>
<td>220</td>
<td></td>
<td></td>
<td></td>
<td>11</td>
<td>0, 255</td>
<td>0, 255</td>
<td>0, 255</td>
<td>(64K, MAXIMUM)</td>
<td>(0, 64)</td>
<td>(0, 0)</td>
<td>MINIMUM, MAXIMUM</td>
<td>PUBLIC, MAXIMUM</td>
</tr>
</tbody>
</table>

Appendix A. IBM-supplied tables 1355
<table>
<thead>
<tr>
<th>Transmission Group</th>
<th>Profile Definition</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security Level</th>
<th>Propagation Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Ethernet</strong></td>
<td>ETHERNET TGP</td>
<td>0</td>
<td>0</td>
<td>UNSECURE</td>
<td>NEGLIGIB</td>
<td>10M</td>
</tr>
<tr>
<td><strong>Token ring</strong></td>
<td>TOKNRING TGP</td>
<td>0</td>
<td>0</td>
<td>UNSECURE</td>
<td>NEGLIGIB</td>
<td>4M</td>
</tr>
<tr>
<td><strong>ISDN non-switched</strong></td>
<td>ISDNNSWT TGP</td>
<td>0</td>
<td>0</td>
<td>UNSECURE</td>
<td>TERRESTR</td>
<td>64K</td>
</tr>
<tr>
<td><strong>ISDN switched</strong></td>
<td>ISDNSWTD TGP</td>
<td>128</td>
<td>128</td>
<td>UNSECURE</td>
<td>TERRESTR</td>
<td>10K</td>
</tr>
<tr>
<td><strong>SDLC non-switched</strong></td>
<td>SDLCNNSWT TGP</td>
<td>0</td>
<td>0</td>
<td>UNSECURE</td>
<td>TERRESTR</td>
<td>10K</td>
</tr>
<tr>
<td><strong>SDLC switched</strong></td>
<td>SDLCSWTD TGP</td>
<td>128</td>
<td>128</td>
<td>UNSECURE</td>
<td>TERRESTR</td>
<td>10K</td>
</tr>
<tr>
<td><strong>X.25</strong></td>
<td>X25 TGP</td>
<td>128</td>
<td>128</td>
<td>PUBLIC</td>
<td>PACKET</td>
<td>10K</td>
</tr>
<tr>
<td><strong>16M token ring</strong></td>
<td>TRING16M TGP</td>
<td>0</td>
<td>0</td>
<td>UNSECURE</td>
<td>NEGLIGIB</td>
<td>16M</td>
</tr>
<tr>
<td><strong>Channel</strong></td>
<td>CHANNEL TGP</td>
<td>0</td>
<td>0</td>
<td>SECURE</td>
<td>NEGLIGIB</td>
<td>36M</td>
</tr>
<tr>
<td><strong>ESCON channel</strong></td>
<td>ESCON TGP</td>
<td>0</td>
<td>0</td>
<td>SECURE</td>
<td>NEGLIGIB</td>
<td>200M</td>
</tr>
</tbody>
</table>
### XCF

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>XCF</td>
<td>TGP</td>
<td>COST=0,0</td>
<td>SECURITY=SECURE,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=SECURE,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=NEGLIGIB</td>
<td>CAPACITY=25M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ATM TGs over PVCs [campus (private) ATM best effort]

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATMPVCCB</td>
<td>TGP</td>
<td>COST=0,0</td>
<td>SECURITY=UNSECURE,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=UNSECURE,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=TERRESTR</td>
<td>CAPACITY=25M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ATM TGs over PVCs (campus ATM reserved bandwidth)

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATMPVCCR</td>
<td>TGP</td>
<td>COST=0,0</td>
<td>SECURITY=UNSECURE,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=UNSECURE,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=NEGLIGIB</td>
<td>CAPACITY=25M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ATM TGs over PVCs [WAN (public) ATM best effort]

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATMPVCW</td>
<td>TGP</td>
<td>COST=0,128</td>
<td>SECURITY=PUBLIC,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=PUBLIC,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=PACKET</td>
<td>CAPACITY=25M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ATM TGs over PVCs (WAN ATM reserved bandwidth)

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATMPVCWR</td>
<td>TGP</td>
<td>COST=0,0</td>
<td>SECURITY=PUBLIC,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=PUBLIC,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=TERRESTR</td>
<td>CAPACITY=25M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ATM TGs over SVCs (campus ATM best effort)

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATM SVCBB</td>
<td>TGP</td>
<td>COST=0,0</td>
<td>SECURITY=UNSECURE,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=UNSECURE,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=TERRESTR</td>
<td>CAPACITY=25M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ATM TGs over SVCs (campus ATM reserved bandwidth)

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATM SVCBR</td>
<td>TGP</td>
<td>COST=64,0</td>
<td>SECURITY=UNSECURE,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=UNSECURE,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=NEGLIGIB</td>
<td>CAPACITY=25M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ATM TGs over SVCs (WAN ATM best effort)

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATM SVCWB</td>
<td>TGP</td>
<td>COST=0,128</td>
<td>SECURITY=PUBLIC,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=PUBLIC,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=PACKET</td>
<td>CAPACITY=25M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ATM TGs over SVCs (WAN ATM reserved bandwidth)

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATM SVCWR</td>
<td>TGP</td>
<td>COST=128,0</td>
<td>SECURITY=PUBLIC,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=PUBLIC,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=TERRESTR</td>
<td>CAPACITY=25M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Enterprise Extender, campus

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>EEEXTCamp</td>
<td>TGP</td>
<td>COST=0,0</td>
<td>SECURITY=UNSECURE,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=UNSECURE,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=TERRESTR</td>
<td>CAPACITY=4M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Enterprise Extender, wide area network

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>EEEXTWAN</td>
<td>TGP</td>
<td>COST=0,0</td>
<td>SECURITY=PUBLIC,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=PUBLIC,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=PACKET</td>
<td>CAPACITY=56K</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Fast Ethernet

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>FASTENET</td>
<td>TGP</td>
<td>COST=0,0</td>
<td>SECURITY=UNSECURE,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=UNSECURE,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=NEGLIGIB</td>
<td>CAPACITY=100M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Gigabit Ethernet

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Protocol</th>
<th>Cost Time</th>
<th>Cost Byte</th>
<th>Security</th>
<th>P Delay</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>GIGENET</td>
<td>TGP</td>
<td>COST=0,0</td>
<td>SECURITY=UNSECURE,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>COSTBYTE=0</td>
<td>*</td>
<td>0,SECURITY=UNSECURE,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDELAY=NEGLIGIB</td>
<td>CAPACITY=1000M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Message flooding table (ISTMSFLD)

* Module Name: ISTMSFLD
* GDE: MESSAGE FLOODING TABLE
* DOC: TABLE OF MESSAGES THAT ARE CANDIDATES FOR SUPPRESSION.
* * Copyright: Licensed Materials - Property of IBM
* * "Restricted Materials of IBM"
* 5650-ZOS
* Copyright IBM Corp. 1992, 2013
*
* Status: CSV2R1
* *
* TABLE FOR ACF/VTAM PRE-DESIGNATED SUPPRESSION CANDIDATES
* NOTE: INTERNAL TABLE IS ISTMSFLI
* $MAC(ISTMSFLD),COMP(MSGS),PROD(VTAM): EXTERNAL MESSAGE FLOOD TABLE
* *
<table>
<thead>
<tr>
<th>Flag Reason</th>
<th>Release Date</th>
<th>Origin</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I0=</td>
<td>FIFOOMST</td>
<td>DPSITEM 920623</td>
<td>450767: External Flood Table Support</td>
</tr>
<tr>
<td>$U2=</td>
<td>P108126</td>
<td>DPSITEM 930325</td>
<td>226987: Flood Table Enhancements</td>
</tr>
<tr>
<td>$U3=</td>
<td>FIF0R004</td>
<td>DPSITEM 920430</td>
<td>226987: Update for defaults</td>
</tr>
<tr>
<td>$T1=</td>
<td>P104816</td>
<td>HATAPPN 921012</td>
<td>591683: Add NQN replacement messages</td>
</tr>
<tr>
<td>$O4=</td>
<td>OY57097</td>
<td>SNAIP 930323</td>
<td>324086: New message IST1386I</td>
</tr>
<tr>
<td>$H1=</td>
<td>VS@@@MST</td>
<td>VSE/ESA 931110</td>
<td>544872: New message IST1000I</td>
</tr>
<tr>
<td>$U4=</td>
<td>P112611</td>
<td>SNA/IP2 940314</td>
<td>139000: Add message IST1300I</td>
</tr>
<tr>
<td>$L1=</td>
<td>KFRDR019</td>
<td>FREEDOM 950329</td>
<td>699466: Implement VTA09960752</td>
</tr>
<tr>
<td>$Y1=</td>
<td>P118721</td>
<td>FREEDOM 950419</td>
<td>699466: Correct M664 link</td>
</tr>
<tr>
<td>$F1=</td>
<td>P127165</td>
<td>CSV2R6</td>
<td>980226 MCGEACHY: Add some HPR/UDP messages</td>
</tr>
<tr>
<td>$A1=</td>
<td>OW30833</td>
<td>CSV2R7</td>
<td>980518 MTLTA: Add message IST1693I</td>
</tr>
<tr>
<td>$A2=</td>
<td>OW25675</td>
<td>CSV2R7</td>
<td>980517 MTLTA: Add message IST520I</td>
</tr>
<tr>
<td>$G1=</td>
<td>P127943</td>
<td>CSV2R7</td>
<td>980918 JPERKINS: Correct the integration of OW30833 and OW25675</td>
</tr>
<tr>
<td>$L2=</td>
<td>R12CNN</td>
<td>CSV1R2</td>
<td>000706 GMN: CNN Routing Failure Msg</td>
</tr>
<tr>
<td>$N1=</td>
<td>R14CSALM</td>
<td>CSV1R4</td>
<td>010716 NMCAREY: CSALIMIT Enhancement</td>
</tr>
<tr>
<td>$A3=</td>
<td>OW55284</td>
<td>CSV1R5</td>
<td>020715 KUNKEL: Add message IST1762I</td>
</tr>
<tr>
<td>$Q1=</td>
<td>R16EVRN</td>
<td>CSV1R5</td>
<td>020715 RKunkel: Multiple VRNs for EE</td>
</tr>
<tr>
<td>$Q2=</td>
<td>D131648</td>
<td>R16BASE</td>
<td>021018 RKunkel: Same VRN name allowed for Local and Global</td>
</tr>
<tr>
<td>$Q3=</td>
<td>D132176</td>
<td>R16EIP6</td>
<td>030205 PatBrown: Add message IST1890I</td>
</tr>
<tr>
<td>$Q4=</td>
<td>D132467</td>
<td>R16EEMGT</td>
<td>030408 VANZIMM: Add messages for SNAMDMT Server subtask</td>
</tr>
<tr>
<td>$R1=</td>
<td>X16RTPRC</td>
<td>CSV1R6</td>
<td>030709 Zebrow: Stalled HPR Pipe Recovery</td>
</tr>
<tr>
<td>ISTMSFLD FLDTAB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERSIST017I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* UNABLE TO LOAD PHASE % +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MD17 FLDENT MESSAGE=0017, MESSAGE IST017I X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME=, DEFAULT TO 30 SECONDS X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VARTX=(YES), PHASE NAME X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HARDCOPY= DEFAULT TO YES</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* IST118I +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* ANOMALY FOUND NEAR RECORD % IN MEMBER % - CODE % +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MD18 FLDENT MESSAGE=0118, MESSAGE IST118I X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME=, DEFAULT TO 30 SECONDS X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VARTX=(NO, COUNT X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES, MEMBER NAME X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES), CODE X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HARDCOPY= DEFAULT TO YES</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* IST154I +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* EXPANSION FAILED FOR % BUFFER POOL - CODE %, %</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* USERID = % +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MD14 FLDENT MESSAGE=0154, MESSAGE IST154I X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME=, DEFAULT TO 30 SECONDS X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VARTX=(YES, BUFFER POOL X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES, CODE X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES), USERID X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HARDCOPY= DEFAULT TO YES</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* IST182I +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* UNABLE TO TERMINATE SESSIONS FOR ID=% - % - % - %</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* INSUFFICIENT STORAGE +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MD18 FLDENT MESSAGE=0182, MESSAGE IST182I X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME=, DEFAULT TO 30 SECONDS X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VARTX=(YES, DATASET NAME X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES, NODE NAME X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HARDCOPY= DEFAULT TO YES</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* IST192I +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* POA MSG TRANSFER FAILED - INSUFFICIENT STORAGE +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MD19 FLDENT MESSAGE=0192, MESSAGE IST192I X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME=, DEFAULT TO 30 SECONDS X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VARTX=(YES, MINOR NODE X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HARDCOPY= DEFAULT TO YES</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* IST208I +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* UNABLE TO TERMINATE SESSIONS FOR ID = % - % - %</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* INSUFFICIENT STORAGE +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MD20 FLDENT MESSAGE=0208, MESSAGE IST208I X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME=, DEFAULT TO 30 SECONDS X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VARTX= YES, MINOR NODE X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HARDCOPY= DEFAULT TO YES</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* IST234I +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/* I/O ERROR % +*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MD23 FLDENT MESSAGE=0234, MESSAGE IST234I X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME=, DEFAULT TO 30 SECONDS X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VARTX= YES, ONE TEXT FIELD X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HARDCOPY= DEFAULT TO YES</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
/**/ IST301I */ INSUFFICIENT VFBUF STORAGE FOR I/O TRACE TABLE */ M301 FLDENT MESSAGE=0301, MESSAGE IST301I X TIME=, DEFAULT TO 30 SECONDS X HARDCOPY= DEFAULT TO YES */ /**/ IST348I */ UNABLE TO PROCESS DISCONNECTION FOR PU = % */ DUE TO LACK OF STORAGE */ M348 FLDENT MESSAGE=0348, MESSAGE IST348I X TIME=, DEFAULT TO 30 SECONDS X VARTXT= (YES), PUNAME X HARDCOPY= DEFAULT TO YES */ /**/ IST366I */ CONFIG % UNABLE TO DEFINE % - MAXIMUM NUMBER OF */ NETWORK ADDRESSES FOR HOST SUBAREA EXCEEDED */ M366 FLDENT MESSAGE=0366, MESSAGE IST366I X TIME=, DEFAULT TO 30 SECONDS X VARTXT= (YES, CONFIG NAME X NO), NODE NAME X HARDCOPY= DEFAULT TO YES */ /**/ IST367I */ NO STORAGE TO DEFINE NODE % CONFIG % */ M367 FLDENT MESSAGE=0367, MESSAGE IST367I X TIME=, DEFAULT TO 30 SECONDS X VARTXT= (NO, NODE NAME X YES), CONFIG NAME X HARDCOPY= DEFAULT TO YES */ /**/ IST380I */ ERROR FOR ID = % REQUEST: % SENSE % */ M380 FLDENT MESSAGE=0380, MESSAGE IST380I X TIME=, DEFAULT TO 30 SECONDS X VARTXT= (NO, NODE NAME X YES), RU NAME X YES), SENSE CODE X HARDCOPY= DEFAULT TO YES */ /**/ IST436I */ STORAGE NOT AVAILABLE FOR TUNING STATISTICS DATA */ M436 FLDENT MESSAGE=0436, MESSAGE IST436I X TIME=, DEFAULT TO 30 SECONDS X HARDCOPY= DEFAULT TO YES */ /**/ IST520I */ UNABLE TO PROCESS % */ M520 FLDENT MESSAGE=0520, MESSAGE IST520I 0A2AX TIME=, DEFAULT 30 SECONDS 0A2AX VARTXT= (YES), REQUEST 0A2AX LIST=1694, LINK TO IST1694I 0A2AX HARDCOPY= DEFAULT TO YES */ /**/ IST522I */ ROUTE FAILED FROM % TO % DSA % NETID % */ M522 FLDENT MESSAGE=0522, MESSAGE IST522I X TIME=, DEFAULT TO 30 SECONDS X VARTXT= (YES, ER/VR NUMBER X OPTION X YES, FAILED/REJECTED X YES, SA1 X YES, SA2 X YES, FOR TPi X HARDCOPY= DEFAULT TO YES */ /**/ IST526I */ ROUTE FAILED FROM % TO % DSA % NETID % */ M526 FLDENT MESSAGE=0526, MESSAGE IST526I X
Appendix A. IBM-supplied tables 1361
Appendix A. IBM-supplied tables
HARDCOPY= DEFAULT TO YES @Q1A

/* IST1903I */
/* FAILURE OVER VRN vrnname TO CP partner */
M1903 FLDENT MESSAGE=1903, Message IST1903I 0A5AX
TIME=, Default to 30 sec 0A5AX
VARTXT=(YES,NO), VRNName, Partner 0A5AX
HARDCOPY= Default to yes 0A5A

/* IST1915I */
/* % ACTIVATION FAILED - VNAME ALREADY ACTIVE AS GLOBAL */
M1915 FLDENT MESSAGE=1915, Message IST1915I 0Q2AX
TIME=, DEFAULT TO 30 SEC 0Q2AX
VARTXT=(NO), LINE NAME 0Q2AX
HARDCOPY= DEFAULT TO YES 0Q2A

/* IST1916I */
/* % ACTIVATION FAILED - VNAME ALREADY ACTIVE AS LOCAL */
M1916 FLDENT MESSAGE=1916, Message IST1916I 0Q2AX
TIME=, DEFAULT TO 30 SEC 0Q2AX
VARTXT=(NO), LINE NAME 0Q2AX
HARDCOPY= DEFAULT TO YES 0Q2A

/* IST1928I */
/* SNAMGMT CONNECTION TO userid IS ACTIVE */
M1928 FLDENT MESSAGE=1928, Message IST1928I 0Q4AX
TIME=, DEFAULT TO 30 SEC 0Q4AX
VARTXT=(YES), USERID 0Q4AX
HARDCOPY= DEFAULT TO YES 0Q4A

/* IST1929I */
/* SNAMGMT CONNECTION TO userid HAS ENDED */
M1929 FLDENT MESSAGE=1929, Message IST1929I 0Q4AX
TIME=, DEFAULT TO 30 SEC 0Q4AX
VARTXT=(YES), USERID 0Q4AX
HARDCOPY= DEFAULT TO YES 0Q4A

/* IST1931I */
/* SNAMGMT CONNECTION REFUSED FOR userid */
M1931 FLDENT MESSAGE=1931, Message IST1931I 0Q4AX
TIME=, DEFAULT TO 30 SEC 0Q4AX
VARTXT=(YES), USERID 0Q4AX
HARDCOPY= DEFAULT TO YES 0Q4A

/* IST1932I */
/* SNAMGMT SECURITY CHECK FAILED FOR userid */
M1932 FLDENT MESSAGE=1932, Message IST1932I 0Q4AX
TIME=, DEFAULT TO 30 SEC 0Q4AX
VARTXT=(YES), USERID 0Q4AX
HARDCOPY= DEFAULT TO YES 0Q4A

/* IST1933I */
/* SNAMGMT SERVER CLOSING CONNECTION TO userid */
M1933 FLDENT MESSAGE=1933, Message IST1933I 0Q4AX
TIME=, DEFAULT TO 30 SEC 0Q4AX
VARTXT=(YES), USERID 0Q4AX
HARDCOPY= DEFAULT TO YES 0Q4A

/* IST1955I */
/* STALL DETECTED FOR RTP puname TO cpname */
M1955 FLDENT MESSAGE=1955, Message IST1955I 0R1AX
TIME=, Default to 30 sec 0R1AX
VARTXT=(YES,YES), PName, CName 0R1AX
HARDCOPY= Default to yes 0R1A

/* IST1957I */
/* STALL ALLEVIATED FOR RTP puname TO cpname */
M1957 FLDENT MESSAGE=1957, Message IST1957I 0R1AX
TIME=, Default to 30 sec 0R1AX
CMIP services directory definition file (ACYDDF)

An IBM-supplied CMIP services directory definition file with specific code might have security definitions that can contradict your security definitions. Hence, IBM supplies only the following information in the file:

```bash
# This is the VTAM OSI CMIP Directory Definition File
```

A more detailed example of a directory definition file is in the `z/OS Communications Server: SNA Resource Definition Samples`. For more information on the file and CMIP services, refer to the `z/OS Communications Server: SNA Network Implementation Guide` and see "CMIP services directory definition file" on page 1008.
Appendix B. Examples of USS command conversion

This appendix contains examples of how to code USS tables to convert commands. Each example includes the following items:
- The USS table used
- The format for entering the command
- The converted command

**Example 1:** Parameters omitted

The following VTAM operator command is issued:

```
F NET,TNSTAT
```

All parameters are omitted and, therefore, defaults are used during conversion. The following operation-level USS table is used:

```
USSTAB
USSCMD CMD=TNSTAT,FORMAT=BAL
USSPARM PARM=P1,REP=CNSL,DEFAULT=NO
USSPARM PARM=P2,REP=TIME,DEFAULT=60
USSEND
```

The converted command is

```
TNSTAT CNSL(NO) TIME(60)
```

**Example 2:** Positional and keyword parameters (FORMAT=PL1)

The terminal operator command

```
LOFF(PROG) T(COND)
```

is issued. Positional and keyword parameters are used with FORMAT=PL1. The following session-level USS table is used:

```
USSTAB
USSCMD CMD=LOFF,REP=LOGOFF,FORMAT=PL1
USSPARM PARM=P1,REP=APPLID
USSPARM PARM=T,REP=TYPE,DEFAULT=COND
USSEND
```

The converted command is

```
LOGOFF APPLID(PROG) TYPE(COND)
```

**Example 3:** Positional and keyword parameters (FORMAT=PL1)

The terminal operator command

```
LOFF (PGM) T
```

is issued. A null value is taken instead of a default value. Positional and keyword parameters are used with FORMAT=PL1. The following session-level USS table is used:

```
USSTAB
USSCMD CMD=LOFF,REP=LOGOFF,FORMAT=PL1
USSPARM PARM=P1,REP=APPLID
USSPARM PARM=T,REP=TYPE,DEFAULT=COND
USSEND
```
The converted command is
LOGOFF APPLID(PGM) TYPE( )

Because T was coded, the default value specified in the table is not used. If T had not been coded, TYPE(COND) would have resulted. TYPE( ) causes unconditional termination. Note that if you replace DEFAULT=COND with VALUE=COND in the table entry (USSPARM PARM=T,REP=TYPE,VALUE=COND), this command results in a TYPE(COND) instead of TYPE( ).

**Example 4:** Positional and keyword parameters (FORMAT=BAL)

The terminal operator command
LON PROGRAM,IN='7,3,JOhn'

is issued. Positional and keyword parameters are used with FORMAT=BAL.

The following session-level USS table is used:

```plaintext
USSTAB
USSCMD CMD=LON,REP=LOGON,FORMAT=BAL
USSPARM PARM=P1,REP=APPLID,DEFAULT=TESTPROG
USSPARM PARM=MODE,REP=LOGMODE,DEFAULT='PROMPT'
USSPARM PARM=IN,REP=DATA
USSEND
```

The converted command is
LOGON APPLID(PROGRAM) LOGMODE(PROMPT) DATA('7,3,JOhn')

Note that no character translation was performed on JOhn because there were single quotation marks around it. Note also that the single quotation marks around PROMPT in the default declaration have been deleted (by the assembler when the USS table was assembled).

**Example 5:** Using parameter not defined in USS table

The VTAM operator command
F NET,IMR,ID=PU12,OPT=ACT,RECLIM=200

is issued. A parameter (ID) is used that is not defined on the USS table.

The following parameter (ID) is used that is not defined on the USS table.

The following operation-level USS table is used:

```plaintext
USSTAB
USSCMD CMD=IMR,FORMAT=BAL
USSPARM PARM=OPT,DEFAULT=ACT
USSPARM PARM=RECLIM,DEFAULT=10
USSEND
```

The converted command is
IMR ID(PU12) OPT(ACT) RECLIM(200)

Because ID is not defined by a USSPARM macroinstruction in the USS table, the keyword is not changed during conversion.

**Example 6:** Renaming keyword replaced by verb

The VTAM operator command
F NET,XYZ,ID=PU12
is issued. This example shows how to rename a keyword that is replaced by a verb.

The following VTAM operator USS table is used:

```
USSTAB
*  
USSCMD CMD=MODIFY,FORMAT=BAL
USSPARM PARM=XYZ,REP=VERB,VALUE=XYZ
*  
USSCMD CMD=XYZ,REP=IMR,FORMAT=BAL
USSPARM PARM=OPT,DEFAULT=ACT
USSPARM PARM=RECLIM,DEFAULT=10
USSEND
```

This results, after the first pass, in the following command of the form

```
VERB(XYZ) ID(PU12)
```

which in turn, after the second pass, results in the following converted command

```
IMR ID(PU12) OPT(ACT) RECLIM(10)
```

Note that the FORMAT=BAL specification on CMD=XYZ is ignored. After the first pass, VTAM uses the PL/I format for command processing.

**Example 7: Using interpret and USS tables**

If you want to use the facilities of both the USS table and the interpret table, the SEQNCE parameters in the interpret table should be coded to avoid a match during the first search of the interpret table (which would prevent the USS table from being used). Instead, a USSPARM statement in the USS table should be used to specify an APPLID that will match an interpret table entry during the second pass.

- The interpret table is coded as follows:

  ```
  XRFINTAB INTAB
  LOGCHAR APPLID=(USERVAR,ZZIMSUV),SEQNCE='ZZIMS'
  ENDINTAB XRFINTAB
  ```

- The USS table is coded as follows:

  ```
  XRFUSTAB USSTAB
  USSCMD CMD=IMS,REP=LOGON,FORMAT=PL1
  USSPARM PARM=P1,REP=APPLID,DEFAULT=ZZIMS
  USSPARM PARM=P2,REP=LOGMODE,DEFAULT=ILOGMODE
  USSEND
  END XRFUSTAB
  ```

- If the LOGON sequence entered by the terminal user is “IMS,”
  - The interpret table will be searched for a matching entry, but none will be found.
  - The message will be translated, using the information in the USS table, to read
    
    ```LOGON APPLID(ZZIMS) LOGMODE(ILOGMODE)```
  - The interpret table will be searched again, this time using “ZZIMS,” the string specified as the APPLID in the reformatted LOGON message. Because “ZZIMS” does match an entry in the interpret table, the information in that entry will be used. If the current value of the USERVAR ZZIMSUV is “IMS1,” the session will be established to the IMS1 application.

- On the other hand, if the LOGON sequence entered by the terminal user is “ZZIMS,” a match will be found during the initial search of the interpret table;
Therefore, the USS table will not be used (the LOGMODE operand specified in the USS table will not be added). The USERVAR specified in the interpret table entry will still be used.

Example 8: Using TRANSLATE=NO to pass a mixed case password, assuming the translation table is used to convert all text to upper case.

The terminal operator command

```
app11 user1/PassWrD1 interact
```

is issued. Positional parameters are used with FORMAT=PL1. The following session-level USS table is used:

```
AUSSTAB USSTAB
  APPL1 USSCMD CMD=APPL1,REP=LOGON,FORMAT=PL1
  USSPARAM PARM=APPLID,REP=APPLID,DEFAULT=APPL1
  USSPARAM PARM=P1,REP=DATA,TRANSLATE=NO
  USSPARAM PARM=P2,REP=LOGMODE
  USSEND
```

The converted command is

```
LOGON APPLID(APPL1) DATA(user1/PassWrD1) LOGMODE(INTERACT)
```

Note: No character translation was performed on user1/PassWrD1 because TRANSLATE=NO was coded on the DATA USSPARAM.

Example 9: Using the default TRANSLATE=YES to pass a mixed case password, assuming the translation table is used to convert all text to upper case.

The terminal operator command

```
app11 'user1/PassWrD1'
```

is issued. Positional parameters are used with FORMAT=PL1. The following session-level USS table is used:

```
AUSSTAB USSTAB
  APPL1 USSCMD CMD=APPL1,REP=LOGON,FORMAT=PL1
  USSPARAM PARM=APPLID,REP=APPLID,DEFAULT=APPL1
  USSPARAM PARM=P1,REP=DATA
  USSPARAM PARM=P2,REP=LOGMODE
  USSEND
```

The converted command is

```
LOGON APPLID(APPL1) DATA('user1/PassWrD1') LOGMODE(INTERACT)
```

Note: No character translation was performed on user1/PassWrD1 because there were single quotation marks around it. The single quotation marks are removed before the data is passed to the application.

For further information on converting USS commands, see the z/OS Communications Server: SNA Network Implementation Guide.
Appendix C. VTAMTOPO filtering option reporting

Table 99 summarizes the results of using the VTAMTOPO filtering option for reporting a switched PU under an NCP.

The following information shows the legend for the table:

| notIGNR/INCL | Neither IGNORE or INCLUDE specified. |
| r VTAMTOPO= not specified, REPORT inherited from node above. |
| R VTAMTOPO=REPORT specified (or NOLLINES at containing GROUP). |
| nr VTAMTOPO= not specified, NOREPORT inherited from node above. |
| NR VTAMTOPO=NOREPORT specified (or NOSWPUS at containing GROUP). |
| NotRep | Switched PU is not reported. |
| IGNR | VTAMTOPO=IGNORE specified at designated major node. |
| INCL | VTAMTOPO=INCLUDE specified at the designated major node. |
| any | Value of VTAMTOPO= does not matter, inclusion not specified. |
| R-NCP | Switched PU is reported under the NCP under which the Switched PU is connected. |
| R-SSCP | Switched PU is reported under the SSCP directly (not under any NCP), as it is when it is not connected. |

Note:
1. Values shown are assumed to be set in the applicable major node, individual PUX or SW PU before the connection is established. A MODIFY VTAMTOPO to set these values after the connection is established may not show the expected result before the connection is taken down and reestablished.
2. This table can also be used for the Switched PUs that are connected under an XCA. In the results column, all R-NCP will be R-SSCP, because the XCA is not represented by a PU type, and does not appear as a discrete node in SNA local topology.
3. The PUX is a place holder for a future connected switched PU under the switched line. It takes its VTAMTOPO value either explicitly from the line GROUP value, or implicitly from the NCP or the XCA major node value. If its VTAMTOPO value is explicitly set, it cannot be modified.

Table 99. Connected switched PU report

<table>
<thead>
<tr>
<th>VTAMTOPO value on NCP</th>
<th>PUX</th>
<th>SWND</th>
<th>SWPU</th>
<th>SPWU result</th>
</tr>
</thead>
<tbody>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>R</td>
<td>r</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>INCL</td>
<td>r</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>R</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>INCL</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>VTAMTOPO value on</td>
<td>NCP</td>
<td>PUX</td>
<td>SWND</td>
<td>SWPU</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>R</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>INCL</td>
<td>NR</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>NR</td>
<td>nr</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>IGNR</td>
<td>nr</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>NR</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>IGNR</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>NR</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>r/R</td>
<td>IGNR</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>R</td>
<td>r</td>
<td>R-SSCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>INCL</td>
<td>r</td>
<td>R-SSCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>R</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>INCL</td>
<td>R</td>
<td>R-SSCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>R</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>IGNR</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>NR</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>IGNR</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>NR</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>notIGNR/INCL</td>
<td>nr/NR</td>
<td>IGNR</td>
<td>R</td>
<td>NotRep</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>R</td>
<td>r</td>
<td>R-NCP</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>INCL</td>
<td>r</td>
<td>R-NCP</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>R</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>INCL</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>R</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>INCL</td>
<td>NR</td>
<td>R-NCP</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>NR</td>
<td>nr</td>
<td>R-NCP</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>IGNR</td>
<td>nr</td>
<td>R-NCP</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>NR</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>IGNR</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>NR</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>INCL</td>
<td>any</td>
<td>IGNR</td>
<td>R</td>
<td>R-NCP</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>R</td>
<td>r</td>
<td>R-SSCP</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>INCL</td>
<td>r</td>
<td>R-SSCP</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>R</td>
<td>R</td>
<td>R-SSCP</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>R</td>
<td>NR</td>
<td>NotRep</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>INCL</td>
<td>NR</td>
<td>R-SSCP</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>NR</td>
<td>nr</td>
<td>NotRep</td>
</tr>
</tbody>
</table>
Table 99. Connected switched PU report (continued)

<table>
<thead>
<tr>
<th>VTAMTOPO value on</th>
<th>NCP</th>
<th>PUX</th>
<th>SWND</th>
<th>SWPU</th>
<th>SPWU result</th>
</tr>
</thead>
<tbody>
<tr>
<td>IGNR</td>
<td>any</td>
<td>IGNR</td>
<td>nr</td>
<td></td>
<td>NotRep</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>NotRep</td>
<td>NotRep</td>
<td></td>
<td>NotRep</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>IGNR</td>
<td>NR</td>
<td></td>
<td>NotRep</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>NR</td>
<td>R</td>
<td></td>
<td>R-SSCP</td>
</tr>
<tr>
<td>IGNR</td>
<td>any</td>
<td>IGNR</td>
<td>R</td>
<td></td>
<td>NotRep</td>
</tr>
</tbody>
</table>
Appendix D. Related protocol specifications

This appendix lists the related protocol specifications (RFCs) for TCP/IP. The Internet Protocol suite is still evolving through requests for comments (RFC). New protocols are being designed and implemented by researchers and are brought to the attention of the Internet community in the form of RFCs. Some of these protocols are so useful that they become recommended protocols. That is, all future implementations for TCP/IP are recommended to implement these particular functions or protocols. These become the *de facto* standards, on which the TCP/IP protocol suite is built.

You can request RFCs through electronic mail, from the automated Network Information Center (NIC) mail server, by sending a message to service@nic.ddn.mil with a subject line of RFC nnnn for text versions or a subject line of RFC nnnn.PS for PostScript versions. To request a copy of the RFC index, send a message with a subject line of RFC INDEX.

For more information, contact nic@nic.ddn.mil or at:

Government Systems, Inc.
Attn: Network Information Center
14200 Park Meadow Drive
Suite 200
Chantilly, VA  22021

Hard copies of all RFCs are available from the NIC, either individually or by subscription. Online copies are available at the following Web address:

http://www.rfc-editor.org/rfc.html

Draft RFCs that have been implemented in this and previous Communications Server releases are listed at the end of this topic.

Many features of TCP/IP Services are based on the following RFCs:

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title and Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 652</td>
<td>Telnet output carriage-return disposition option D. Crocker</td>
</tr>
<tr>
<td>RFC 653</td>
<td>Telnet output horizontal tabstops option D. Crocker</td>
</tr>
<tr>
<td>RFC 654</td>
<td>Telnet output horizontal tab disposition option D. Crocker</td>
</tr>
<tr>
<td>RFC 655</td>
<td>Telnet output formfeed disposition option D. Crocker</td>
</tr>
<tr>
<td>RFC 657</td>
<td>Telnet output vertical tab disposition option D. Crocker</td>
</tr>
<tr>
<td>RFC 658</td>
<td>Telnet output linefeed disposition D. Crocker</td>
</tr>
<tr>
<td>RFC 698</td>
<td>Telnet extended ASCII option T. Mock</td>
</tr>
</tbody>
</table>
RFC 726  
Remote Controlled Transmission and Echoing Telnet option J. Postel, D. Crocker

RFC 727  
Telnet logout option M.R. Crispin

RFC 732  
Telnet Data Entry Terminal option J.D. Day

RFC 733  
Standard for the format of ARPA network text messages D. Crocker, J. Vittal, K.T. Pogran, D.A. Henderson

RFC 734  
SUPDUP Protocol M.R. Crispin

RFC 735  
Revised Telnet byte macro option D. Crocker, R.H. Gumpertz

RFC 736  
Telnet SUPDUP option M.R. Crispin

RFC 749  
Telnet SUPDUP—Output option B. Greenberg

RFC 765  
File Transfer Protocol specification J. Postel

RFC 768  
User Datagram Protocol J. Postel

RFC 779  
Telnet send-location option E. Killian

RFC 783  
TFTP Protocol (revision 2) K.R. Sollins

RFC 791  
Internet Protocol J. Postel

RFC 792  
Internet Control Message Protocol J. Postel

RFC 793  
Transmission Control Protocol J. Postel

RFC 820  
Assigned numbers J. Postel

RFC 821  
Simple Mail Transfer Protocol J. Postel

RFC 822  
Standard for the format of ARPA Internet text messages D. Crocker

RFC 823  
DARPA Internet gateway R. Hinden, A. Sheltzer

RFC 826  
Ethernet Address Resolution Protocol: Or converting network protocol addresses to 48-bit Ethernet address for transmission on Ethernet hardware D. Plummer

RFC 854  
Telnet Protocol Specification J. Postel, J. Reynolds
RFC 855
Telnet Option Specification J. Postel, J. Reynolds

RFC 856
Telnet Binary Transmission J. Postel, J. Reynolds

RFC 857
Telnet Echo Option J. Postel, J. Reynolds

RFC 858
Telnet Suppress Go Ahead Option J. Postel, J. Reynolds

RFC 859
Telnet Status Option J. Postel, J. Reynolds

RFC 860
Telnet Timing Mark Option J. Postel, J. Reynolds

RFC 861
Telnet Extended Options: List Option J. Postel, J. Reynolds

RFC 862
Echo Protocol J. Postel

RFC 863
Discard Protocol J. Postel

RFC 864
Character Generator Protocol J. Postel

RFC 865
Quote of the Day Protocol J. Postel

RFC 868
Time Protocol J. Postel, K. Harrenstien

RFC 877
Standard for the transmission of IP datagrams over public data networks J.T. Korb

RFC 883
Domain names: Implementation specification P.V. Mockapetris

RFC 884
Telnet terminal type option M. Solomon, E. Wimmers

RFC 885
Telnet end of record option J. Postel

RFC 894
Standard for the transmission of IP datagrams over Ethernet networks C. Hornig

RFC 896
Congestion control in IP/TCP internetworks J. Nagle

RFC 903
Reverse Address Resolution Protocol R. Finlayson, T. Mann, J. Mogul, M. Theimer

RFC 904
Exterior Gateway Protocol formal specification D. Mills

RFC 919
Broadcasting Internet Datagrams J. Mogul
RFC 922
Broadcasting Internet datagrams in the presence of subnets J. Mogul

RFC 927
TACACS user identification Telnet option B.A. Anderson

RFC 933
Output marking Telnet option S. Silverman

RFC 946
Telnet terminal location number option R. Nedved

RFC 950
Internet Standard Subnetting Procedure J. Mogul, J. Postel

RFC 952
DoD Internet host table specification K. Harrenstien, M. Stahl, E. Feinler

RFC 959
File Transfer Protocol J. Postel, J.K. Reynolds

RFC 961
Official ARPA-Internet protocols J.K. Reynolds, J. Postel

RFC 974
Mail routing and the domain system C. Partridge

RFC 1001

RFC 1002

RFC 1006
ISO transport services on top of the TCP: Version 3 M.T. Rose, D.E. Cass

RFC 1009
Requirements for Internet gateways R. Braden, J. Postel

RFC 1011
Official Internet protocols J. Reynolds, J. Postel

RFC 1013
X Window System Protocol, version 11: Alpha update April 1987 R. Scheifler

RFC 1014
XDR: External Data Representation standard Sun Microsystems

RFC 1027
Using ARP to implement transparent subnet gateways S. Carl-Mitchell, J. Quarterman

RFC 1032
Domain administrators guide M. Stahl

RFC 1033
Domain administrators operations guide M. Lottor

RFC 1034
Domain names—concepts and facilities P.V. Mockapetris
RFC 1035
Domain names—implementation and specification P.V. Mockapetris

RFC 1038
Draft revised IP security option M. St. Johns

RFC 1041
Telnet 3270 regime option Y. Rekhter

RFC 1042
Standard for the transmission of IP datagrams over IEEE 802 networks J. Postel, J. Reynolds

RFC 1043
Telnet Data Entry Terminal option: DODIIS implementation A. Yasuda, T. Thompson

RFC 1044
Internet Protocol on Network System's HYPERchannel: Protocol specification K. Hardwick, J. Lekashman

RFC 1053
Telnet X.3 PAD option S. Levy, T. Jacobson

RFC 1055
Nonstandard for transmission of IP datagrams over serial lines: SLIP J. Romkey

RFC 1057

RFC 1058
Routing Information Protocol C. Hedrick

RFC 1060
Assigned numbers J. Reynolds, J. Postel

RFC 1067

RFC 1071
Computing the Internet checksum R.T. Braden, D.A. Borman, C. Partridge

RFC 1072
TCP extensions for long-delay paths V. Jacobson, R.T. Braden

RFC 1073
Telnet window size option D. Waitzman

RFC 1079
Telnet terminal speed option C. Hedrick

RFC 1085
ISO presentation services on top of TCP/IP based internets M.T. Rose

RFC 1091
Telnet terminal-type option J. VanBokkelen

RFC 1094
NFS: Network File System Protocol specification Sun Microsystems

RFC 1096
Telnet X display location option G. Marcy

RFC 1101
DNS encoding of network names and other types P. Mockapetris
RFC 1112
Host extensions for IP multicasting S.E. Deering

RFC 1113
Privacy enhancement for Internet electronic mail: Part I — message encipherment and authentication procedures J. Linn

RFC 1118
Hitchhikers Guide to the Internet E. Krol

RFC 1122
Requirements for Internet Hosts—Communication Layers R. Braden, Ed.

RFC 1123
Requirements for Internet Hosts—Application and Support R. Braden, Ed.

RFC 1146
TCP alternate checksum options J. Zweig, C. Partridge

RFC 1155
Structure and identification of management information for TCP/IP-based internets M. Rose, K. McCloghrie

RFC 1156
Management Information Base for network management of TCP/IP-based internets K. McCloghrie, M. Rose

RFC 1157

RFC 1158
Management Information Base for network management of TCP/IP-based internets: MIB-II M. Rose

RFC 1166
Internet numbers S. Kirkpatrick, M.K. Stahl, M. Recker

RFC 1179
Line printer daemon protocol L. McLaughlin

RFC 1180
TCP/IP tutorial T. Socolofsky, C. Kale

RFC 1183
New DNS RR Definitions C.F. Everhart, L.A. Mamakos, R. Ullmann, P.V. Mockapetris

RFC 1184
Telnet Linemode Option D. Borman

RFC 1186
MD4 Message Digest Algorithm R.L. Rivest

RFC 1187
Bulk Table Retrieval with the SNMP M. Rose, K. McCloghrie, J. Davin

RFC 1188
Proposed Standard for the Transmission of IP Datagrams over FDDI Networks D. Katz

RFC 1190
Experimental Internet Stream Protocol: Version 2 (ST-II) C. Topolcic
RFC 1191  
Path MTU discovery J. Mogul, S. Deering

RFC 1198  
FYI on the X window system R. Scheifler

RFC 1207  
FYI on Questions and Answers: Answers to commonly asked “experienced Internet user” questions G. Malkin, A. Marine, J. Reynolds

RFC 1208  
Glossary of networking terms O. Jacobsen, D. Lynch

RFC 1213  
Management Information Base for Network Management of TCP/IP-based internets: MIB-II K. McCloghrie, M.T. Rose

RFC 1215  
Convention for defining traps for use with the SNMP M. Rose

RFC 1227  
SNMP MUX protocol and MIB M.T. Rose

RFC 1228  
SNMP-DPI: Simple Network Management Protocol Distributed Program Interface G. Carpenter, B. Wijnen

RFC 1229  
Extensions to the generic-interface MIB K. McCloghrie

RFC 1230  
IEEE 802.4 Token Bus MIB K. McCloghrie, R. Fox

RFC 1231  
IEEE 802.5 Token Ring MIB K. McCloghrie, R. Fox, E. Decker

RFC 1236  
IP to X.121 address mapping for DDN L. Morales, P. Hasse

RFC 1256  
ICMP Router Discovery Messages S. Deering, Ed.

RFC 1267  
Border Gateway Protocol 3 (BGP-3) K. Lougheed, Y. Rekhter

RFC 1268  
Application of the Border Gateway Protocol in the Internet Y. Rekhter, P. Gross

RFC 1269  
Definitions of Managed Objects for the Border Gateway Protocol: Version 3 S. Willis, J. Burruss

RFC 1270  
SNMP Communications Services F. Kastenholz, ed.

RFC 1285  
FDDI Management Information Base J. Case

RFC 1315  
Management Information Base for Frame Relay DTEs C. Brown, F. Baker, C. Carvalho

RFC 1321  
The MD5 Message-Digest Algorithm R. Rivest
| RFC 1323 | TCP Extensions for High Performance V. Jacobson, R. Braden, D. Borman |
| RFC 1325 | FYI on Questions and Answers: Answers to Commonly Asked "New Internet User" Questions G. Malkin, A. Marine |
| RFC 1327 | Mapping between X.400 (1988)/ISO 10021 and RFC 822 S. Hardcastle-Kille |
| RFC 1340 | Assigned Numbers J. Reynolds, J. Postel |
| RFC 1344 | Implications of MIME for Internet Mail Gateways N. Bornstein |
| RFC 1349 | Type of Service in the Internet Protocol Suite P. Almquist |
| RFC 1350 | The TFTP Protocol (Revision 2) K.R. Sollins |
| RFC 1351 | SNMP Administrative Model J. Davin, J. Galvin, K. McCloghrie |
| RFC 1352 | SNMP Security Protocols J. Galvin, K. McCloghrie, J. Davin |
| RFC 1353 | Definitions of Managed Objects for Administration of SNMP Parties K. McCloghrie, J. Davin, J. Galvin |
| RFC 1354 | IP Forwarding Table MIB F. Baker |
| RFC 1356 | Multiprotocol Interconnect on X.25 and ISDN in the Packet Mode A. Malis, D. Robinson, R. Ullmann |
| RFC 1358 | Charter of the Internet Architecture Board (IAB) L. Chapin |
| RFC 1363 | A Proposed Flow Specification C. Partridge |
| RFC 1368 | Definition of Managed Objects for IEEE 802.3 Repeater Devices D. McMaster, K. McCloghrie |
| RFC 1372 | Telnet Remote Flow Control Option C. L. Hedrick, D. Borman |
| RFC 1374 | IP and ARP on HIPPI J. Renwick, A. Nicholson |
| RFC 1381 | SNMP MIB Extension for X.25 LAPB D. Throop, F. Baker |
| RFC 1382 | SNMP MIB Extension for the X.25 Packet Layer D. Throop |
| RFC 1387 | RIP Version 2 Protocol Analysis G. Malkin |
| RFC 1388 | RIP Version 2 Carrying Additional Information G. Malkin |
RFC 1389
RIP Version 2 MIB Extensions G. Malkin, F. Baker

RFC 1390
Transmission of IP and ARP over FDDI Networks D. Katz

RFC 1393
Traceroute Using an IP Option G. Malkin

RFC 1398
Definitions of Managed Objects for the Ethernet-Like Interface Types F. Kastenholz

RFC 1408
Telnet Environment Option D. Borman, Ed.

RFC 1413
Identification Protocol M. St. Johns

RFC 1416
Telnet Authentication Option D. Borman, ed.

RFC 1420
SNMP over IPX S. Bostock

RFC 1428
Transition of Internet Mail from Just-Send-8 to 8bit-SMTP/MIME G. Vaudreuil

RFC 1442

RFC 1443

RFC 1445
Administrative Model for version 2 of the Simple Network Management Protocol (SNMPv2) J. Galvin, K. McCloghrie

RFC 1447
Party MIB for version 2 of the Simple Network Management Protocol (SNMPv2) K. McCloghrie, J. Galvin

RFC 1448

RFC 1464
Using the Domain Name System to Store Arbitrary String Attributes R. Rosenbaum

RFC 1469
IP Multicast over Token-Ring Local Area Networks T. Pusateri

RFC 1483
Multiprotocol Encapsulation over ATM Adaptation Layer 5 Juha Heinanen

RFC 1514
Host Resources MIB P. Grillo, S. Waldbusser

RFC 1516
Definitions of Managed Objects for IEEE 802.3 Repeater Devices D. McMaster, K. McCloghrie
RFC 1521
MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies N. Borenstein, N. Freed

RFC 1535
A Security Problem and Proposed Correction With Widely Deployed DNS Software E. Gavron

RFC 1536
Common DNS Implementation Errors and Suggested Fixes A. Kumar, J. Postel, C. Neuman, P. Danzig, S. Miller

RFC 1537
Common DNS Data File Configuration Errors P. Beertema

RFC 1540
Internet Official Protocol Standards J. Postel

RFC 1571
Telnet Environment Option Interoperability Issues D. Borman

RFC 1572
Telnet Environment Option S. Alexander

RFC 1573
Evolution of the Interfaces Group of MIB-II K. McCloghrie, F. Kastenholz

RFC 1577
Classical IP and ARP over ATM M. Laubach

RFC 1583
OSPF Version 2 J. Moy

RFC 1591
Domain Name System Structure and Delegation J. Postel

RFC 1592

RFC 1594
FYI on Questions and Answers—Answers to Commonly Asked “New Internet User” Questions A. Marine, J. Reynolds, G. Malkin

RFC 1644
T/TCP — TCP Extensions for Transactions Functional Specification R. Braden

RFC 1646
TN3270 Extensions for LUname and Printer Selection C. Graves, T. Butts, M. Angel

RFC 1647
TN3270 Enhancements B. Kelly

RFC 1652
SMTP Service Extension for 8bit-MIMEtransport J. Klensin, N. Freed, M. Rose, E. Stefferud, D. Crocker

RFC 1664
Using the Internet DNS to Distribute RFC1327 Mail Address Mapping Tables C. Allochio, A. Bonito, B. Cole, S. Giordano, R. Hagens

RFC 1693
An Extension to TCP: Partial Order Service T. Connolly, P. Amer, P. Conrad
RFC 1695
Definitions of Managed Objects for ATM Management Version 8.0 using SMIv2
M. Ahmed, K. Tesink

RFC 1701
Generic Routing Encapsulation (GRE) S. Hanks, T. Li, D. Farinacci, P. Traina

RFC 1702
Generic Routing Encapsulation over IPv4 networks S. Hanks, T. Li, D. Farinacci, P. Traina

RFC 1706
DNS NSAP Resource Records B. Manning, R. Colella

RFC 1712
DNS Encoding of Geographical Location C. Farrell, M. Schulze, S. Pleitner D. Baldoni

RFC 1713
Tools for DNS debugging A. Romao

RFC 1723
RIP Version 2—Carrying Additional Information G. Malkin

RFC 1752
The Recommendation for the IP Next Generation Protocol S. Bradner, A. Mankin

RFC 1766
Tags for the Identification of Languages H. Alvestrand

RFC 1771
A Border Gateway Protocol 4 (BGP-4) Y. Rekhter, T. Li

RFC 1794
DNS Support for Load Balancing T. Brisco

RFC 1819

RFC 1826
IP Authentication Header R. Atkinson

RFC 1828
IP Authentication using Keyed MD5 P. Metzger, W. Simpson

RFC 1829
The ESP DES-CBC Transform P. Karn, P. Metzger, W. Simpson

RFC 1830
SMTP Service Extensions for Transmission of Large and Binary MIME Messages G. Vaudreuil

RFC 1831

RFC 1832
XDR: External Data Representation Standard R. Srinivasan

RFC 1833
Binding Protocols for ONC RPC Version 2 R. Srinivasan

RFC 1850
OSPF Version 2 Management Information Base F. Baker, R. Coltun
RFC 1907

RFC 1908

RFC 1912
Common DNS Operational and Configuration Errors D. Barr

RFC 1918
Address Allocation for Private Internets Y. Rekhter, B. Moskowitz, D. Karrenberg, G.J. de Groot, E. Lear

RFC 1928

RFC 1930
Guidelines for creation, selection, and registration of an Autonomous System (AS) J. Hawkkinson, T. Bates

RFC 1939
Post Office Protocol-Version 3 J. Myers, M. Rose

RFC 1981
Path MTU Discovery for IP version 6 J. McCann, S. Deering, J. Mogul

RFC 1982
Serial Number Arithmetic R. Elz, R. Bush

RFC 1985
SMTP Service Extension for Remote Message Queue Starting J. De Winter

RFC 1995
Incremental Zone Transfer in DNS M. Ohta

RFC 1996
A Mechanism for Prompt Notification of Zone Changes (DNS NOTIFY) P. Vixie

RFC 2010
Operational Criteria for Root Name Servers B. Manning, P. Vixie

RFC 2011
SNMPv2 Management Information Base for the Internet Protocol using SMIv2 K. McCloghrie, Ed.

RFC 2012

RFC 2013
SNMPv2 Management Information Base for the User Datagram Protocol using SMIv2 K. McCloghrie, Ed.

RFC 2018
TCP Selective Acknowledgement Options M. Mathis, J. Mahdavi, S. Floyd, A. Romanow

RFC 2026
The Internet Standards Process — Revision 3 S. Bradner
RFC 2030
Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI D. Mills

RFC 2033
Local Mail Transfer Protocol J. Myers

RFC 2034
SMTP Service Extension for Returning Enhanced Error Codes N. Freed

RFC 2040

RFC 2045
Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies N. Freed, N. Borenstein

RFC 2052
A DNS RR for specifying the location of services (DNS SRV) A. Gulbrandsen, P. Vixie

RFC 2065
Domain Name System Security Extensions D. Eastlake 3rd, C. Kaufman

RFC 2066
TELNET CHARSET Option R. Gellens

RFC 2080
RIPng for IPv6 G. Malkin, R. Minnear

RFC 2096
IP Forwarding Table MIB F. Baker

RFC 2104
HMAC: Keyed-Hashing for Message Authentication H. Krawczyk, M. Bellare, R. Canetti

RFC 2119
Keywords for use in RFCs to Indicate Requirement Levels S. Bradner

RFC 2133
Basic Socket Interface Extensions for IPv6 R. Gilligan, S. Thomson, J. Bound, W. Stevens

RFC 2136
Dynamic Updates in the Domain Name System (DNS UPDATE) P. Vixie, Ed., S. Thomson, Y. Rekhter, J. Bound

RFC 2137
Secure Domain Name System Dynamic Update D. Eastlake 3rd

RFC 2163
Using the Internet DNS to Distribute MIXER Conformant Global Address Mapping (MCGAM) C. Allocchio

RFC 2168
Resolution of Uniform Resource Identifiers using the Domain Name System R. Daniel, M. Meallling

RFC 2178
OSPF Version 2 J. Moy

RFC 2181
Clarifications to the DNS Specification R. Elz, R. Bush
RFC 2205

RFC 2210
The Use of RSVP with IETF Integrated Services J. Wroclawski

RFC 2211
Specification of the Controlled-Load Network Element Service J. Wroclawski

RFC 2212
Specification of Guaranteed Quality of Service S. Shenker, C. Partridge, R. Guerin

RFC 2215
General Characterization Parameters for Integrated Service Network Elements S. Shenker, J. Wroclawski

RFC 2217
Telnet Com Port Control Option G. Clarke

RFC 2219
Use of DNS Aliases for Network Services M. Hamilton, R. Wright

RFC 2228
FTP Security Extensions M. Horowitz, S. Lunt

RFC 2230
Key Exchange Delegation Record for the DNS R. Atkinson

RFC 2233
The Interfaces Group MIB using SMIv2 K. McCloghrie, F. Kastenholz

RFC 2240
A Legal Basis for Domain Name Allocation O. Vaughn

RFC 2246
The TLS Protocol Version 1.0 T. Dierks, C. Allen

RFC 2251

RFC 2253

RFC 2254
The String Representation of LDAP Search Filters T. Howes

RFC 2261
An Architecture for Describing SNMP Management Frameworks D. Harrington, R. Presuhn, B. Wijnen

RFC 2262

RFC 2271
An Architecture for Describing SNMP Management Frameworks D. Harrington, R. Presuhn, B. Wijnen

RFC 2273
SNMPv3 Applications D. Levi, P. Meyer, B. Stewart
RFC 2274
User-based Security Model (USM) for version 3 of the Simple Network
Management Protocol (SNMPv3) U. Blumenthal, B. Wijnen

RFC 2275
View-based Access Control Model (VACM) for the Simple Network Management
Protocol (SNMP) B. Wijnen, R. Presuhn, K. McCloghrie

RFC 2279
UTF-8, a transformation format of ISO 10646 F. Yergeau

RFC 2292
Advanced Sockets API for IPv6 W. Stevens, M. Thomas

RFC 2308
Negative Caching of DNS Queries (DNS NCACHE) M. Andrews

RFC 2317
Classless IN-ADDR.ARPA delegation H. Eidnes, G. de Groot, P. Vixie

RFC 2320
Definitions of Managed Objects for Classical IP and ARP Over ATM Using
SMIv2 (IPOA-MIB) M. Greene, J. Luciani, K. White, T. Kuo

RFC 2328
OSPF Version 2 J. Moy

RFC 2345
Domain Names and Company Name Retrieval J. Klensin, T. Wolf, G. Oglesby

RFC 2352
A Convention for Using Legal Names as Domain Names O. Vaughn

RFC 2355
TN3270 Enhancements B. Kelly

RFC 2358
Definitions of Managed Objects for the Ethernet-like Interface Types J. Flick, J.
Johnson

RFC 2373
IP Version 6 Addressing Architecture R. Hinden, S. Deering

RFC 2374
An IPv6 Aggregatable Global Unicast Address Format R. Hinden, M. O'Dell, S.
Deering

RFC 2375
IPv6 Multicast Address Assignments R. Hinden, S. Deering

RFC 2385
Protection of BGP Sessions via the TCP MD5 Signature Option A. Hefferman

RFC 2389
Feature negotiation mechanism for the File Transfer Protocol P. Hethmon, R. Elz

RFC 2401
Security Architecture for Internet Protocol S. Kent, R. Atkinson

RFC 2402
IP Authentication Header S. Kent, R. Atkinson

RFC 2403
The Use of HMAC-MD5–96 within ESP and AH C. Madson, R. Glenn
Appendix D. Related protocol specifications
RFC 2523
Photuris: Extended Schemes and Attributes P. Karn, W. Simpson

RFC 2535
Domain Name System Security Extensions D. Eastlake 3rd

RFC 2538
Storing Certificates in the Domain Name System (DNS) D. Eastlake 3rd, O. Gudmundsson

RFC 2539
Storage of Diffie-Hellman Keys in the Domain Name System (DNS) D. Eastlake 3rd

RFC 2540
Detached Domain Name System (DNS) Information D. Eastlake 3rd

RFC 2554
SMTP Service Extension for Authentication J. Myers

RFC 2570
Introduction to Version 3 of the Internet-standard Network Management Framework J. Case, R. Mundy, D. Partain, B. Stewart

RFC 2571
An Architecture for Describing SNMP Management Frameworks B. Wijnen, D. Harrington, R. Presuhn

RFC 2572

RFC 2573
SNMP Applications D. Levi, P. Meyer, B. Stewart

RFC 2574

RFC 2575
View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP) B. Wijnen, R. Presuhn, K. McCloghrie

RFC 2576

RFC 2578

RFC 2579
Textual Conventions for SMIv2 K. McCloghrie, D. Perkins, J. Schoenwaelder

RFC 2580
Conformance Statements for SMIv2 K. McCloghrie, D. Perkins, J. Schoenwaelder

RFC 2581
TCP Congestion Control M. Allman, V. Paxson, W. Stevens

RFC 2583
Guidelines for Next Hop Client (NHC) Developers R. Carlson, L. Winkler
<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
<th>Authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>2591</td>
<td>Definitions of Managed Objects for Scheduling Management Operations</td>
<td>D. Levi, J. Schoenwaelder</td>
</tr>
<tr>
<td>2625</td>
<td>IP and ARP over Fibre Channel</td>
<td>M. Rajagopal, R. Bhagwat, W. Rickard</td>
</tr>
<tr>
<td>2635</td>
<td>Don’t SPEW A Set of Guidelines for Mass Unsolicited Mailings and Postings (spam*)</td>
<td>S. Hambridge, A. Lunde</td>
</tr>
<tr>
<td>2637</td>
<td>Point-to-Point Tunneling Protocol</td>
<td>K. Hamzeh, G. Pall, W. Verthein, J. Taarud,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>W. Little, G. Zorn</td>
</tr>
<tr>
<td>2640</td>
<td>Internationalization of the File Transfer Protocol</td>
<td>B. Curtin</td>
</tr>
<tr>
<td>2665</td>
<td>Definitions of Managed Objects for the Ethernet-like Interface Types</td>
<td>J. Flick, J. Johnson</td>
</tr>
<tr>
<td>2671</td>
<td>Extension Mechanisms for DNS (EDNS0)</td>
<td>P. Vixie</td>
</tr>
<tr>
<td>2672</td>
<td>Non-Terminal DNS Name Redirection</td>
<td>M. Crawford</td>
</tr>
<tr>
<td>2675</td>
<td>IPv6 Jumbograms</td>
<td>D. Borman, S. Deering, R. Hinden</td>
</tr>
<tr>
<td>2710</td>
<td>Multicast Listener Discovery (MLD) for IPv6</td>
<td>S. Deering, W. Fenner, B. Haberman</td>
</tr>
<tr>
<td>2711</td>
<td>IPv6 Router Alert Option</td>
<td>C. Partridge, A. Jackson</td>
</tr>
<tr>
<td>2740</td>
<td>OSPF for IPv6</td>
<td>R. Coltun, D. Ferguson, J. Moy</td>
</tr>
<tr>
<td>2753</td>
<td>A Framework for Policy-based Admission Control</td>
<td>R. Yavatkar, D. Pendarakis, R. Guerin</td>
</tr>
<tr>
<td>2782</td>
<td>A DNS RR for specifying the location of services (DNS SRV)</td>
<td>A. Gubrandsen, P. Vixix, L. Esibov</td>
</tr>
<tr>
<td>2821</td>
<td>Simple Mail Transfer Protocol</td>
<td>J. Klensin, Ed.</td>
</tr>
<tr>
<td>2822</td>
<td>Internet Message Format</td>
<td>P. Resnick, Ed.</td>
</tr>
<tr>
<td>2840</td>
<td>TELNET KERMIT OPTION</td>
<td>J. Altman, F. da Cruz</td>
</tr>
<tr>
<td>2845</td>
<td>Secret Key Transaction Authentication for DNS (TSIG)</td>
<td>P. Vixie, O. Gudmundsson, D. Eastlake 3rd, B. Wellington</td>
</tr>
<tr>
<td>2851</td>
<td>Textual Conventions for Internet Network Addresses</td>
<td>M. Daniele, B. Haberman, S. Routhier, J. Schoenwaelder</td>
</tr>
</tbody>
</table>
RFC 2852
Deliver By SMTP Service Extension D. Newman

RFC 2874
DNS Extensions to Support IPv6 Address Aggregation and Renumbering M. Crawford, C. Huitema

RFC 2915
The Naming Authority Pointer (NAPTR) DNS Resource Record M. Mealling, R. Daniel

RFC 2920
SMTP Service Extension for Command Pipelining N. Freed

RFC 2930
Secret Key Establishment for DNS (TKEY RR) D. Eastlake, 3rd

RFC 2941
Telnet Authentication Option T. Ts'o, ed., J. Altman

RFC 2942
Telnet Authentication: Kerberos Version 5 T. Ts'o

RFC 2946
Telnet Data Encryption Option T. Ts'o

RFC 2952
Telnet Encryption: DES 64 bit Cipher Feedback T. Ts'o

RFC 2953
Telnet Encryption: DES 64 bit Output Feedback T. Ts'o

RFC 2992
Analysis of an Equal-Cost Multi-Path Algorithm C. Hopps

RFC 3019
IP Version 6 Management Information Base for The Multicast Listener Discovery Protocol B. Haberman, R. Worzella

RFC 3060
Policy Core Information Model—Version 1 Specification B. Moore, E. Ellesson, J. Strassner, A. Westerinen

RFC 3152
Delegation of IP6.ARPA R. Bush

RFC 3164
The BSD Syslog Protocol C. Lonvick

RFC 3207
SMTP Service Extension for Secure SMTP over Transport Layer Security P. Hoffman

RFC 3226
DNSSEC and IPv6 A6 aware server/resolver message size requirements O. Gudmundsson

RFC 3291
Textual Conventions for Internet Network Addresses M. Daniele, B. Haberman, S. Routhier, J. Schoenwaelder

RFC 3363
RFC 3542

Advanced Sockets Application Programming Interface (API) for IPv6 W. Richard Stevens, M. Thomas, E. Nordmark, T. Jinmei

RFC 3566

The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec S. Frankel, H. Herbert

RFC 3569

An Overview of Source-Specific Multicast (SSM) S. Bhattacharyya, Ed.

RFC 3584


RFC 3602

The AES-CBC Cipher Algorithm and Its Use with IPsec S. Frankel, R. Glenn, S. Kelly

RFC 3629

UTF-8, a transformation format of ISO 10646 R. Kermode, C. Vicisano

RFC 3658

Delegation Signer (DS) Resource Record (RR) O. Gudmundsson

RFC 3678

Socket Interface Extensions for Multicast Source Filters D. Thaler, B. Fenner, B. Quinn

RFC 3715

IPsec-Network Address Translation (NAT) Compatibility Requirements B. Aboba, W. Dixon

RFC 3810


RFC 3826


RFC 3947

Negotiation of NAT-Traversal in the IKE T. Kivinen, B. Swander, A. Huttunen, V. Volpe

RFC 3948

UDP Encapsulation of IPsec ESP Packets A. Huttunen, B. Swander, V. Volpe, L. DiBurro, M. Stenberg

RFC 4001

Textual Conventions for Internet Network Addresses M. Daniele, B. Haberman, S. Routhier, J. Schoenwaelder

RFC 4007

IPv6 Scoped Address Architecture S. Deering, B. Haberman, T. Jinmei, E. Nordmark, B. Zill

RFC 4022

Management Information Base for the Transmission Control Protocol (TCP) R. Raghunarayan

RFC 4106

The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP) J. Viega, D. McGrew
RFC 4109
Algorithms for Internet Key Exchange version 1 (IKEv1) P. Hoffman

RFC 4113
Management Information Base for the User Datagram Protocol (UDP) B. Fenner, J. Flick

RFC 4191
Default Router Preferences and More-Specific Routes R. Draves, D. Thaler

RFC 4217
Securing FTP with TLS P. Ford-Hutchinson

RFC 4292
IP Forwarding Table MIB B. Haberman

RFC 4293
Management Information Base for the Internet Protocol (IP) S. Routhier

RFC 4301
Security Architecture for the Internet Protocol S. Kent, K. Seo

RFC 4302
IP Authentication Header S. Kent

RFC 4303
IP Encapsulating Security Payload (ESP) S. Kent

RFC 4304
Extended Sequence Number (ESN) Addendum to IPsec Domain of Interpretation (DOI) for Internet Security Association and Key Management Protocol (ISAKMP) S. Kent

RFC 4307
Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2) J. Schiller

RFC 4308
Cryptographic Suites for IPsec P. Hoffman

RFC 4434
The AES-XCBC-PRF-128 Algorithm for the Internet Key Exchange Protocol P. Hoffman

RFC 4443
Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification A. Conta, S. Deering

RFC 4552
Authentication/Confidentiality for OSPFv3 M. Gupta, N. Melam

RFC 4678
Server/Application State Protocol v1 A. Bivens

RFC 4753
ECP Groups for IKE and IKEv2 D. Fu, J. Solinas

RFC 4754
IKE and IKEv2 Authentication Using the Elliptic Curve Digital Signature Algorithm (ECDSA) D. Fu, J. Solinas

RFC 4809
Internet drafts

Internet drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Other groups can also distribute working documents as Internet drafts. You can see Internet drafts at [http://www.ietf.org/ID.html](http://www.ietf.org/ID.html).
Appendix E. Architectural specifications

This appendix lists documents that provide architectural specifications for the SNA Protocol.

The APPN Implementers' Workshop (AIW) architecture documentation includes the following architectural specifications for SNA APPN and HPR:

- APPN Architecture Reference (SG30-3422-04)
- APPN Branch Extender Architecture Reference Version 1.1
- APPN Dependent LU Requester Architecture Reference Version 1.5
- APPN Extended Border Node Architecture Reference Version 1.0
- APPN High Performance Routing Architecture Reference Version 4.0
- SNA Formats (GA27-3136-20)
- SNA Technical Overview (GC30-3073-04)


The following RFC also contains SNA architectural specifications:

- RFC 2353 *APPN/HPR in IP Networks APPN Implementers' Workshop Closed Pages Document*

RFCs can be obtained from:

Government Systems, Inc.
Attn: Network Information Center
14200 Park Meadow Drive
Suite 200
Chantilly, VA 22021

Many RFCs are available online. Hardcopies of all RFCs are available from the NIC, either individually or by subscription. Online copies are available using FTP from the NIC at [http://www.rfc-editor.org/rfc.html](http://www.rfc-editor.org/rfc.html).

Use FTP to download the files, using the following format:

RFC:RFC-INDEX.TXT
RFC:RFCnnnn.TXT
RFC:RFCnnnn.PS

where:

- *nnnn* is the RFC number.
- TXT is the text format.
- PS is the postscript format.

You can also request RFCs through electronic mail, from the automated NIC mail server, by sending a message to service@nic.ddn.mil with a subject line of RFC *nnnn* for text versions or a subject line of RFC *nnnn*.PS for PostScript versions.

To request a copy of the RFC index, send a message with a subject line of RFC INDEX.
For more information, contact nic@nic.ddn.mil.
Appendix F. Accessibility

Publications for this product are offered in Adobe Portable Document Format (PDF) and should be compliant with accessibility standards. If you experience difficulties when using PDF files, you can view the information through the z/OS Internet Library website or IBM Knowledge Center. If you continue to experience problems, send an email to mhvrcfs@us.ibm.com or write to:

IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/OS enable users to:

• Use assistive technologies such as screen readers and screen magnifier software
• Operate specific or equivalent features using only the keyboard
• Customize display attributes such as color, contrast, and font size

Using assistive technologies

Assistive technology products, such as screen readers, function with the user interfaces found in z/OS. Consult the assistive technology documentation for specific information when using such products to access z/OS interfaces.

Keyboard navigation of the user interface

Users can access z/OS user interfaces using TSO/E or ISPF. See z/OS TSO/E Primer and z/OS ISPF User’s Guide Vol 1 for information about accessing TSO/E and ISPF interfaces. These guides describe how to use TSO/E and ISPF, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

z/OS information

z/OS information is accessible using screen readers with the BookServer or Library Server versions of z/OS books in the Internet library at www.ibm.com/systems/z/os/zos/bkserv/

One exception is command syntax that is published in railroad track format, which is accessible using screen readers with IBM Knowledge Center, as described in “Dotted decimal syntax diagrams.”

Dotted decimal syntax diagrams

Syntax diagrams are provided in dotted decimal format for users accessing IBM Knowledge Center using a screen reader. In dotted decimal format, each syntax element is written on a separate line. If two or more syntax elements are always...
present together (or always absent together), they can appear on the same line, because they can be considered as a single compound syntax element.

Each line starts with a dotted decimal number; for example, 3 or 3.1 or 3.1.1. To hear these numbers correctly, make sure that your screen reader is set to read out punctuation. All the syntax elements that have the same dotted decimal number (for example, all the syntax elements that have the number 3.1) are mutually exclusive alternatives. If you hear the lines 3.1 USERID and 3.1 SYSTEMID, you know that your syntax can include either USERID or SYSTEMID, but not both.

The dotted decimal numbering level denotes the level of nesting. For example, if a syntax element with dotted decimal number 3 is followed by a series of syntax elements with dotted decimal number 3.1, all the syntax elements numbered 3.1 are subordinate to the syntax element numbered 3.

Certain words and symbols are used next to the dotted decimal numbers to add information about the syntax elements. Occasionally, these words and symbols might occur at the beginning of the element itself. For ease of identification, if the word or symbol is a part of the syntax element, it is preceded by the backslash (\) character. The * symbol can be used next to a dotted decimal number to indicate that the syntax element repeats. For example, syntax element *FILE with dotted decimal number 3 is given the format 3 \* FILE. Format 3* FILE indicates that syntax element FILE repeats. Format 3* \* FILE indicates that syntax element * FILE repeats.

Characters such as commas, which are used to separate a string of syntax elements, are shown in the syntax just before the items they separate. These characters can appear on the same line as each item, or on a separate line with the same dotted decimal number as the relevant items. The line can also show another symbol giving information about the syntax elements. For example, the lines 5.1*, 5.1 LASTRUN, and 5.1 DELETE mean that if you use more than one of the LASTRUN and DELETE syntax elements, the elements must be separated by a comma. If no separator is given, assume that you use a blank to separate each syntax element.

If a syntax element is preceded by the % symbol, this indicates a reference that is defined elsewhere. The string following the % symbol is the name of a syntax fragment rather than a literal. For example, the line 2.1 %OP1 means that you should see separate syntax fragment OP1.

The following words and symbols are used next to the dotted decimal numbers:
- A question mark (?) means an optional syntax element. A dotted decimal number followed by the ? symbol indicates that all the syntax elements with a corresponding dotted decimal number, and any subordinate syntax elements, are optional. If there is only one syntax element with a dotted decimal number, the ? symbol is displayed on the same line as the syntax element, (for example 5? NOTIFY). If there is more than one syntax element with a dotted decimal number, the ? symbol is displayed on a line by itself, followed by the syntax elements that are optional. For example, if you hear the lines 5 ?, 5 NOTIFY, and 5 UPDATE, you know that syntax elements NOTIFY and UPDATE are optional; that is, you can choose one or none of them. The ? symbol is equivalent to a bypass line in a railroad diagram.
- An exclamation mark (!) means a default syntax element. A dotted decimal number followed by the ! symbol and a syntax element indicate that the syntax element is the default option for all syntax elements that share the same dotted
decimal number. Only one of the syntax elements that share the same dotted
decimal number can specify a ! symbol. For example, if you hear the lines 2?
FILE, 2.1! (KEEP), and 2.1 (DELETE), you know that (KEEP) is the default option
for the FILE keyword. In this example, if you include the FILE keyword but do
not specify an option, default option KEEP will be applied. A default option also
applies to the next higher dotted decimal number. In this example, if the FILE
keyword is omitted, default FILE(KEEP) is used. However, if you hear the lines
2? FILE, 2.1, 2.1.1! (KEEP), and 2.1.1 (DELETE), the default option KEEP applies
only to the next higher dotted decimal number, 2.1 (which does not have an
associated keyword), and does not apply to 2? FILE. Nothing is used if the
keyword FILE is omitted.

• An asterisk (*) means a syntax element that can be repeated 0 or more times. A
dotted decimal number followed by the * symbol indicates that this syntax
element can be used zero or more times; that is, it is optional and can be
repeated. For example, if you hear the line 5.1* data area, you know that you
can include one data area, more than one data area, or no data area. If you hear
the lines 3*, 3 HOST, and 3 STATE, you know that you can include HOST,
STATE, both together, or nothing.

Notes:
1. If a dotted decimal number has an asterisk (*) next to it and there is only one
item with that dotted decimal number, you can repeat that same item more
than once.
2. If a dotted decimal number has an asterisk next to it and several items have
that dotted decimal number, you can use more than one item from the list,
but you cannot use the items more than once each. In the previous example,
you could write HOST STATE, but you could not write HOST HOST.
3. The * symbol is equivalent to a loop-back line in a railroad syntax diagram.

• + means a syntax element that must be included one or more times. A dotted
decimal number followed by the + symbol indicates that this syntax element
must be included one or more times; that is, it must be included at least once
and can be repeated. For example, if you hear the line 6.1+ data area, you must
include at least one data area. If you hear the lines 2+, 2 HOST, and 2 STATE,
you know that you must include HOST, STATE, or both. Similar to the * symbol,
the + symbol can only repeat a particular item if it is the only item with that
dotted decimal number. The + symbol, like the * symbol, is equivalent to a
loop-back line in a railroad syntax diagram.
Notices

This information was developed for products and services offered in the USA.

IBM may not offer all of the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user’s responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.
IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who want to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Site Counsel
IBM Corporation
P.O. Box 12195
3039 Cornwallis Road
Research Triangle Park, North Carolina 27709-2195
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM’s future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

If you are viewing this information softcopy, the photographs and color illustrations might not appear.

COPYRIGHT LICENSE:

This information might contain sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing
application programs conforming to the application programming interface for the
operating platform for which the sample programs are written. These examples
have not been thoroughly tested under all conditions. IBM, therefore, cannot
guarantee or imply reliability, serviceability, or function of these programs. The
sample programs are provided "AS IS", without warranty of any kind. IBM shall
not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must
include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp.
Sample Programs. © Copyright IBM Corp. _enter the year or years_.

IBM is required to include the following statements in order to distribute portions
of this document and the software described herein to which contributions have
been made by The University of California. Portions herein © Copyright 1979,
permission. Portions herein were developed at the Electrical Engineering and
Computer Sciences Department at the Berkeley campus of the University of
California under the auspices of the Regents of the University of California.

Portions of this publication relating to RPC are Copyright © Sun Microsystems,

Some portions of this publication relating to X Window System** are Copyright ©
1987, 1988 by Digital Equipment Corporation, Maynard, Massachusetts, and the
Massachusetts Institute Of Technology, Cambridge, Massachusetts.

Some portions of this publication relating to X Window System are Copyright ©

Permission to use, copy, modify, and distribute the M.I.T., Digital Equipment
Corporation, and Hewlett-Packard Corporation portions of this software and its
documentation for any purpose without fee is hereby granted, provided that the
above copyright notice appears in all copies and that both that copyright notice
and this permission notice appear in supporting documentation, and that the
names of M.I.T., Digital, and Hewlett-Packard not be used in advertising or
publicity pertaining to distribution of the software without specific, written prior
permission. M.I.T., Digital, and Hewlett-Packard make no representation about the
suitability of this software for any purpose. It is provided "as is" without express
or implied warranty.


Copyright © 1988, 1993 The Regents of the University of California.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list
   of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this
   list of conditions and the following disclaimer in the documentation and/or
   other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must
   display the following acknowledgment:
This product includes software developed by the University of California, Berkeley and its contributors.

4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software program contains code, and/or derivatives or modifications of code originating from the software program "Popper." Popper is Copyright ©1989-1991 The Regents of the University of California. Popper was created by Austin Shelton, Information Systems and Technology, University of California, Berkeley.

Permission from the Regents of the University of California to use, copy, modify, and distribute the "Popper" software contained herein for any purpose, without fee, and without a written agreement is hereby granted, provided that the above copyright notice and this paragraph and the following two paragraphs appear in all copies. HOWEVER, ADDITIONAL PERMISSIONS MAY BE NECESSARY FROM OTHER PERSONS OR ENTITIES, TO USE DERIVATIVES OR MODIFICATIONS OF POPPER.

IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THE POPPER SOFTWARE, OR ITS DERIVATIVES OR MODIFICATIONS, AND ITS DOCUMENTATION, EVEN IF THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE POPPER SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF CALIFORNIA HAS NO OBLIGATIONS TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

Copyright © 1983 The Regents of the University of California.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the University of California, Berkeley. The name of the University may not be used to endorse or promote products derived from this software without specific prior
written permission. THIS SOFTWARE IS PROVIDED ``AS IS'' AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Copyright © 1991, 1993 The Regents of the University of California.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:
   This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Furthermore if you modify this software you must label your software as modified software and not distribute it in such a fashion that it might be confused with the original M.I.T. software. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Copyright © 1998 by the FundsXpress, INC.
Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FundsXpress not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. FundsXpress makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.


Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright © 1995-1998 Eric Young (eay@cryptsoft.com)

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscape's SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young’s, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)". The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related.

4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include acknowledgment:
   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The license and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution license [including the GNU Public License.]

This product includes cryptographic software written by Eric Young.


Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright © 2004 IBM Corporation and its licensors, including Sendmail, Inc., and the Regents of the University of California.

Copyright © 1999,2000,2001 Compaq Computer Corporation

Copyright © 1999,2000,2001 Hewlett-Packard Company
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

X Window System is a trademark of The Open Group.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

You can obtain softcopy from the z/OS Collection (SK3T-4269), which contains BookManager and PDF formats.

**Minimum supported hardware**

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see [http://www-01.ibm.com/software/support/systemsz/lifecycle/](http://www-01.ibm.com/software/support/systemsz/lifecycle/)
- For information about currently-supported IBM hardware, contact your IBM representative.
Policy for unsupported hardware

Various z/OS elements, such as DFSMS, HCD, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.

Intel is a registered trademark of Intel Corporation or its subsidiaries in the United States and other countries.

Java™ and all Java-based trademarks are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

Adobe and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other product and service names might be trademarks of IBM or other companies.
Bibliography

This bibliography contains descriptions of the documents in the z/OS Communications Server library.

z/OS Communications Server documentation is available in the following forms:
- In softcopy on CD-ROM collections. See “Softcopy information” on page xxiv.

z/OS Communications Server library updates


z/OS Communications Server information

z/OS Communications Server product information is grouped by task in the following tables.

### Planning

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: New Function Summary</td>
<td>GC27-3664</td>
<td>This document is intended to help you plan for new IP or SNA function, whether you are migrating from a previous version or installing z/OS for the first time. It summarizes what is new in the release and identifies the suggested and required modifications needed to use the enhanced functions.</td>
</tr>
<tr>
<td>z/OS Communications Server: IPv6 Network and Application Design Guide</td>
<td>SC27-3663</td>
<td>This document is a high-level introduction to IPv6. It describes concepts of z/OS Communications Server’s support of IPv6, coexistence with IPv4, and migration issues.</td>
</tr>
</tbody>
</table>

### Resource definition, configuration, and tuning

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: IP Configuration Guide</td>
<td>SC27-3650</td>
<td>This document describes the major concepts involved in understanding and configuring an IP network. Familiarity with the z/OS operating system, IP protocols, z/OS UNIX System Services, and IBM Time Sharing Option (TSO) is recommended. Use this document with the z/OS Communications Server: IP Configuration Reference.</td>
</tr>
</tbody>
</table>
### Title Number Description

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
</table>
| z/OS Communications Server: IP Configuration Reference | SC27-3651 | This document presents information for people who want to administer and maintain IP. Use this document with the z/OS Communications Server: IP Configuration Guide. The information in this document includes:  
  - TCP/IP configuration data sets  
  - Configuration statements  
  - Translation tables  
  - Protocol number and port assignments |
| z/OS Communications Server: SNA Network Implementation Guide | SC27-3672 | This document presents the major concepts involved in implementing an SNA network. Use this document with the z/OS Communications Server: SNA Resource Definition Reference. |
| z/OS Communications Server: SNA Resource Definition Reference | SC27-3675 | This document describes each SNA definition statement, start option, and macroinstruction for user tables. It also describes NCP definition statements that affect SNA. Use this document with the z/OS Communications Server: SNA Network Implementation Guide. |
| z/OS Communications Server: SNA Resource Definition Samples | SC27-3676 | This document contains sample definitions to help you implement SNA functions in your networks, and includes sample major node definitions. |
| z/OS Communications Server: IP Network Print Facility | SC27-3658 | This document is for systems programmers and network administrators who need to prepare their network to route SNA, JES2, or JES3 printer output to remote printers using TCP/IP Services. |

### Operation

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: IP User's Guide and Commands</td>
<td>SC27-3662</td>
<td>This document describes how to use TCP/IP applications. It contains requests with which a user can log on to a remote host using Telnet, transfer data sets using FTP, send and receive electronic mail, print on remote printers, and authenticate network users.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP System Administrator's Commands</td>
<td>SC27-3661</td>
<td>This document describes the functions and commands helpful in configuring or monitoring your system. It contains system administrator's commands, such as TSO NETSTAT, PING, TRACERTE and their UNIX counterparts. It also includes TSO and MVS commands commonly used during the IP configuration process.</td>
</tr>
<tr>
<td>z/OS Communications Server: SNA Operation</td>
<td>SC27-3673</td>
<td>This document serves as a reference for programmers and operators requiring detailed information about specific operator commands.</td>
</tr>
<tr>
<td>z/OS Communications Server: Quick Reference</td>
<td>SC27-3665</td>
<td>This document contains essential information about SNA and IP commands.</td>
</tr>
</tbody>
</table>
## Customization

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>z/OS Communications Server:</strong></td>
<td></td>
<td><strong>SNA Customization</strong></td>
</tr>
<tr>
<td></td>
<td>SC27-3666</td>
<td>This document enables you to customize SNA, and includes the following information:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Communication network management (CNM) routing table</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Logon-interpret routine requirements</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Logon manager installation-wide exit routine for the CLU search exit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• TSO/SNA installation-wide exit routines</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SNA installation-wide exit routines</td>
</tr>
</tbody>
</table>

## Writing application programs

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>z/OS Communications Server:</strong></td>
<td></td>
<td><strong>IP Sockets Application Programming Interface Guide and Reference</strong></td>
</tr>
<tr>
<td></td>
<td>SC27-3660</td>
<td>This document describes the syntax and semantics of program source code necessary to write your own application programming interface (API) into TCP/IP. You can use this interface as the communication base for writing your own client or server application. You can also use this document to adapt your existing applications to communicate with each other using sockets over TCP/IP.</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong></td>
<td></td>
<td><strong>IP CICS Sockets Guide</strong></td>
</tr>
<tr>
<td></td>
<td>SC27-3649</td>
<td>This document is for programmers who want to set up, write application programs for, and diagnose problems with the socket interface for CICS® using z/OS TCP/IP.</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong></td>
<td></td>
<td><strong>IP IMS Sockets Guide</strong></td>
</tr>
<tr>
<td></td>
<td>SC27-3653</td>
<td>This document is for programmers who want application programs that use the IMS™ TCP/IP application development services provided by the TCP/IP Services of IBM.</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong></td>
<td></td>
<td><strong>IP Programmer’s Guide and Reference</strong></td>
</tr>
<tr>
<td></td>
<td>SC27-3659</td>
<td>This document describes the syntax and semantics of a set of high-level application functions that you can use to program your own applications in a TCP/IP environment. These functions provide support for application facilities, such as user authentication, distributed databases, distributed processing, network management, and device sharing. Familiarity with the z/OS operating system, TCP/IP protocols, and IBM Time Sharing Option (TSO) is recommended.</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong></td>
<td></td>
<td>SNA Programming**</td>
</tr>
<tr>
<td></td>
<td>SC27-3674</td>
<td>This document describes how to use SNA macroinstructions to send data to and receive data from (1) a terminal in either the same or a different domain, or (2) another application program in either the same or a different domain.</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong></td>
<td></td>
<td><strong>SNA Programmer’s LU 6.2 Guide</strong></td>
</tr>
<tr>
<td></td>
<td>SC27-3669</td>
<td>This document describes how to use the SNA LU 6.2 application programming interface for host application programs. This document applies to programs that use only LU 6.2 sessions or that use LU 6.2 sessions along with other session types. (Only LU 6.2 sessions are covered in this document.)</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong></td>
<td></td>
<td><strong>SNA Programmer’s LU 6.2 Reference</strong></td>
</tr>
<tr>
<td></td>
<td>SC27-3670</td>
<td>This document provides reference material for the SNA LU 6.2 programming interface for host application programs.</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong></td>
<td></td>
<td><strong>CSM Guide</strong></td>
</tr>
<tr>
<td></td>
<td>SC27-3647</td>
<td>This document describes how applications use the communications storage manager.</td>
</tr>
</tbody>
</table>
## Title, Number, Description

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>z/OS Communications Server:</strong> CMIP Services and Topology Agent Guide</td>
<td>SC27-3646</td>
<td>This document describes the Common Management Information Protocol (CMIP) programming interface for application programmers to use in coding CMIP application programs. The document provides guide and reference information about CMIP services and the SNA topology agent.</td>
</tr>
</tbody>
</table>

### Diagnosis

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>z/OS Communications Server:</strong> IP Diagnosis Guide</td>
<td>GC27-3652</td>
<td>This document explains how to diagnose TCP/IP problems and how to determine whether a specific problem is in the TCP/IP product code. It explains how to gather information for and describe problems to the IBM Software Support Center.</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong> ACF/TAP Trace Analysis Handbook</td>
<td>GC27-3645</td>
<td>This document explains how to gather the trace data that is collected and stored in the host processor. It also explains how to use the Advanced Communications Function/Trace Analysis Program (ACF/TAP) service aid to produce reports for analyzing the trace data information.</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong> SNA Diagnosis Vol 1, Techniques and Procedures and SNA Diagnosis Vol 2, FFST Dumps and the VIT</td>
<td>GC27-3667 GC27-3668</td>
<td>These documents help you identify an SNA problem, classify it, and collect information about it before you call the IBM Support Center. The information collected includes traces, dumps, and other problem documentation.</td>
</tr>
<tr>
<td><strong>z/OS Communications Server:</strong> SNA Data Areas Volume 1 and SNA Data Areas Volume 2</td>
<td>GC31-6852 GC31-6853</td>
<td>These documents describe SNA data areas and can be used to read an SNA dump. They are intended for IBM programming service representatives and customer personnel who are diagnosing problems with SNA.</td>
</tr>
</tbody>
</table>

### Messages and codes

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
</table>
| **z/OS Communications Server:** SNA Messages            | SC27-3671 | This document describes the ELM, IKT, IST, IUT, IVT, and USS messages. Other information in this document includes:  
• Command and RU types in SNA messages  
• Node and ID types in SNA messages  
• Supplemental message-related information |
| **z/OS Communications Server:** IP Messages Volume 1 (EZA) | SC27-3654 | This volume contains TCP/IP messages beginning with EZA.                                                                                     |
| **z/OS Communications Server:** IP Messages Volume 2 (EZB, EZD) | SC27-3655 | This volume contains TCP/IP messages beginning with EZB or EZD.                                                                             |
| **z/OS Communications Server:** IP Messages Volume 3 (EZY) | SC27-3656 | This volume contains TCP/IP messages beginning with EZY.                                                                                     |
| **z/OS Communications Server:** IP Messages Volume 4 (EZZ, SNM) | SC27-3657 | This volume contains TCP/IP messages beginning with EZZ and SNM.                                                                            |
| **z/OS Communications Server:** IP and SNA Codes        | SC27-3648 | This document describes codes and other information that appear in z/OS Communications Server messages.                                        |
Index

A

ACBNAME operand (APPL definition statement) 41
accessibility 1403
ACTIVATE operand
dynamic reconfiguration 699
ADDRESS operand
NCP major node 451
PU definition statement
dynamic reconfiguration 700
model major node 369
switched major node 576
address of channel-attached NCP 92
ADDRESS operand
LINE definition statement
channel-to-channel 92
channel-to-NCP 92
ADJNET operand
GWPATH definition statement (CDRM major node) 152
ADJNETCS operand
GWPATH definition statement (CDRM major node) 152
ADJNETEL operand
GWPATH definition statement (CDRM major node) 153
ADJNETSA operand
GWPATH definition statement (CDRM major node) 153
ADJNET operand
GWPATH definition statement (CDRM major node) 153
ADVDELAY start option 794
alias translation
coding APPL definition statement 36
ALIASrch operand 21
ALL option (VTAM internal trace) 959
allocating outbound calls 579
ALSLIST operand (CDRSC definition statement) 120
ALSREQ operand
CDRSC definition statement 121
ALSREQ start option 794
ANS operand
NCP definition statements
takeover restriction 453
VTAM restrictions on NCP definition statements 453
PU (switched) definition statement
description 576
takeover restriction 369, 576
ANSWER operand 369
LINE definition statement
external communication adapter peripheral node 205
VTAM restrictions on NCP definition statements 454
API option (VTAM internal trace) 960
API64R start option 795
APPC operand
APPL definition statement 42
APPC option (VTAM internal trace) 960
APPC trace 960
APPL definition statement
alias translation facility 36
coding 36
logon manager 36
TSO/VTAM 36
application program interface (API), internal trace of 960
application program major node
defining 32
application programs
LU 6.2 34
non-LU 6.2 33
registering 32
application-supplied dynamic parameter (ASDP) 577
APPLID operand
LOGCHAR macroinstruction 1022
LOGOFF macroinstruction 1109
LOGON macroinstruction 1104
APPN cross-domain resources 105
APPN transmission group profile (TGP)
defining 720
described 5
IBM-supplied transmission group profiles 1356
IBMTGPS (member list containing the IBM-supplied TGP) 720
APPN-to-subarea class-of-service mapping table, defining 730
APPNCOs operand
MODEENT macroinstruction, logon mode table 1030
APPNCOs start option 796
ASDP operand
PU definition statement (switched major node) 577
ASIFMSG start option 797
ASLNT macroinstruction 982
ASLNT operand
APPL definition statement 43
CDRSC definition statement 122
LOCAL definition statement 276
LU definition statement
local SNA major node 298
LU group major node 343
model major node 370
switched major node 577
NCP definition statements 454
ASLPLU macroinstruction 982
ASLTAB macroinstruction 982
ASLTAB operand
APPL definition statement 43
CDRSC definition statement 122
LOCAL definition statement 277
LU definition statement
local SNA major node 298
LU group major node 344
model major node 370
switched major node 578
VTAM restrictions on NCP definition statements 455
ASRCVL operand 44
assembler features, restriction on use 16
associated LU table
description 981
installing 982
overview 5

© Copyright IBM Corp. 2000, 2015

1421
channel-attachment major node, defining 84
channel-to-channel support, defining 84
class of Service (COS), APPN 997
character-coded commands
conversion examples 1369
logon 1071
order of USS table use 1071
syntax 1102
CINDXSIZ start option 814
CIO option, VTAM internal trace 960
CKEYNAME operand 301, 344, 373, 581, 1031
LU definition statement 465
Class of Service (COS), APPN 997
APPNCOS definition statement
PRIORITY operand 1000
default values for APPN classes of service 1005
definitions 990
IBM-supplied APPN classes of service
COSAPPN (member list containing the IBM-supplied
COS) 1007
definition list 1313
described 1007
implementing APPN classes of service 991
Linerow definition statement 997
CAPACITY operand 992
COSTBYTE operand 994
COSTTIME operand 995
NUMBER operand 997
PDELAY operand 999
SECURITY operand 1002
UPARM1 operand 1002
UPARM2 operand 1003
UPARM3 operand 1003
WEIGHT operand 1004
Noderow definition statement
CONGEST operand 993
NUMBER operand 998
ROUTERES operand 1001
WEIGHT operand 1005
Class of Service (COS), subarea
SSCP (ISTVTTCOS) 986
table
described 985
unnamed entry 986
CLRSESSQ operand 345
local SNA major node 301
Model major node 373
Switched major node 581
VTAM restrictions on NCP definition statements 466
Cluster definition statement
NCP operands used by VTAM 428
CMD operand (USSCMD macroinstruction) 1081
CMIF option, VTAM internal trace 961
CMIF services directory definition file, coding 1008
CMIP VTAM internal trace of 961
CMPPAPPL operand
APPL definition statement 48
CMPPAPPO operand
APPL definition statement 49
CMPMIPS start option 815
CMPPVTAM start option 815
CNM (communication network management) application
programs 1061
CNMTAB start option 816
CNNRTMSG start option 816
CNOS (change number of sessions) request (APPL definition
statement) (continued)
DDRAINL operand 50
CNOS (change number of sessions) request (APPL definition
statement)
DMINWNL operand 51
DMINWNR operand 51
DRESPL operand 52
DSESFL operand 53
coding conventions for definition statements 8
coding VTAM-only operands on NCP definition
statements 427
COLD or WARM start option 817
comments in VTAM macroinstructions 13
communication network management (CNM)
application 1061
Communications Server for z/OS, online information xxvi
COMPRES operand (MODEENT macroinstruction, logon
mode table) 1032
COMPROT operand (MODEENT macroinstruction, logon
mode table) 1033
COMRAT operand
NCP major node 435
switched major node 558, 560
CONFDS operand
LBUILD definition statement 278
PCCU definition statement 466
VBUILD definition statement
CDRM major node 158
CDRSC major node 125
channel-attachment major node 93
local SNA major node 302
switched major node 582
CONFDPW operand
LBUILD definition statement 278
PCCU definition statement 467
VBUILD definition statement
CDRM major node 158
CDRSC major node 125
channel-attachment major node 93
local SNA major node 303
switched major node 583
Config start option 817
CONGEST operand (APPN Class of Service) 993
connection network
VNGROUP operand
NCP major node 544
XCA major node 267
VNNAME operand
NCP major node 544
XCA major node 268
CONNTYPE operand 212
PU definition statement
local SNA major node 303
model major node 374
NCP major node 467
switched major node 583
CONNTYPE start option 818
continuation character 13
conventions, coding for definition statements 8
COPIES operand (NETWORK definition statement) 468
COS (Class of Service), APPN 997
APPNCOS definition statement
PRIORITY operand 1000
default values for APPN classes of service 1005
definitions 990
IBM-supplied APPN classes of service
COSAPPN (member list containing the IBM-supplied
COS) 1007
COS (Class of Service), APPN (continued)
IBM-supplied APPN classes of service (continued)
definition list 1313
implemented APPN classes of service 991
LINEROW definition statement 997
  CAPACITY operand 992
  COSTBYTE operand 994
  COSTTIME operand 995
  NUMBER operand 997
  PDELAY operand 999
  SECURITY operand 1002
  UPARM1 operand 1002
  UPARM2 operand 1003
  UPARM3 operand 1003
  WEIGHT operand 1004
NODEROW definition statement
  CONGEST operand 993
  NUMBER operand 998
  ROUTERES operand 1001
  WEIGHT operand 1004
COS (Class of Service), subarea
SSCP (ISTVTCOS) 986
table
described 985
unnamed entry 986
COS macroinstruction 986
COS operand
  MAPSTO definition statement
    APPN-to-subarea COS mapping definitions 733, 737
    border node COS mapping definitions 741
  MODEENT macroinstruction 1033
COSEND macroinstruction 986
COSTAB macroinstruction 986
COSTAB operand
  BUILD definition statement (NCP major node) 468
  NETWORK definition statement (NCP major node) 468
COSTBYTE operand
  CDRM definition statement 158
  GROUP definition statement
    Enterprise Extender XCA major node 212
  LINEROW definition statement
  APPN Class of Service 994
  PU definition statement
    local SNA major node 304
    model major node 375
    NCP major node 468
    switched major node 584
  TGP definition statement
    transmission group profile 726
COSTTIME operand
  CDRM definition statement 159
  GROUP definition statement
    Enterprise Extender XCA major node 213
  LINEROW definition statement
  APPN Class of Service 995
  PU definition statement
    local SNA major node 304
    model major node 375
    NCP major node 469
    switched major node 584
  TGP definition statement
    transmission group profile 727
CPCDRS start option 819
CPCP operand (continued)
  PU definition statement (continued)
    model major node 376
    NCP major node 469
    switched major node 584
CPCP start option 820
CPNAME operand 215
  CDRSC definition statement (cross-domain resource major
  node) 125
  NEXTCP definition statement 717
  PU definition statement
    local SNA major node 305
    NCP major node 470
    switched major node 586
  cross-domain resource (CDRSC), defining 103
  cross-domain resource manager (CDRM) major node
defining in interconnected SNA network 145
defining in multiple-domain network 145
cross-domain resources for an APPN network 105
CSA24 start option 822
CSALIMIT start option 820
CSDUMP start option 823
CTCA, defining 84
CUADDR operand
  LOCAL definition statement (local non-SNA major
  node) 279
  PCCU definition statement 471
  PU definition statement (local SNA major node) 306
D
DATA operand (LOGON command) 1105
DATEDLM operand (USSTAB macroinstruction, USS
table) 1081
DATEFORM start option 827
DATEFRM operand (USSTAB macroinstruction, USS
table) 1081
DATMODE operand
  PU definition statement
    model major node 377
    switched major node 587
  VTAM restrictions on NCP definition statements 472
DBCS (double-byte character set) support (MODEENT
macroinstruction) 1037
DCODE operand (MODEENT macroinstruction) 1033
DDRAINL operand (APPL definition statement) 50
DEFAULT operand
  MAPSTO definition statement 734, 738
  USSPARM macroinstruction 1082
default SSCP lists, defining 704
default USS table
  operation-level 1133
  session-level 1132
defaults, using 13
defining major nodes
  adjacent control point 19
  application program 32
  CDRM 145
  CDRSC 103
  channel-attachment 84
  external communication adapter 172
  local non-SNA 274
  local SNA 288
  LU group 340
  model 357
  NCP 422
  switched 551
Definition Source Validation (syntax checker) 8
definition statements, coding conventions 8
DELAY operand
  PCCU definition statement (NCP major node) 472
  PORT definition statement (XCA major node) 217
  PU definition statement
    channel-to-channel 93
    channel-to-NCP 93
    local SNA major node 307
DELETE statement (dynamic reconfiguration) 696
DELETE operand
  NCP dynamic path update 683
  VTAM dynamic path update 690
  VTAM PATH definition statement 679
DESC operand (USSMSG macroinstruction) 1083
description of VTAM major nodes 2
description of VTAM tables and filter 5
DESTSA operand
dynamic path update
  NCP 684
  VTAM 690
  PATH definition statement 679
device number of channel-attached NCP 92
DIAL operand
  GROUP definition statement
    external communication adapter major node 217
  VTAM restrictions on NCP definition statements 473
DIALNO operand
  PATH definition statement (switched major node) 587
DIALRTRY start option 828
directory definition file for CMIP services, coding 1008
DIRSIZE start option 829
DIRTIME start option 829
disability 1403
DISCNT operand
  PU definition statement
    local SNA major node 307
    switched major node 377, 590
  VTAM restrictions on NCP definition statements 473
DISCNTIM start option 830
disconnect (DISCNT), automatic, for switched subarea network 377, 590
DISJOINT operand
  CDRM definition statement (cross-domain resource manager major node) 159
  DLCADDR operand 217
  PATH definition statement 591
DLCI operand 474
  NCP major node 435
  switched major node 558, 560
DLOGMOD operand
  APPL definition statement 50
  CDRSC definition statement 127
  LOCAL definition statement 280
  LU definition statement
    local SNA major node 308
    LU group major node 346
    model major node 379
    switched major node 597
  VTAM restrictions on NCP definition statements 475
  DUMPSTA operand (PCCU definition statement) 477
DLRORDER start option 831
DLYFCB start option 832
DLRNAME operand
  PU definition statement 597
  DLYRSW start option 833
DMINWNL operand (APPL definition statement) 51
DMINNWR operand (APPL definition statement) 51
DNS, online information xxvii
domain
  defining multiple to VTAM 1
  defining single to VTAM 1
double-byte character set (DBCS) support (MODEENT macroinstruction) 1037
DRESPL operand (APPL definition statement) 52
DRTYPE operand (DR ADD definition statement) 700
DSESXI operand (APPL definition statement) 53
DSIRFMG start option 834
DSPLYDEF start option 835
DSPLYMAX start option 836
DSPLYWLD operand (APPL definition statement) 53
DSPLYWLD start option 837
dump communication controller
  after failure 457
  automatic 457
  communication controller dump file 475
  CSP dump file 464
  MOS dump file 510
  dump station name (DUMPSTA) 477
  DUMPDS operand (PCCU definition statement) 475
  DUMPLD operand (PCCU definition statement) 476
  DUMPSTA operand (PCCU definition statement) 477
  DUPDEFS start option 838
  DWACT operand (PU definition statement, switched major node) 598
DWINOP operand 379
  PU definition statement
  switched major node 598
  DYNADCP operand
  PU definition statement
    local SNA major node 307
    model major node 379
    switched major node 599
  DYNADCP start option 839
  dynamic path update
    description 682
    NCP 682
    VTAM 688
dynamic reconfiguration
  active and inactive requirements 694
  ADD definition statement 695
  adding a PU or LU to a line 695
  adding an LU to a PU 695
  deleting a PU and its LUs 695
  deleting an LU 695
  MOVE definition statement 697
  moving a PU 696
  dynamic reconfiguration and dynamic change 694
  DYNASSCP start option 840
  DYNADCP start option 841
  DYNADCP start option 842
  DYNLU operand 223
  ADJCP definition statement, ADJCP major node 23
  PU definition statement
    local SNA major node 310
    switched major node 601
  VTAM restrictions on NCP definition statements 479
  DYNLU start option 842
  DYMODTB start option 843
  DYNLU operand
  GROUP definition statement
    NCP major node 479
DYNPUFX operand
GROUP definition statement
NCP major node 480
DYNPUFX start option 844
DYNPUFX start option 844
DYNPUFX option
external communication adapter major node 224
ncp major node 481
start options 844
DYNPUFX start option 844
DYNWIND operand
NCP major node 435
switched major node 558, 560
E
EAS operand
APPL definition statement 54
CDRSC definition statement 127
LU definition statement
local SNA major node 311
ECLTYPE operand
VTAM restrictions on NCP definition statements 482
EEPRANR start option 845
EEPORTCK start option 845
EEVERIFY start option 846
EEVEREY start option 846
ELEMENT operand (CDRM major node)
CDRM definition statement 161
GWPATH definition statement 161
EN (end node) 859
ENBCAST operand 745
ENCRCR operand 127
APPL definition statement 55, 128
LU definition statement
local SNA major node 311
LU group major node 346
model major node 385
NCP major node 482
switched major node 603
MODEENT macroinstruction 1034
ENCRCR PREF start option 847
ENCRYPTN operand
cross domain resource major node 128
local SNA major node 311
LU group major node 346
model major node 385
ncp major node 482
switched major node 603
ENCRYPTN start option 847
END macroinstruction 1062
end node (EN) 859
ENINTAB macroinstruction 1018
ENHADDR start option 849
Enterprise Extender
external communication adapter major node 176, 196, 237, 241, 252, 256
switched major node 556, 571, 589, 607, 639
ERN operand
NCP dynamic path update 685
PATH definition statement 680
VTAM dynamic path update 691
ESC option (VTAM internal trace) 961
ESRFSMSG start option 850
extended recovery facility (XRF) (continued)
ANS operand (continued)
NCP major node 453
switched major node 576
HAVAIL operand (application program major node) 58
external communication adapter major node, defining 172
EXTRN definition statement 1096
F
F parameter, buffer pool start option 806
FASTPASS operand
APPL definition statement 57
LU definition statement
local SNA major node 313
LU group major node 346
model major node 387
switched major node 605
VTAM restrictions on NCP definition statements 484
FEATUR2 operand
LOCAL definition statement 280
LU definition statement
model major node 388
switched major node 606
VTAM restrictions on NCP definition statements 485
filing VTAM network definition statements 16
filter and tables, description of VTAM 5
FLDEND macroinstruction 1047
FLDENT macroinstruction 1047
FLDTAB macroinstruction 1047
FLDTAB start option 852
FMFPROF operand (MODEENT macroinstruction) 1035
FORMAT operand
USCMD macroinstruction 1084
USSTAB macroinstruction 1084
format, definition statements 8
format, macroinstructions 10
formats, coding for macroinstructions 10
frame relay 486
FRELAY operand (NCP major node) 486
FROM operand
DELETE (DR) definition statement 700
MOVE (DR) definition statement 700
FRSESET definition statement (NCP major node) 426
FSRFSMSG start option 853
G
gateway paths 148
general trace facility (GTF) 958
GID operand (switched PATH definition statement) 607
GP3174 operand
description 487
VTAM restrictions on NCP definition statements 487
GPOOLL operand
VTAM restrictions on NCP definition statements 487
GREXIT operand 750
GROUP definition statement
channel-attachment major node
cross-domain resource major node 115
element major node 113
element major node
extended recovery facility (XRF) (continued)
element major node
external communication adapter major node
external communication adapter peripheral node 175
external communication adapter subarea node 173
GROUP definition statement (continued)
  NCP major node  427
  GRPNM operand
    PATH definition statement
      switched major node  607
GTF (general trace facility)  958
GVBKDLY start option  854
GWCTL operand (PCCU definition statement)  488
GWN operand (CDRM major node)
  GWPATH definition statement  162
GWNAU definition statement (NCP) restrictions  427
GWPATH definition statement (CDRM major node)  148
GWSELECT operand (CDRM definition statement)  162
GWSSCP start option  854
H
  HARDCOPY operand (FLDENT macroinstruction)  1050
  HAVAIL operand (APPL definition statement)  58
  HAVAIL operand (LOGOFF command)  1110
  HOST definition statement
    considerations when defining channel-attached major node  88
    VTAM restrictions in NCP  424
HOSTNAME  607
HOSTNAME start option  856
HOSTPU start option  858
HOSTSA start option  858
HOSTTOTRM start option  860
HPR operand
  PU definition statement  228, 229
    local SNA major node  162, 313
    NCP major node  489
  switched major node  608
HPR option (VTAM internal trace)  962
HPR start option  860
HPRAR8B start option  862
HPRECLKRT start option  862
HPREER8AR operand  229, 389, 609
HPREELIV operand  229
HPRTIMMSG start option  863
HPRPUSDLY operand  230, 389, 609
HPRPSDLYL start option  865
HPRPSSML start option  866
HPRPSMST start option  871
HPRQQLIM operand
  NCP major node  435
  switched major node  558, 560
HPRSELSML start option  872
HPRSTALL start option  874
HSRTSIZE start option  875

I

IBM Software Support Center, contacting xviii
IBMTEST command  1113
ICN (interchange network node)  859
IDBLK operand (PU definition statement)
  switched major node  611
IDIUM operand (PU definition statement)
  switched major node  612
INBFRR operand (HOST definition statement)  490
INCLUD0E  314, 391, 490, 613
INCLUDSL operand  613
independent LU cross-domain resource, defining  104
independent LUs, registering  104
Information APARs xxiv
INITDB start option  876
INITEST operand (PCCU definition statement)  490
INPORT operand
  VTAM restrictions on NCP definition statements  492
INPCODE start option  876
INOPDUMP start option  877
INTAB macroinstruction  1017
interchange network node (ICN)  859
Internet, finding z/OS information online xxvi
interpret table
  described  1017
  installing and changing  1020
  logon message, defining  1017
  logon-interpret routines, coding  1018
INTERFACgef operand
  NCP major node  435
  switched major node  558, 560
  JOINT start option  878
  IOMSGLIM start option  879
  IOPURGE start option  879
IPADDR operand  231
IPADDR start option
  start option definition  881
  switched major node  614
  syntax diagram  756
IPPORT  232
IPQlim operand
  NCP major node  435
  switched major node  558, 560
IPRESOLV  233
IPRESOLV operand  614
IPTOS  234
IQDCHPID start option  883
IQDIOSTG start option  885
IRETRY operand
  PU definition statement
    model major node  391
    switched major node  615
IRNSTRGE start option  886
ISTADJCP
  dynamic adjacent control point major node  19
ISTAPNCP
  generic representation for an APPN CDRM  708
ISTAPNPUPU
  generic representation for an APPN adjacent link station  120
ISTATUS operand
  CDRM definition statement (CDRM major node)  163
  CDRSC definition statement (CDRSC major node)  129
GROUP definition statement
  CDRSC major node  129
    channel-to-channel  94
    channel-to-NCP  94
    external communication adapter major node  234
    multipath channel (MPC)  94
LINE definition statement
  channel-to-channel  94
  channel-to-NCP  87
  external communication adapter major node  234
  multipath channel (MPC)  94
LOCAL definition statement  281
LU definition statement
  local SNA major node  314
  switched major node  616
ISTATUS operand (continued)  
PU definition statement  
channel-attached NCP  94  
channel-to-channel 94  
external communication adapter major node  234  
local SNA major node  314  
multipath channel (MPC)  94  
switched major node  616  
VTAM restrictions on NCP definition statements  494  
ISTCOSDF start option  887  
ISTCOSDF, default COS entry in default logon mode table  986  
ISTGENERIC, IBM-supplied generic resources structure name  939  
ISTINCDT, session-level USS table  
discussed 1070  
listed 1132  
ISTINCLM, default logon mode table  
described 1026  
listed 1117  
ISTINCNO, operation-level USS table contents  1070  
listed 1133  
ISTLSXCF, dynamic local SNA major node  401  
ISTMGC10, default session awareness data filter  1061  
ISTPDILU, file for independent LUs  108  
ISTSDCOS, assembled COS table  989  
ISTVTCos, the SSCP COS table entry  986  

K  
kanji character set support  1037  
katakana character set support  1037  
KCLASS macroinstruction  1062  
KCLASS operand (MAPSESS macroinstruction)  1063  
KEEPACT operand  235  
KEEPMEM macroinstruction  1062  
keyboard  1403  

L  
LADDR operand  
NCP major node  435  
switched major node  558, 560  
LANG operand  
IBMTEST command  1114  
LOGOFF command  1110  
LOGON command  1105  
MODEENT macroinstruction  1036  
LANGTAB operand  
IBMTEST command  1114  
LOGOFF command  1111  
LOGON command  1105  
LANTEST operand  
switched major node  617  
XCA major node  236  
LASTRW operand  663  
LBUILD definition statement  275  
LCS option (VTAM internal trace)  962  
license, patent, and copyright information  1407  
LIMINTCP start option  887  
LIMOSINT operand (APPL definition statement)  58  
LIMRES operand  
LINE definition statement  
external communication adapter peripheral node  237  
LIMRES operand (continued)  
PU definition statement  
local SNA major node  316  
model major node  394  
switched major node  620  
VTAM restrictions on NCP definition statements  494  
LINE definition statement  
channel-attachment major node  
channel-to-channel adapter  87  
channel-to-NCP  87  
external communication adapter major node  
external communication adapter peripheral node  175  
external communication adapter subarea node  173  
NCP major node  427  
LINEAUT operand (NCP) definition statement, VTAM restrictions on  495  
LINENM operand (PATH definition statement)  
switched major node  619  
LINEROW definition statement (APPN Class of Service)  991  
lk station for dumps  477  
LIST operand (message-flooding prevention table)  1050  
LIST start option  888  
LISTBKUP start option  888  
LIVTIME  237  
LLERP operand  
PU definition statement  
NCP major node  495  
switched major node  619  
LMDEDNT operand (APPL definition statement)  59  
LMODADR operand  
VTAM restrictions on NCP definition statements  496  
LNCTL operand  
GROUP definition statement  
channel-attached NCP  95  
channel-to-channel attachment  95  
multipath channel (MPC)  94  
TRLE definition statement  664  
VTAM restrictions on NCP definition statements  497  
LOADERFOM operand (PCCU definition statement)  498  
LOADLIB (BUILD definition statement)  499  
LOADSTA operand (PCCU definition statement)  499  
LOCADDR operand  
LU definition statement  
local SNA major node  317  
model major node  394  
switched major node  620  
VTAM restrictions on NCP definition statements  501  
LOCAL definition statement  275  
local non-SNA major node, defining  274  
local SNA major node, defining  288  
LOCAPPL operand  750  
LOCLU operand  751  
LOGAPPL operand  
CDRSC definition statement  130  
LOCAL definition statement  282  
LU definition statement  
local SNA major node  318  
LU group major node  349  
model major node  394  
switched major node  620  
VTAM restrictions on NCP definition statements  502  
LOGCHAR macroinstruction  1017  
LOGMODE operand  
LOGON command  1106  
MODEENT macroinstruction  1037  
LOGOFF command  1108  
LOGON command  1103  

1428  z/OS V2R1.0 Communications Server: SNA Resource Definition Reference
logon manager, defining to VTAM 36
logon mode table
  defined 1026
  installing 1028
MVS defaults listed 1117
  session parameters 1027
LOGTAB operand
  LOCAL definition statement 283
  LU definition statement
    local SNA major node 319
    LU group major node 341
    model major node 395
    switched major node 621
VTAM restrictions on NCP definition statements 503
LOSTREAM exit routine 1113
LOSTREAM operand (APPL definition statement) 59
LPDA operand
VTAM restrictions on NCP definition statements 503
LU 6.2 communication, VTAM internal trace of' 960
LU definition statement
  local SNA major node 290
  LU group major node 341
  model major node 360
  switched major node 557, 560
LU definition statement (NCP) operands used by VTAM 428
LU group major node, defining 340
LUALIAS operand
  CDRSC definition statement (cross-domain resource major node) 131
LUALIAS values after a merge 108
LUAPLEX operand 60
LUDR operand
  local SNA major node 319
  LU group major node 350
  NCP major node 503
LUDRPOOL (NCP) definition statement 425
LUGROUP definition statement 341
LUGROUP operand
  PU definition statement
    local SNA major node 320
    model major node 395
    switched major node 622
VTAM restrictions on NCP definition statements 505
LUNAME argument of TEXT operand 1096
LUSEED operand
  PU definition statement
    local SNA major node 320
    model major node 396
    switched major node 622
VTAM restrictions on NCP definition statements 505

M
MAC operand 60, 1038
MACLTH operand 61, 1039
macroinstructions, coding formats 10
MACTYPE operand 62, 1040
mainframe education xxiv
MAINTLVL start option 890
major nodes, defining
  adjacent control point 19
  application program 32
  CDRM 145
  CDRSC 103
  channel-attachment 84
  external communication adapter 172
major nodes, defining (continued)
  local non-SNA 274
  local SNA 288
  LU group 340
  model 357
  NCP 422
  switched 551
major nodes, description of VTAM 2
MAPSESS macroinstruction 1062
MAPSTO definition statement 741
MAXBRU operand
  HOST definition statement 506
  LINE definition statement
    channel-to-channel 95
    channel-to-NCP 95
  multipath channel (MPC) 95
PU definition statement
  local SNA major node 320
  TRLE definition statement 665
MAXDATA operand
  LOCAL SNA definition statement 321
  PCCU definition statement 506
  PU definition statement
    channel-attached NCP 96
    model major node 396
    switched major node 622
MAVLSTRT start option 892
MAVLURE start option 893
MAXOUT operand
  PU definition statement
    model major node 398
    switched major node 624
VTAM restrictions on NCP definition statements 506
MAXPVT operand (APPL definition statement) 63
MAXREADS operand 665
MAXSESS operand
  CDRSC definition statement 132
  LU definition statement
    NCP major node 507
    switched major node 624
  PU definition statement
    model major node 398
MAXSLOW operand 240
MAXSSCPU start option 894
MAXSUSBA operand
  BUILD definition statement 507
  NETWORK definition statement 507
  MAXSUSBA start option 894
MCSFLAG operand (USMSG macroinstruction) 1086
MDH (migration data host) 859
MDL ENT macroinstruction 1054
MDL ENT operand
  APPL definition statement 64
  CDRSC definition statement 133
  LOCAL definition statement 283
  LU definition statement
    local SNA major node 323
    LU group major node 351
    model major node 399
    switched major node 625
VTAM restrictions on NCP definition statements 508
MDLPLU macroinstruction 1055
MDLTAB macroinstruction 1054
MDLTAB operand
  APPL definition statement 64
  CDRSC definition statement 133
  LOCAL definition statement 284
NETID operand (continued)
  PU definition statement (continued)
    model major node 403
    switched major node 631
  VPATH definition statement 692
  VTAM restrictions on NCP definition statements 513
NETID start option 902
NETSRVR definition statement, network node server list 744
NETSRVR operand
  CDRSC definition statement (cross-domain resource major
    node) 135
Network Control Program (NCP) generation
  definition statements used by VTAM 422
  operands used by VTAM 429
  major node, defining 422
  PCCU definition statement 423
  resources controlled by 427
  VTAM restrictions on
    BUILD definition statement 424
    GWNAU definition statement 427
    HOST definition statement 424
    LUDRPOOL definition statement 425
    NETWORK definition statement 426
    PUDRPOOL definition statement 425
  SYSCNTRL definition statement 425
NETWORK definition statement
  adjacent cluster routing table 715
  adjacent SSCP table 704
  border node COS mapping table 740
  CDRM major node 147
  CDRSC major node 115
  NCP major node 426
  network node (NN) 859
  network node server list (NNSL)
    defining 5
    described 743
  NEWNAME operand (BUILD definition statement) 514
  NEWPATH operand (PCCU definition statement) 515
NEXTCP definition statement
  adjacent cluster routing table 715
NMVTLOG start option 902
NN (network node) 859
NN operand 247
  ADJCP definition statement (adjacent control point major
    node) 28, 29
  CDRM definition statement 165
  PU definition statement
    local SNA major node 326
    model major node 403
    NCP major node 515
    switched major node 632
NNSL (network node server list)
  defining 5
  described 743
NNSPREF start option
  start option definition 903
  syntax diagram 756
node names to avoid 15
node types and their functions 859
NODELST start option 904
NODEROW definition statement (APPN Class of Service) 991
NODETYPE start option 904
NONE option, VTAM internal trace 962
nonnative network attachment (XNETALS start option) 978
NOPROMPT or PROMPT start option 913
NOTNSTAT or TNSTAT start option 946
NOTRACE operand, VTAM internal trace 958
NQNMODE operand
  CDRSC definition statement (cross-domain resource major
    node) 136
  NQNMODE values after a merge 109
  NQNMODE start option 905
NRM option, VTAM internal trace 962
NSEXIT exit routine 1113
NSRTSIZE start option 905
NUMBER operand (APPN Class of Service)
  LINEROW definition statement 997
  NODEROW definition statement 998
  NUMTREES start option 906
NOTRACE operand, VTAM internal trace 958
NQNMODE operand
  CDRSC definition statement (cross-domain resource major
    node) 136
  NQNMODE values after a merge 109
  NQNMODE start option 905
NRM option, VTAM internal trace 962
NSEXIT exit routine 1113
NSRTSIZE start option 905
NUMBER operand (APPN Class of Service)
  LINEROW definition statement 997
  NODEROW definition statement 998
  NUMTREES start option 906

O
operators field
  VTAM definition statements 9
  VTAM macroinstructions 10
operation field of VTAM macroinstructions 10
operation-level USS table 1069
operator commands
  terminal 1102
  VTAM 1101
OPERCNOS operand (APPL definition statement) 67
OPT=BLKSUP operand (USSMSG macroinstruction) 1090
OPTION operand, VTAM internal trace
  ALL 959
  API (application program interface) 960
  APPC (LU 6.2 communication) 960
  CIO (channel I/O) 960
  CMIP (Common Management Information Protocol) 961
  default 960
  ESC (execution sequence control) 961
  HPR (high performance routing) 962
  LCS (LAN Channel Station) 962
  LOCK (locking) 962
  MSG (messages) 962
  NONE (no traces) 962
  NRM (network resource management) 962
  PIU (path information unit) 962
  PSS (process scheduling services) 962
  SMS (storage management services) 963
  SSCP (system services control point) 963
  VCNS (VTAM Common Network Services) 963
order of USS table use 1071
ORDER operand
  VBUILD definition statement (network node server
    list) 747
OSA-Express
  transport resource list major node 658, 660, 662, 666, 667,
    670, 671, 674
OSIEVENT start option 906
OSIMGMT start option 907
OSITOPO start option 908
OSRTSIZE start option 909
outbound calls, allocating 579
overriding sift-down value 14
overview of VTAM major nodes 2
OWNER operand
  NCP definition statements, VTAM restrictions on 517
  PCCU definition statement 518
Index 1431
reconfiguration, dynamic (continued)
  adding an LU to a PU 695
  DELETE definition statement 696
  deleting a PU and its LUs 695
  deleting an LU 695
  MOVE definition statement 697
  moving a PU 696
RECOVERY operand
  CDRM definition statement (CDRM major node) 167
  REDDELAY operand 407, 636
  REDIAL operand 408
  REDIAL operand (PATH definition statement)
  switched major node 637
REGISTER operand
  APPL definition statement (application program major
  node) 71
  CDRSC definition statement (cross-domain resource major
  node) 138
LU definition statement
  local SNA major node 329
  LU group major node 353
  model major node 408
  switched major node 637
PU definition statement
  model major node 408
  NCP major node 523
registering application programs 32
registering independent LUs 104
registering resources 32
REMOVE operand 1023
REP operand
  USSCMD macroinstruction 1092
  USSPARM macroinstruction 1092
REPLYTO operand
  GROUP definition statement
  channel-to-channel 100
  multipath channel (MPC) 100
  TRLE definition statement 672
RESETLU operand 330, 355, 410, 525, 639
resource registration 32
RESSCB operand
  CDRSC definition statement 140
  LU definition statement
  switched major node 639
  NCP definition statements, VTAM restrictions on 525
  restrictions on VTAM names 14
RESUSAGE start option 919
RETRIES operand
  PU definition statement
  NCP major node 525
RFC (request for comments) 1377
accessing online xxvi
RNAME operand
  PCCU definition statement 527
ROUTCDE operand (USMSGS macroinstruction) 1093
route addition resistance (RAR) 1001
ROUTEDIR operand 252
ROUTERES operand (APPN Class of Service) 1001
ROUTERES start option 920
routes, defining 677
RUSIZES operand (MODEENT macroinstruction) 1042
S
SACONNS start option 921
SALIMIT operand
  BUILD definition statement 528
SALIMIT operand (continued)
  NETWORK definition statement 528
SAMAP 1057
SAPADDR 639
SAPADDR operand
  PU definition statement
  external communication adapter major node 253
SAVEMOD operand (PCCU definition statement) 528
SAVERSCV
  start option definition 922
  syntax diagram 756
SAW (session awareness) data filter 1061
SAW operand (KCLASS macroinstruction) 1065
SAWMAXDS start option 923
SAWMPQPK start option 924
SBCS (single-byte character set) support 1037
SDLCDMRDS start option 924
SEC operand (MAPSESS macroinstruction) 1065
SECACPT operand
  APPL definition statement 73
  SECLVL operand
  APPL definition statement 73
  SECLVLCP start option 925
SECNET operand
  MAPSESS macroinstruction 1066
  NCP definition statements, VTAM restrictions on 529
  PU definition statement
  local SNA major node 331
  model major node 410
  switched major node 641
SECPROT operand (MODEENT macroinstruction) 1043
SECURITY operand
  CDRM definition statement 167
  GROUP definition statement
  Enterprise Extender XCA major node 254
  LINEROW definition statement
  APPN Class of Service 1002
  PORT definition statement
  XCA major node 254
  PU definition statement
  local SNA major node 331
  model major node 411
  NCP major node 529
  switched major node 641
  TGIP definition statement
  transmission group profile 728
  security, session
  operands affecting
    ENCR 55, 128, 311, 346, 385, 482, 603, 1034
    SECLVL 73
  VERIFY 82, 654
  start options affecting 780
SECOND operand
  VTAM restrictions on NCP definition statements  531
  SEQNCCE operand (LOGCHAR macroinstruction) 1024
  serial optical channel (SOC) line 523
  session awareness (SAW) data filter 1061
  session parameters 1026
  session security
  operands affecting
    ENCR 55, 128, 311, 346, 385, 482, 603, 1034
    SECLVL 73
    VERIFY 82, 654
  start options affecting 780
  session-level USS table 1069
  SESSSLIM operand 75
SHM operand
PATH definition statement
switched major node 642
VTAM restrictions on NCP definition statements 532
SHMTIM operand
PATH definition statement (switched major node) 644
shortcut keys 1403
sift-down effect 14
sifting 14
single-byte character set (SBCS) support 1037
SIRFMSG operand
appl definition statement 75
cross-domain resource statement 140
SIRFMSG start option 926
SIZE operand, VTAM internal trace 964
SLOWPT buffer pool start option operand 806
SLOWVAL start option 927
SLUALMSG start option 928
SLUNIT operand, NETSRVR definition statement 747
SMAUTH start option 929
SMS option, VTAM internal trace 963
SNA protocol specifications 1401
SNAMGMT start option 929
SNAPREQ start option 930
SNETMASK operand
NCP major node 435
switched major node 558, 560
SNVC operand
NEXTCP definition statement 719
SOC (serial optical channel) line 523
softcopy information xxiv
SONLIM start option 931
SONSCIP operand (APPL definition statement) 76
SORDER start option 932
SPAN operand
APPL definition statement 77
CDRM definition statement 168
CDRSC definition statement 141
GROUP definition statement
channel-attached NCP 101
cannel-to-channel 102
external communication adapter peripheral node 255
eexternal communication adapter subarea node 255
multipath channel (MPC) 101
LINE definition statement
channel-attached NCP 101
cannel-to-channel 101
eexternal communication adapter peripheral node 255
eexternal communication adapter subarea node 255
multipath channel (MPC) 101
LOCAL definition statement 286
LU definition statement
local SNA major node 331
model major node 412
switched major node 644
PU definition statement
channel-attached NCP 101
cannel-to-channel 101
eexternal communication adapter peripheral node 255
eexternal communication adapter subarea node 255
local SNA major node 332
model major node 411
switched major node 644
SRBEXIT operand
APPL definition statement 77
SRCHRED start option 934
SRCOUNT operand (CDRSC definition statement) 142
SRCOUNT start option 934
SRCVPAC operand (MODEENT macroinstruction) 1044
SRQRETTRY 255
SRQTME 256
SRRTMR start option 934
SRTMR operand (CDRSC definition statement) 142
SRTMR start option 934
SSCP option, VTAM internal trace 963
SSCPDYN start option 935
SSCPF operand
APPL definition statement 78
LOCAL definition statement 287
LU definition statement
local SNA major node 332
model major node 412
switched major node 645
VTAM restrictions on NCP definition statements 532
SSCPID start option 935
SSCPNAME start option 936
SSCPORD operand 711
SSCPORD start option 937
SSDTMOUT start option 937
SSEARCH start option 938
SSNDPAC operand (MODEENT macroinstruction) 1044
start options
AFFDELAY 794
ALREQ 794
AP64R 795
APPINCOS 796
ASIRFMSG 797
ASYDE 798
AUTHLEN 798
AUTORTY 799
AUTOI 799
BN 800
BNDRYN 800
BNORD 801
BSCMTRS 802
BSCTMOUT 803
buffer pools 804
CACHETI 811
CDRDYN 811
CDSCTI 812
CDSERVR 813
cDSREFER 814
CIDXSIZE 814
CMPMIPS 815
CNMTAB 816
cNNRTMSG 816
coding format 755
cOLD or WARM 817
CONFIG 817
cONNTYPE 818
cPDRSC 819
cPCP 820
cSA24 822
cSALIMIT 820
cSDUMP 823
cDATEFORM 827
cDIALRTY 828
dIRSIZE 829
dIRTIME 829
dISCNTIM 830
dLRORDER 831
dLRRCB 832
dLURSAW 833
| Index | 1435 |

start options (continued)

- DSIRFMSG 834
- DSPLYDEF 835
- DSPLYMAX 836
- DSPLYWLD 837
- DUPDFS 838
- DYNADCJP 839
- DYNASSCP 840
- DYNDLGMD 841
- DYNHPFPFX 842
- DYNLU 842
- DYNMODTB 843
- DYNPUFX 844
- DYNVNPFX 844
- EEHPRANR 845
- EEPORTCK 845
- ENCRYPTN 847
- ENHADDR 849
- ESIRFMSG 850
- EXPFLTRM 851
- FLDTAB 852
- FSIRFMSG 853
- GVBDL 854
- GWSSCP 854
- HNTSIZE 855
- HOSTNAME 856
- HOSTPU 858
- HOSTSA 858
- HSTIOTRM 860
- HPR 860
- HPRARB 862
- HPRCLKRT 862
- HPRIMSG 863
- HPRPSDL 865
- HPRPSMSG 866
- HPRPSST 871
- HPRSLESML 872
- HPRSTALL 874
- HSRTSIZ 875
- INITDB 876
- INOPCODE 876
- INOPDUMP 877
- JOIN 878
- IOMSC 879
- IOPERG 879
- IOPINFO 882
- IQDCHPID 883
- IQDIOSTG 885
- IRNSTRG 886
- ISTCOSDF 887
- LIMINTCP 887
- LIST 888
- LISTbkup 888
- LSIRFMSG 889
- MAINTLVL 890
- MAXLOCAT 892
- MAXLURU 893
- MAXSSCP 894
- MAXSUBA 894
- MIHTMOUT 895
- MSGLEVEL 896
- MSGMOD 898
- MXSAWBUF 899
- MXSCPRU 900
- MXSUBNUM 900
- NACPR 900
- NCPBUFFS 901
- NETID 902
- NMVTLOG 902
- NODELST 904
- NODETYPE 904
- NOTNSTAT 946
- NUMTREES 906
- OSIEVENT 906
- OSIMGMT 907
- OSITOPO 908
- OSRTSIZE 909
- PDTRCBUF 910
- PIUMAXDS 910
- PLUALMSG 911
- PMTUD 911
- PPOLOG 912
- PROMPT or NOPROMPT 913
- PSRETRY 914
- PSTRACE 916
- QDIOSTG 918
- RESUSAGE 919
- ROUTERES 920
- RPSFMSG 920
- SACONNS 921
- SAWMADE 923
- SAWMQPK 924
- SDLNCRDS 924
- SECLVCL 925
- SIRFMSG 926
- SLOWVAL 927
- SLUALMSG 928
- SRE.Auth 929
- SNAPFCNT 929
- SNAPREQ 930
- SONLIM 931
- SORDER 932
- SRCORE 934
- SRCPRED 934
- SSCPDE 935
- SSCPID 935
- SSCPNAME 936
- SSCPORE 937
- SSDTMOUT 937
- SSEARCH 938
- STRG 938, 939
- STRMPS 939
- SUPP 940
- SVNSORDER 942
- TNSTAT 946
- TOP frmin 946
- TRAC 947, 955
- TRNSTAT 965
- VOSDEACT 973
- VRTG 974
- VRTGCPCP 974
- VTAMEAS 975
- XCFINIT 976
takover restrictions
ANS operand (PU definition statement in switched major node) 369, 576
ANS operand on NCP definition statements 453
dynamic restrictions 696
TCP/IP
online information xxvi
protocol specifications 1377
TCPNAME start option
start option definition 943
syntax diagram 756
Technotes xxiv
TERM operand
LOCAL definition statement 287
LU definition statement
model major node 413
switched major node 646
VTAM restrictions on NCP definition statements 536
TERMINAL definition statement
VTAM restrictions on NCP definition statements 428
terminal operator commands 1102
TEXT operand (USSMSG macroinstruction) 1096
TGCONF operand 646
TGN operand
PCCU definition statement 537
PU definition statement
channel-to-channel 102
channel-to-NCP 102
external communication adapter subarea node 259
local SNA major node, for APPN connections 332
multipath channel (MPC) 102
NCP major node 536
switched major node 647, 648
TGP (APPN transmission group profile)
defining 720
described 5
IBM-supplied transmission group profiles 1356
IBMTGPS (member list containing the IBM-supplied TGP) 720
TGP definition statement 720
TGP operand
CDRM definition statement 169
GROUP definition statement
Enterprise Extender XCA major node 259
PORT definition statement
XCA major node 259
PU definition statement
local SNA major node 333
model major node 414
NCP major node 537
switched major node 648
TIME operand (FLDENT macroinstruction) 1053
TIMEDLM operand (USSTAB macroinstruction, USS table) 1087
TIMER operand (external communication adapter subarea node) 261
TNSTAT start option 946
TO operand
ADD definition statement (dynamic reconfiguration) 702
MOVE definition statement (dynamic reconfiguration) 702
TOPO operand
PU definition statement
local SNA major node 261, 334
model major node 414
NCP major node 538
switched major node 649
TOPOTIME start option 946
trace

start options
  description 947
  NOTRACE (cancel trace) 952
  TYPE=BUF (VTAM buffer trace) 954
  TYPE=IO (VTAM I/O trace) 954
  TYPE=LINE (NCP line trace) 954
  TYPE=IOR (input/output trace) 954
  TYPE=SIT (scanner interface trace) 954
  TYPE=SMS (VTAM buffer request trace) 954
  TYPE=VTAM (VTAM internal trace) 955

VTAM internal trace
  ALL option 959
  API option (applications program interface) 960
  APPC option (LU 6.2 communication) 960
  CIO option (channel I/O) 960
  CMIP option (Common Management Information Protocol) 961
  default 960
  ESC option (execution sequence control) 961
  HPR option (high performance routing) 962
  LCS option (LAN Channel Station) 962
  LOCK option (locking) 962
  MODE operand 958
  MSG option (messages) 962
  NONE option (no traces) 962
  NOTRACE operand 958
  NRM option (network resource management) 962
  OPTION operand 958
  PIU option (path information units) 962
  PSS option (process scheduling services) 962
  SIZE operand 964
  SMS option (storage management services) 963
  SSCP option (system services control point) 963
  VCNS option (VTAM Common Network Services) 963

TRACE start option 947

trademark information 1415

TRANSLAT start option 965

translation table 1070

transmission group profile (APPN)
  defining 720
  described 5
  IBM-supplied transmission group profiles 1356
  IBMMTGPS (member list containing the IBM-supplied TGP) 720

transmission priority indicator number 986

transport resource list major node defining 658

TRGL definition statement 660

TRGL operand 414

PU definition statement 334

TSO/VTAM, defining application programs 36

TSPROF operand (MODEENT macroinstruction) 1045

type operand
  LOGOFF command 1112
  MODEENT macroinstruction 1045

U

unformatted system services (USS) commands
  terminal operator 1102
  VTAM operator 1101

unformatted system services (USS) messages
coding 1074
redefining 1098
rules for creating 1099

UNRCHTLM operand
  GROUP definition statement
    Enterprise Extender XCA major node 262
  PORT definition statement
    Enterprise Extender XCA major node 262
  UNRCHTLM start option 966

UPARM1 operand
  CDRM definition statement 170
  GROUP definition statement
    Enterprise Extender XCA major node 263
  LINEROW definition statement
    APPN Class of Service 1002
    PORT definition statement
      XCA major node 263
    PU definition statement
      local SNA major node 335
      model major node 416
      NCP major node 538
      switched major node 650
    TGP definition statement
      transmission group profile 729

UPARM2 operand
  CDRM definition statement 170
  GROUP definition statement
    Enterprise Extender XCA major node 264
  LINEROW definition statement
    APPN Class of Service 1003
    PORT definition statement
      XCA major node 264
    PU definition statement
      local SNA major node 336
      model major node 417
      NCP major node 539
      switched major node 650
    TGP definition statement
      transmission group profile 730

UPARM3 operand
  CDRM definition statement 170
  GROUP definition statement
    Enterprise Extender XCA major node 265
  LINEROW definition statement
    APPN Class of Service 1003
    PORT definition statement
      XCA major node 265
    PU definition statement
      local SNA major node 336
      model major node 417
      NCP major node 539
      switched major node 651
    TGP definition statement
      transmission group profile 730

UPDDELAY start option 967

USE operand
  PATH definition statement
    switched major node 651
  VTAM restrictions on VTAM definition statements 540

USERVER
  LOGCHAR macroinstruction 1017
  LOGOFF command 1109
  LOGON command 1104

USS (unformatted system services) commands
  terminal operator 1102
  VTAM operator 1101

USS (unformatted system services) messages
coding 1074
redefining 1098
rules for creating 1099
USS (unformatted system services) table
  creating and modifying 1074
  installing 1073
  operator messages and commands 1101
USS command syntax rules 1102
USS table
  command syntax rules 1102
  default USS table
  operation-level 1133
  session-level 1132
  operation-level 1069
  order of use 1071
  session-level 1069
USS command conversion examples 1369
USSCMD macroinstruction 1074
USSEND macroinstruction 1075
USSMSG macroinstruction
  for terminal operator messages 1074
  for VTAM operator messages 1074
USSPARM macroinstruction 1074
USSTAB macroinstruction 1074
USSTAB operand
  APPL definition statement 80
  LOCAL definition statement 288
  LU definition statement
    local SNA major node 337
    LU group major node 356
    model major node 418
    switched node 652
  VTAM restrictions on NCP definition statements 541
USSTAB start option 968

V

VALUE operand (USSPARM macroinstruction) 1097
VARTEXT operand (FLDENT macroinstruction) 1053
VARYWLDE operand
  appl definition statement 80
  start option 969
VBUILD definition statement
  adjacent control point major node 20
  adjacent SSCP table 703
  application program major node 36
  channel-attachment major node 89
  cross-domain resource major node 115
  cross-domain resource manager major node 147
  dynamic reconfiguration 694
  local SNA major node 290
  LU group major node 540
  model major node 360
  network node server list 744
  switched major node 559
VCNS operand (APPL definition statement) 81
VCNS option, VTAM internal trace 963
VERALSID operand
  PU definition statement
    local SNA major node 266, 337
    model major node 418
    NCP major node 541
    switched major node 652
VERID operand
  PATH definition statement (switched major node) 653
VERIFY operand
  APPL definition statement 82
  PATH definition statement 654
VERIFYCP start option 970
VFYC operand (PCCU definition statement) 543
VFYLM operand (PCCU definition statement) 544
VFYOWNER operand (CDRSC definition statement) 144
VFYRED start option 971
VFYREDTI start option 972
VN operand
  ADJCP definition statement 31
VNGROUP operand
  GROUP definition statement
    NCP major node 544
    LINE definition statement
    NCP major node 544
    PORT definition statement
    XCA major node 267
VNNAME operand
  GROUP definition statement
    NCP major node 544
    LINE definition statement
    NCP major node 544
    PORT definition statement
    XCA major node 268
VNTYPE operand
    XCA major node 269
VOSDEACT start option 973
VPACING operand
  APPL definition statement 82
  CDRM definition statement (CDRM major node) 171
  CDRSC definition statement 145
  LU definition statement
    local SNA major node 337
    LU group major node 356
    model major node 419
    switched major node 654
  VTAM restrictions on NCP definition statements 545
VPATH definition statement (dynamic path update) 689
VR operand
  COS macroinstruction 989
  NCP dynamic path update 687
  VTAM dynamic path update 693
  VTAM PATH definition statement 680
VRPWS operand
  NCP dynamic path update 688
  VTAM dynamic path update 694
  VTAM PATH definition statement 681
VRTG operand
  CDRM definition statement 171
  VRTG start option 974
  VRTGCPCP operand
    CDRM definition statement 172
  VRTGCPCP start option 974
VTAM
  associated LU table 981
  class-of-service table 985
  definition statement coding conventions 8
  description of nodes 2
  description of tables 5
  interpret table 1017
  logon mode table 1026
  macroinstruction coding formats 10
  message-flooding prevention table 1046
  model name table 1054
  names, restrictions on 14
  nodes, overview 2
  routes, defining 677
  tables, overview 5
  USS table 1069
VTAM internal trace
  MODE operand 958
VTAM internal trace (continued)

NOTRACE operand 958
OPTION operand
  ALL 959
  API (application program interface) 960
  APPC (LU 6.2 communication) 960
  CIO (channel I/O) 960
  CMIP (Common Management Information Protocol) 961
default 960
  ESC (execution sequence control) 961
  HPR (high performance routing) 962
  LCS (LAN Channel Station) 962
  LOCK (locking) 962
  MSG (messages) 962
  NONE (no traces) 962
  NRM (network resource management) 962
  PIU (path information units) 962
  PSS (process scheduling services) 962
  SMS (storage management services) 963
  SSCP (system services control point) 963
  VCNS (VTAM Common Network Services) 963
  SIZE operand 964
VTAM major nodes, overview of 2
VTAM network definition statements, filing 16
VTAM restrictions on NCP definition statements
  BUILD definition statement 424
  GWNAU definition statement 427
  HOST definition statement 424
  LUDRPOOL definition statement 425
  NETWORK definition statement 426
  PUDRPOOL definition statement 425
  SYSCNTRL definition statement 425
VTAM-only operands on NCP definition statements,
coding 427
VTAM, online information xxvi
VTAMEAS start option 975
VTAMFRR operand (APPL definition statement) 84
VTAMTOPO operand
  Model major node 420
  NCP major node 547
  Switched major node 655
  XCA major node 270

W

WARM or COLD start option 817
WEIGHT operand (APPN Class of Service)
  LINEROW definition statement 1005
  NODEROW definition statement 1004
WLM operand 753
WRITE operand 103
  LINE definition statement (channel-attachment major node) 103
  TRLE definition statement 674
WTO/WTOR macroinstruction 1084

X

XCA major node, defining 172
XCFINIT start option 976
XID operand
  PU definition statement
    local SNA major node 272, 339
  VTAM restrictions on NCP definition statements 550
Communicating your comments to IBM

If you especially like or dislike anything about this document, you can send us comments electronically by using one of the following methods:

Internet email:
comsvrcf@us.ibm.com

World Wide Web:

If you would like a reply, be sure to include your name, address, and telephone number. Make sure to include the following information in your comment or note:

• Title and order number of this document
• Page number or topic related to your comment

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this document. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.