Before using this information and the product it supports, be sure to read the general information under "Notices" on page 265.

This edition applies to Version 1 Release 13 of z/OS (5694-A01), and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SA22-7602-19.

© Copyright IBM Corporation 1994, 2011.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
## Contents

### Figures
- vii

### Tables
- ix

### About This Information
- xi
  - Who Should Use This Information: xi
  - Where to Find More Information: xi
  - Other Referenced Documents: xii
  - Information Updates on the Web: xiii

### The z/OS Basic Skills Information Center
- xv

### How to send your comments to IBM
- xvii
  - If you have a technical problem: xvii

### Summary of Changes
- xix
  - Changes made in z/OS Version 1 Release 13: xix
  - Changes made in z/OS Version 1 Release 12: xix
  - Changes made in z/OS Version 1 Release 11: xix

### Chapter 1. Why MVS Workload Management?
- 1
  - Problems: 1
  - MVS Workload Management Solution for Today and Tomorrow: 2

### Chapter 2. What is MVS Workload Management?
- 3
  - Performance Administration: 4
  - Performance Management: 4
  - Workload Balancing: 5
  - Workload Management Concepts: 5
    - What is a Service Definition?: 5
    - Why Use Service Policies?: 6
    - Organizing Work into Workloads and Service Classes: 6
    - Why Use Resource Groups?: 8
    - Assigning Work to a Service Class: 10
    - Why Use Application Environments?: 10
    - Why Use Scheduling Environments?: 12
    - Summary of Service Definition and Service Policy Concepts: 13

### Chapter 3. Workload Management Participants
- 17
  - Workload Management Work Environments: 17
    - Subsystem Support for Goal Types and Multiple Periods: 18
    - Subsystem-Specific Performance Hints: 19
    - Workload Balancing: 19
    - Workload Management in a CPSM Environment: 20
    - Workload Management in a DB2 Distributed Data Facility (DDF) Environment: 21
    - Batch Workload Management: 21
    - Multisystem Enclave Support: 22
    - Intelligent Resource Director: 22
    - HiperDispatch Mode: 23
      - Overview: 23
      - The Concept of HiperDispatch Mode: 23
      - Setting of the HiperDispatch Mode in SYS1.PARMLIB: 24
    - I/O Storage Management: 25
      - Handling Service Class Periods with a Response Time Goal: 25
      - Handling Service Class Periods with a Velocity Goal: 26
      - Handling Other I/O Requests: 26
      - Controlling the Information Passed to the I/O Manager: 26
      - Non z/OS Partition CPU Management: 27
      - Workload Management and Workload License Charges: 27
        - Defining the Capacity of a Partition: 28
        - Defining Group Capacity: 30
      - Workload Management with Other Products: 32

### Chapter 4. Setting up a Service Definition
- 33
  - Specifying a Service Definition: 33
  - Storing Service Definitions: 34
  - Defining the Parts of a Service Definition: 34

### Chapter 5. Defining Service Policies
- 35
  - Using Policy Overrides: 36

### Chapter 6. Defining Workloads
- 39
  - Defining a Departmental Workload: 39

### Chapter 7. Defining Resource Groups
- 41
  - Calculating an LPAR Share - Example 1: 43
  - Specifying the Capacity as a Number of CPs - Example 2: 45

### Chapter 8. Defining Service Classes and Performance Goals
- 47
  - Velocity Formula: 50
  - Defining Performance Goals: 50
    - Determining System Response Time Goals: 51
    - Examples: 52
  - Defining Velocity Goals: 52
    - Adjusting Velocity Goals Based on Samples: 53
    - Included in Velocity Calculation: 53
    - Using Velocity Goals for Started Tasks: 53
    - Using Discretionary Goals: 54
    - Using Performance Periods: 55
Defining Goals Appropriate for Performance Periods ............... 55
Using Importance Levels in Performance Periods 55

Chapter 9. Defining Classification Rules ............... 57
Defining Classification Rules for Each Subsystem ..... 59
Defining Work Qualifiers ...................... 64
Defining z/OS WLM Classification Rules for Performance Policies from the Unified Resource Manager .......... 76
Defining the Order of Classification Rules .............. 78
Defining a Subsystem Service Class Default ....... 79
Organizing Work for Classification .............. 84
Using the Start Position ...................... 84
Using Masking Notation ...................... 85
Using Wildcarding Notation ...................... 85
Using Groups ...................... 86
Using the System-Supplied Service Classes ....... 87

Chapter 10. Defining Report Classes 91

Chapter 11. Defining Service Coefficients and Options ............... 93
Calculating the Amount of Service Consumed .............. 93
Service Definition Coefficients ...................... 94
Changing Your Coefficient Values .............. 94
Using the Storage (MSO) Coefficient for Calculations ...................... 95
Specifying I/O Priority Management ...................... 96
Considerations for I/O Priority Management .............. 96
Specifying Dynamic Alias Management ...................... 96
Workload Management Considerations .............. 96
HCD Considerations ...................... 97

Long-Term Storage Protection ...................... 99
Long-Term CPU Protection ...................... 100
Modifications of Transaction Response Time .......... 101
Management ...................... 102
Sample Scenarios .............. 102
Scenario 1 ...................... 104
Scenario 2 ...................... 106
Scenario 3 ...................... 107
Scenario 4 ...................... 108
Scenario 5 ...................... 108
Scenario 6 ...................... 108
Reporting ...................... 110
Option Summary ...................... 111

Chapter 13. Defining Application Environments ............... 113
Getting Started with Application Environments .............. 113
Specifying Application Environments to Workload Management ...................... 114
Selecting Server Limits for Application Environments ...................... 116
How WLM Manages Servers for an Application Environment ...................... 119
Using Application Environments .............. 120
Managing Application Environments .............. 120
  Using Operator Commands for Application Environments .............. 121
  Making Changes to the Application Environment Servers .............. 122
  Changing the Definition of an Application Environment .............. 122
  Handling Error Conditions in Application Environments .............. 122
Authorizing Application Environment Servers .............. 123
  Example for Restricting Access to Application Environment Servers .............. 123

Chapter 14. Defining Scheduling Environments .............. 125
Getting Started with Scheduling Environments .............. 125
Specifying Scheduling Environments to Workload Management .............. 126
  Managing Resource States .............. 127
  Associating Scheduling Environments with Incoming Work .............. 131
  Displaying Information about Scheduling Environments and Resource States .............. 131
  MVS Operator Commands .............. 131
  JES2/JES3 Operator Commands .............. 133
  SDSF Commands .............. 134

Chapter 15. Managing Performance with the Guest Platform Management Provider (GPMP) .............. 135
Overview of the Guest Platform Management Provider (GPMP) .............. 135
  Configuring the Guest Platform Management Provider on z/OS .............. 136
  Before You Begin .............. 136
  About This Task .............. 137
  Procedure .............. 137
  Results .............. 139
  What To Do Next .............. 139

Chapter 16. Workload Management Migration .............. 141
Creating a Service Definition for the First Time .............. 141
Migrating to a New z/OS Release With an Existing Service Definition .............. 143
  Migration Activities .............. 144
  Restricting Access to the WLM Service Definition .............. 144
  Start the Application and Enter/Edit the Service Definition .............. 145
  Calculate the Size of the WLM Couple Data Set .............. 148
  Allocate a WLM Couple Data Set .............. 148
  Make a WLM Couple Data Set Available to the Sysplex for the First Time .............. 151
  Make a Newly Formatted Couple Data Set Available to the Sysplex .............. 152
Figures

1. MVS Workload Management Overview .......................... 3
2. Workload Organized by Subsystem ............................. 8
3. Workload Organized by Department ............................ 8
4. Resource Groups .................................................. 9
5. Work Classification ................................................ 10
6. Application Environment Example .............................. 11
7. Scheduling Environment Example ............................... 13
8. Service Definition, Including Two Service Policies ....... 15
9. Sysplex View of the Management Environment ............ 18
10. Workload Management in a CICS Environment ........... 20
11. Workload Management Enforcing a Defined Capacity Limit 28
12. Formula for Calculating a Phantom Weight ................. 29
13. Example for Workload Consumption of Partitions MVS1, MVS2, and MVS3 .... 31
14. Resource Group Overview - Example ......................... 44
15. Working With A Resource Type 2 - Sample Calculation .... 45
16. Using Classification Rules to Assign Work to Service Classes .... 58
17. Sample z/OS WLM Classification Rules Panel for Mapping Unified Resource Manager performance policies .................. 77
18. Formula for Calculating Service Consumption ............... 93
19. MSO Coefficient Formula ...................................... 95
20. Specifying the “Storage Critical” Option .................. 99
21. Specifying the “CPU Critical” Option ....................... 100
22. Specifying the “Manage Region Using Goals Of:” Option .... 101
23. Scenario 1: Address Spaces .................................. 102
24. Scenarios 2, 3, 4, 5: CICS/IMS Regions .................... 104
25. Scenario 6: CICS Regions Adhering to a Work Manager/Consumer Model ......................................... 109
26. Sample Systems and Scheduling Environments .............. 130
27. One LPAR Cluster on One CPC ................................ 161
28. Four LPAR Clusters on Two CPCs ............................ 162
29. Intelligent Resource Director Example – Day Shift ...................... 165
30. Intelligent Resource Director Example – Night Shift .......... 166
31. Menu Bar on the Definition Menu ............................. 184
32. Definition Menu File Choices .................................. 184
33. Service Class Selection List ................................... 185
34. Service Class Selection List ................................... 186
35. Action field on the Subsystem Type Selection List ............ 186
36. Function Key Area ................................................ 187
37. Choose Service Definition Pop-Up ............................ 188
38. Definition Menu .................................................. 189
39. Create a Service Policy ........................................ 193
40. Service Policy Selection List ..................................... 194
41. Create a Workload ............................................... 194
42. Workload Selection List ......................................... 195
43. Create a Resource Group ..................................... 195
44. Create a Service Class ............................................ 196
45. Choose a Goal Type Pop-Up .................................... 196
46. Average Response Time Goal Pop-Up ......................... 197
47. Create a Service Class Panel .................................. 197
48. Action Codes for Goal ........................................... 197
49. Service Policy Selection List .................................... 198
50. Override Service Class Selection List ......................... 199
51. Override Attributes for a Service Class ...................... 199
52. Subsystem Type Selection List for Rules .................... 200
53. Modify Rules for the Subsystem Type ....................... 201
54. Modify Rules for the Subsystem Type, Scrolled Right to Description Fields .... 201
55. Modify Rules for the Subsystem Type, Scrolled Right to Storage Critical and Region Management Fields ........... 202
56. Action Codes for Classification Rules ....................... 203
57. Create a Group .................................................. 205
58. Modify Rules for STC Subsystem ............................ 206
59. Creating a Report Class on the Modify Rules Panel ....... 206
60. Create a Report Class Confirmation Panel .................. 207
61. Service Coefficients Panel ................................. 208
62. Application Environments ................................. 208
63. Decide What to Create Panel .................................. 209
64. Scheduling Environment Selection List Panel ............... 209
65. Create a Scheduling Environment Panel .................... 210
66. Resource Definition List Panel ................................ 210
67. Create a Scheduling Environment Panel .................... 211
68. Scheduling Environment Selection List Panel ............... 211
69. Modify a Scheduling Environment Panel .................... 212
70. Copy a Scheduling Environment Panel ....................... 212
71. Browse a Scheduling Environment Panel .................... 213
72. Delete a Scheduling Environment Panel ....................... 213
73. Scheduling Environment Selection List Panel ............... 214
74. Resource Definition List Panel ................................ 214
75. Define Resource Panel ........................................ 214
76. Resource Definition List Panel ................................ 215
77. Modify a Scheduling Environment Panel .................... 215
78. Resource Definition List Panel ................................ 216
79. Resource Definition List Panel ................................ 216
80. Modify a Scheduling Environment Panel .................... 216
81. Resource Definition List Panel ................................ 217
82. Resource Cross-Reference Of Scheduling Environments Panel ................................................... 218
83. Resource Definition List Panel ................................ 218
84. Guest Platform Management Provider Settings - Sample Panel ............. 219
85. Overwrite service definition ................................... 220
86. Allocate Couple Data Set Using CDS Values ............... 221
87. Allocate Couple Data Set Using Service Definition Values .......... 221
88. Activating a Service Policy .................................... 223
89. Browse Function from the Service Class Selection List ........... 224
90. Service Class Subsystem Xref .................................. 224
91. z/OS Management Facility - Overview Panel ............... 231
<table>
<thead>
<tr>
<th></th>
<th>Tables</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Levels of Processing</td>
</tr>
<tr>
<td>2</td>
<td>Example: Partition Definitions</td>
</tr>
<tr>
<td>3</td>
<td>Example: Partition Definitions</td>
</tr>
<tr>
<td>4</td>
<td>LPAR Configuration - Example</td>
</tr>
<tr>
<td>5</td>
<td>ELPMAX In Sysplex WLMPLEX- Example</td>
</tr>
<tr>
<td>6</td>
<td>IBM-Defined Subsystem Types</td>
</tr>
<tr>
<td>7</td>
<td>Non-IBM Subsystems</td>
</tr>
<tr>
<td>8</td>
<td>Enclave Transactions, Address Space-Oriented Transactions, and CICS/IMS Transactions</td>
</tr>
<tr>
<td>9</td>
<td>Work Qualifiers Supported by Each IBM-Defined Subsystem Type</td>
</tr>
<tr>
<td>10</td>
<td>Effects of WLMPAV Settings on Base and Alias Devices</td>
</tr>
<tr>
<td>11</td>
<td>Summary of Storage Protection, CPU Protection, and Exemption from Transaction Response Time Management Options</td>
</tr>
<tr>
<td>12</td>
<td>IBM-supplied Subsystems Using Application Environments</td>
</tr>
<tr>
<td>13</td>
<td>Application Environment Server Characteristics</td>
</tr>
<tr>
<td>14</td>
<td>z/OS Execution Environment Variables For The Guest Platform Management Provider</td>
</tr>
<tr>
<td>15</td>
<td>Functionality Levels for Service Definition</td>
</tr>
<tr>
<td>16</td>
<td>The Current WLM Couple Data Set Format Level</td>
</tr>
<tr>
<td>17</td>
<td>Values to Use in Storage Estimation Formulas</td>
</tr>
<tr>
<td>18</td>
<td>Menu Bar Options on the Definition Menu</td>
</tr>
<tr>
<td>19</td>
<td>Return Codes from IWMARIN0</td>
</tr>
<tr>
<td>20</td>
<td>IBM zEnterprise 196 (z196) processor models</td>
</tr>
<tr>
<td>21</td>
<td>IBM System z10 Enterprise Class (z10 EC) processor models</td>
</tr>
<tr>
<td>22</td>
<td>IBM System z9 Business Class (z9 BC) processor models</td>
</tr>
<tr>
<td>23</td>
<td>IBM System z9 Enterprise Class (z9 EC) processor models</td>
</tr>
<tr>
<td>24</td>
<td>zSeries 990 processor models</td>
</tr>
<tr>
<td>25</td>
<td>zSeries 900 processor models</td>
</tr>
<tr>
<td>26</td>
<td>zSeries 890 processor models</td>
</tr>
<tr>
<td>27</td>
<td>zSeries 800 processor models</td>
</tr>
<tr>
<td>28</td>
<td>S/390 9672 processor models</td>
</tr>
<tr>
<td>29</td>
<td>S/390 3000 processor models</td>
</tr>
<tr>
<td>30</td>
<td>Return Codes from IWMINSTL</td>
</tr>
</tbody>
</table>
About This Information

This document supports z/OS (5694–A01). This document contains information to help you convert to z/OS® workload management, to use workload management, and to make the most out of workload management.

Note: The z/OS workload management component is sometimes also called “Workload Manager.”

Who Should Use This Information

This document is intended for the system programmers, system analysts, and systems engineers who are responsible for developing a conversion plan for z/OS workload management, and who are responsible for implementing z/OS workload management.

Where to Find More Information

Where necessary, this document references information in other documents, 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.

<table>
<thead>
<tr>
<th>Title</th>
<th>Order Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Common Information Model User’s Guide</td>
<td>SC33-7998</td>
</tr>
<tr>
<td>z/OS Migration</td>
<td>GA22-7499</td>
</tr>
<tr>
<td>z/OS MVS Capacity Provisioning User’s Guide</td>
<td>SC33-8299</td>
</tr>
<tr>
<td>z/OS MVS Initialization and Tuning Guide</td>
<td>SA22-7591</td>
</tr>
<tr>
<td>z/OS MVS Initialization and Tuning Reference</td>
<td>SA22-7592</td>
</tr>
<tr>
<td>z/OS MVS JCL User’s Guide</td>
<td>SA22-7598</td>
</tr>
<tr>
<td>z/OS MVS JCL Reference</td>
<td>SA22-7597</td>
</tr>
<tr>
<td>z/OS MVS System Messages, Vol 5 (EDG-GFS)</td>
<td>SA22-7635</td>
</tr>
<tr>
<td>z/OS MVS System Messages, Vol 9 (IGF-IWM)</td>
<td>SA22-7639</td>
</tr>
<tr>
<td>z/OS MVS Programming: Workload Management Services</td>
<td>SA22-7619</td>
</tr>
<tr>
<td>z/OS MVS Setting Up a Sysplex</td>
<td>SA22-7625</td>
</tr>
<tr>
<td>z/OS MVS System Commands</td>
<td>SA22-7627</td>
</tr>
<tr>
<td>z/OS MVS System Management Facilities (SMF)</td>
<td>SA22-7630</td>
</tr>
<tr>
<td>z/OS Planning for Installation</td>
<td>GA22-7504</td>
</tr>
<tr>
<td>zEnterprise™ System: Ensemble Performance Management Guide</td>
<td>GC27-2607</td>
</tr>
<tr>
<td>zEnterprise System: Ensemble Planning and Configuring Guide</td>
<td>GC27-2608</td>
</tr>
<tr>
<td>IBM z/OS Management Facility Configuration Guide</td>
<td>SA38-0652</td>
</tr>
</tbody>
</table>
## Other Referenced Documents

<table>
<thead>
<tr>
<th>Title</th>
<th>Order Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICS/ESA CICS-RACF Security Guide</td>
<td>SC33-1185</td>
</tr>
<tr>
<td>CICS/ESA Customization Guide</td>
<td>SC33-1165</td>
</tr>
<tr>
<td>CICS/ESA Dynamic Transaction Routing in a CICSpex</td>
<td>SC33-1012</td>
</tr>
<tr>
<td>CICS/ESA Performance Guide</td>
<td>SC33-1183</td>
</tr>
<tr>
<td>CICS/ESA Resource Definition Guide</td>
<td>SC33-1166</td>
</tr>
<tr>
<td>CICSPlex SM Managing Workloads</td>
<td>SC33-1807</td>
</tr>
<tr>
<td><strong>DB2 Administration Guide</strong></td>
<td>SC18-7413</td>
</tr>
<tr>
<td><strong>DB2 Data Sharing: Planning and Administration</strong></td>
<td>SC18-7417</td>
</tr>
<tr>
<td><strong>DB2 SQL Reference</strong></td>
<td>SC18-7426</td>
</tr>
<tr>
<td>DCF SCRIPT/VS User’s Guide</td>
<td>SS44-3191</td>
</tr>
<tr>
<td>IMS/ESA Administration Guide: System</td>
<td>SC26-8730</td>
</tr>
<tr>
<td>IMS/ESA Installation Volume 1 and Volume 2</td>
<td>SC26-8736</td>
</tr>
<tr>
<td><strong>Internet Connection Server User’s Guide</strong></td>
<td>SC31-8204</td>
</tr>
<tr>
<td><strong>z/OS ISPF Dialog Developer’s Guide and Reference</strong></td>
<td>SC34-4821</td>
</tr>
<tr>
<td><strong>z/OS HTTP Server Planning, Installing, and Using</strong></td>
<td>SC34-4826</td>
</tr>
<tr>
<td><strong>z/OS JES2 Commands</strong></td>
<td>SA22-7526</td>
</tr>
<tr>
<td><strong>z/OS JES2 Initialization and Tuning Guide</strong></td>
<td>SA22-7532</td>
</tr>
<tr>
<td><strong>z/OS JES2 Initialization and Tuning Reference</strong></td>
<td>SA22-7533</td>
</tr>
<tr>
<td><strong>z/OS JES2 Installation Exits</strong></td>
<td>SA22-7534</td>
</tr>
<tr>
<td><strong>z/OS JES3 Initialization and Tuning Guide</strong></td>
<td>SA22-7549</td>
</tr>
<tr>
<td><strong>z/OS JES3 Initialization and Tuning Reference</strong></td>
<td>SA22-7550</td>
</tr>
<tr>
<td><strong>z/OS MVS Planning: APPC/MVS Management</strong></td>
<td>SA22-7599</td>
</tr>
<tr>
<td><strong>z/OS RMF Performance Management Guide</strong></td>
<td>SC33-7992</td>
</tr>
<tr>
<td><strong>z/OS RMF Report Analysis</strong></td>
<td>SC33-7991</td>
</tr>
<tr>
<td><strong>z/OS RMF User’s Guide</strong></td>
<td>SC33-7990</td>
</tr>
<tr>
<td>z/OS SOMobjects Configuration and Administration Guide</td>
<td>GC28-1851</td>
</tr>
<tr>
<td><strong>z/OS TSO/E Customization</strong></td>
<td>SA22-7783</td>
</tr>
<tr>
<td><strong>z/OS UNIX System Services Programming Tools</strong></td>
<td>SA22-7805</td>
</tr>
<tr>
<td><strong>z/OS UNIX System Services Planning</strong></td>
<td>GA22-7800</td>
</tr>
<tr>
<td><strong>z/OS UNIX System Services Parallel Environment: Operation and Use</strong></td>
<td>SA22-7810</td>
</tr>
<tr>
<td>WebSphere MQ Workflow Administration Guide</td>
<td>SH12-6289</td>
</tr>
<tr>
<td><strong>PR/SM Planning Guide</strong></td>
<td>GA22-7236</td>
</tr>
<tr>
<td>Support Element Operations Guide</td>
<td>SC28-6802</td>
</tr>
</tbody>
</table>
Information Updates on the Web

For the latest information, visit the WLM/SRM web page at:

http://www.ibm.com/systems/z/os/zos/features/wlm/
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 system 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 Web site, which is available to all users (no login required):

http://publib.boulder.ibm.com/infocenter/zoslnctr/v1r7/index.jsp
How to send your comments to IBM

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or give us any other feedback that you might have.

Use one of the following methods to send us your comments:
1. Send an email to mhvrcfs@us.ibm.com
3. Mail the comments to the following address:
   IBM Corporation
   Attention: MHVRCFS Reader Comments
   Department H6MA, Building 707
   2455 South Road
   Poughkeepsie, NY 12601-5400
   U.S.A.
4. Fax the comments to us as follows:
   From the United States and Canada: 1+845+432-9405
   From all other countries: Your international access code +1+845+432-9405

Include the following information:
• Your name and address
• Your email address
• Your telephone or fax number
• The publication title and order number:
  z/OS V1R13 MVS Planning: Workload Management
  SA22-7602-20
• The topic and page number related to your comment
• The text of your comment.

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.

IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you submit.

If you have a technical problem

Do not use the feedback methods listed above. Instead, do one of the following:
• Contact your IBM service representative
• Call IBM technical support
• Visit the IBM support portal at http://www.ibm.com/systems/z/support/
Summary of Changes

This document contains terminology, maintenance, and editorial 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 1 Release 13

This document contains information previously presented in z/OS MVS™ Planning: Workload Management, SA22-7602-19 which supports z/OS Version 1 Release 12.

New information:

• The information about “I/O Storage Management” on page 25 was added in Chapter 3, “Workload Management Participants,” on page 17.

Updated information:

• Chapter 12, “Defining Special Protection Options for Critical Work,” on page 99 was updated.
• Starting with z/OS V1R13, the HIPERDISPATCH keyword in IEAOPTxx defaults to YES on IBM zEnterprise 196 (z196). For further information, refer to “HiperDispatch Mode” on page 23 in Chapter 3, “Workload Management Participants,” on page 17.
• Chapter 15, “Managing Performance with the Guest Platform Management Provider (GPMP),” on page 135 was updated.

Changes made in z/OS Version 1 Release 12

This document contains information previously presented in z/OS MVS Planning: Workload Management, SA22-7602-18 which supports z/OS Version 1 Release 11.

New information:

• The description of the z/OS Management Facility (z/OSMF) was added. For further information, refer to Chapter 22, “Using the z/OS Management Facility (z/OSMF) to Administer WLM,” on page 229.
• The description of the new option guest platform management provider (GPMP) was added. For further information refer to Chapter 15, “Managing Performance with the Guest Platform Management Provider (GPMP),” on page 135 and to Chapter 21, “Using the WLM ISPF Application,” on page 183.

Updated information:

• The subsystem EWLM no longer supports classification rules of type ETC.
• The "Readers' Comments - We'd like to hear from you" section at the back of the publication has been replaced with a new section "How to send your comments to IBM” on page xvii. The hardcopy mail-in form has been replaced with a page that provides information appropriate for submitting readers’ comments to IBM.

Changes made in z/OS Version 1 Release 11

The book contains information previously presented in z/OS MVS Planning: Workload Management, SA22-7602-17 which supports z/OS Version 1 Release 10.

© Copyright IBM Corp. 1994, 2011 xix
Chapter 1. Why MVS Workload Management?

Before the introduction of MVS workload management, MVS required you to translate your data processing goals from high-level objectives about what work needs to be done into the extremely technical terms that the system can understand. This translation requires high skill-level staff, and can be protracted, error-prone, and eventually in conflict with the original business goals. Multi-system, sysplex, parallel processing, and data sharing environments add to the complexity.

MVS workload management provides a solution for managing workload distribution, workload balancing, and distributing resources to competing workloads. MVS workload management is the combined cooperation of various subsystems (CICS®, IMS/ESA®, JES, APPC, TSO/E, z/OS UNIX System Services, DDF, DB2®, SOM, LSF, and Internet Connection Server) with the MVS workload management (WLM) component.

This information identifies the problems that led to workload management, and describes the high-level objectives for the long term.

Problems

The problems that needed to be solved by workload management include:

• **System Externals:**
  Performance and tuning externals are scattered between MVS and various subsystems, as well as throughout monitoring and reporting products. This is true for the OPT parameters, and, prior to z/OS V1R3, for the MVS installation performance specifications (IPS) and the installation control specifications (ICS). MVS and the subsystems each have their own terminology for similar concepts, each have their own controls, and the controls are not coordinated.

• **Expectations:**
  Many of the MVS externals are geared towards implementation. You tell MVS how to process work, not your expectations of how well it should run work. There is no single way to make sure that important work is getting the necessary system resources.

• **Integration and Feedback:**
  The multiple monitoring and reporting products show many different views of how well MVS is doing, or how well individual subsystems are managing work. Since there is no single way to specify performance goals for your installation, it is also difficult to get feedback from monitors and reporters on how well your installation actually achieved what you expected it to. There is little sense of which reports and fields relate to the externals you specified.

• **Managing towards expectations:**
  Some installations configure their systems to handle a peak load. This could result in inefficient use of expensive resources. Today’s externals do not allow you to reflect your performance expectations for work. Today, a system programmer must completely understand the implications of changing a parameter before he or she can configure an installation to achieve performance objectives. Since there is no direct path between specification and expectation, it is difficult to predict the effects of changing any one control or parameter.
Given this mix of problems, workload management has some high level objectives to provide a solution.

**MVS Workload Management Solution for Today and Tomorrow**

Workload management requires a shift of focus from tuning at a system resources level to defining performance expectations. This requires a basic shift in philosophy towards goal-oriented systems management. The complete workload management solution offers a shift in philosophy in the following areas:

- **Fewer, simpler, and consistent system externals:**
  Workload management provides a way to define MVS externals and tune MVS without having to specify low-level parameters. The focus is on setting performance goals for work, and letting the workload manager handle processing to meet the goals.

- **Externals reflect customer expectations:**
  Workload management provides new MVS performance management externals in a *service policy* that reflects goals for work, expressed in terms commonly used in service level agreements (SLA). Because the terms are similar to those commonly used in an SLA, you can communicate with end-users, with business partners, and with MVS using the same terminology.

- **Expectations-to-feedback correlation:**
  With one common terminology, workload management provides feedback to support performance reporting, analysis, and modelling tools. The feedback describes performance achievements in the same terms as those used to express goals for work.

- **“System Managed” toward expectations:**
  Workload management provides automatic work and resource management support that dynamically adapts as needed. It manages the trade-offs between meeting your service goals for work and making efficient use of system resources.

- **Sysplex externals and management scope:**
  Workload management eliminates the need to micro-manage each individual MVS image, providing a way for you to increase the number of systems in your installation without greatly increasing the necessary skill level.
Chapter 2. What is MVS Workload Management?

Installations today process different types of work with different response times. Every installation wants to make the best use of its resources and maintain the highest possible throughput and achieve the best possible system responsiveness.

With workload management, you define performance goals and assign a business importance to each goal. You define the goals for work in business terms, and the system decides how much resource, such as CPU and storage, should be given to the work to meet its goal.

An installation should know what it expects to accomplish in the form of performance goals, as well as how important it is to the business that each performance goal be achieved. With workload management, you define performance goals for work, and the system matches resources to the work to meet those goals, constantly monitoring and adapting processing to meet the goals. Reporting reflects how well the system is doing compared to its goals.

Figure 1 shows a high level overview of the workload management philosophy. The following section explains the performance administration and performance management processes.

Performance Administration

Figure 1. MVS Workload Management Overview
Performance Administration

Performance administration is the process of defining and adjusting performance goals. Workload management introduces the role of the service level administrator. The service level administrator is responsible for defining the installation's performance goals based on business needs and current performance. This explicit definition of workloads and performance goals is called a service definition. Some installations might already have this kind of information in a service level agreement (SLA). The service definition applies to all types of work, CICS, IMS™, TSO/E, z/OS UNIX System Services, JES, APPC/MVS, LSFM, DDF, DB2, SOM, Internet Connection Server (also referred to as IWEB) and others. You can specify goals for all MVS managed work, whether it is online transactions or batch jobs. The goals defined in the service definition apply to all work in the sysplex.

Because the service definition terminology is similar to the terminology found in an SLA, the service level administrator can communicate with the installation user community, with upper level management, and with MVS using the same terminology. When the service level requirements change, the service level administrator can adjust the corresponding workload management terms, without having to convert them into low-level MVS parameters.

Workload management provides an online panel-based application for setting up and adjusting the service definition. You specify the service definition through this ISPF administrative application.

Workload management provides the capability to collect performance and delay data in context of the service definition. The performance and delay data are available to reporting and monitoring products, so that they can use the same terminology.

Performance Management

Performance management is the process workload management uses to decide how to match resources to work according to performance goals. Workload management algorithms use the service definition information and internal monitoring feedback to check how well they are doing in meeting the goals. The algorithms periodically adjust the allocation of resource as the workload level changes.

For each system, workload management handles the system resources. Workload management coordinates and shares performance information across the sysplex. How well it manages one system is based on how well the other systems are also doing in meeting the goals. If there is contention for resources, workload management makes the appropriate trade-offs based on the importance of the work and how well the goals are being met.

Workload management can dynamically start and stop server address spaces to process work from application environments. Workload management starts and stops server address spaces in a single system or across the sysplex to meet the work's goals.

You can turn over management of batch initiators to workload management, allowing workload management to dynamically manage the number of batch initiators for one or more job classes to meet the performance goals of the work.
In addition to internal feedback monitoring, workload management keeps track of what is happening in the sysplex in the form of real time performance data collection, and delay monitoring. All this information is available for performance monitors and reporters for integration into detailed reports.

**Workload Balancing**

To make the most of workload management, work needs to be properly distributed so that MVS can manage the resources. It is essential that the subsystems distributing work are configured properly for workload distribution in a sysplex. You do this with the controls provided by each subsystem. For example, in a JES2 and JES3 environment, you spread initiator address spaces across each system.

Initial cooperation between MVS and the transaction managers (CICS, IMS, DB2) allows you to define performance goals for all types of MVS-managed work. Workload management dynamically matches resources (access to the processor and storage) to work to meet the goals.

CICS, however, goes further with the CICSpool Systems Manager (CICSpool® SM) to dynamically route CICS transactions to meet the performance goals. CPSM monitors the performance of CICS resources and systems and presents the resources as though they were part of a single system. This type of cooperation greatly improves CICS transaction workload balancing.

Other subsystems also have automatic and dynamic work balancing in a sysplex. For example, DB2 can spread distributed data facility (DDF) work across a sysplex automatically. DB2 can also distribute work in a sysplex through its sysplex query parallelism function. CICS, TSO, and APPC cooperate with VTAM® and workload management in a sysplex to balance the placement of sessions. SOMobjects® can automatically spread its servers across a sysplex to meet performance goals and to balance the work.

For more detail on workload management in different subsystem environments, see Chapter 3, “Workload Management Participants,” on page 17.

**Workload Management Concepts**

The service definition contains all the information about the installation needed for workload management processing. There is one service definition for the entire sysplex. The service level administrator specifies the service definition through the WLM administrative application. The service level administrator sets up “policies” within the service definition to specify the goals for work. A service level administrator must understand how to organize work, and be able to assign it performance objectives.

**What is a Service Definition?**

A service definition consists of:

- One or more *service policies*, which are named sets of overrides to the goals in the service definition. When a policy is activated, the overrides are merged with the service definition. You can have different policies to specify goals intended for different times. Service policies are activated by an operator command, or through the ISPF administrative application utility function.
- *Workloads*, which aggregate a set of service classes together for reporting purposes.
• **Service classes**, which are subdivided into periods, group work with similar performance goals, business importance, and resource requirements for management and reporting purposes. You assign performance goals to the periods within a service class.

• **Report classes**, which group work for reporting purposes. They are commonly used to provide more granular reporting for subsets of work within a single service class.

• **Resource groups**, which define processor capacity boundaries across the sysplex. You can assign a minimum and maximum amount of CPU service units per second to work by assigning a service class to a resource group.

• **Classification rules**, which determine how to assign incoming work to a service class and report class.

• **Application environments**, which are groups of application functions that execute in server address spaces and can be requested by a client. Workload management manages the work according to the defined goal, and automatically starts and stops server address spaces as needed.

• **Scheduling environments**, which are lists of resource names along with their required states. If an MVS image satisfies all of the requirements in a scheduling environment, then units of work associated with that scheduling environment can be assigned to that MVS image.

The following section explains each of the concepts.

### Why Use Service Policies?

A *service policy* is a named set of overrides to the performance goals and processing capacity boundaries in the service definition. A policy applies to all of the work running in a sysplex. Because processing requirements change at different times, service level requirements may change at different times. If you have performance goals that apply to different times or a business need to limit access to processor capacity at certain times, you can define multiple policies. In order to start workload management processing, you must define at least one service policy. You can activate only one policy at a time.

### Organizing Work into Workloads and Service Classes

To workload management, work is a demand for service, such as a batch job, an APPC, CICS, DB2, or IMS transaction, a TSO/E logon, a TSO/E command, or a SOM request. All work running in the installation is divided into *workloads*. Your installations may already have a concept of workload. A workload is a group of work that is meaningful for an installation to monitor. For example, all the work created by a development group could be a workload, or all the work started by an application, or in a subsystem.

Within a workload, you group work with similar performance characteristics into *service classes*. You create a service class for a group of work with similar:

• Performance goals
• Resource requirements
• Business importance to the installation

### Performance Goals

You assign a performance goal to each service class period, such as a response time goal, and you indicate an importance. *Importance* is how important it is to your business that the performance goal be achieved.
There are three kinds of goals: response-time, execution velocity, and discretionary. Response time goals indicate how quickly you want your work to be processed. Since response time goals are not appropriate for all kinds of work, such as long running batch jobs, there are execution velocity goals.

Execution velocity goals define how fast work should run when ready, without being delayed for processor, storage, I/O access, and queue delay. Execution velocity goals are intended for work for which response time goals are not appropriate, such as started tasks, or long running batch work. Discretionary goals are for low priority work for which you do not have any particular performance goal. Workload management then processes the work using resources not required to meet the goals of other service classes.

**Resource Requirements**
Because some work may have variable resource requirements, you can define multiple periods for a service class. Periods are a way of defining different goals for work depending on the amount of resources the work consumes. Typically, periods are used to give shorter transactions more aggressive goals and to give longer running work of the same type less aggressive goals. If you have multiple periods, each period except the last has a duration. *Duration* is the amount of resources, in service units, that the work consumes before it is switched to the goals of the next period.

You can also group work into a service class based on resource requirements. If you have a group of batch work that can consume vast amounts of resources, and you want to limit it, you can define a service class and assign it to a resource group with a maximum amount of capacity. If the work exceeds that capacity, workload management slows the execution rate. Also, if a certain group of work needs a minimum amount of processor capacity, you can set up a service class and assign it to a resource group.

**Business Importance**
When there is not sufficient capacity for all work in the system to meet its goals, business importance is used to determine which work should give up resources and which work should receive more. You assign an importance to a service class period, which indicates how important it is that the goal be met relative to other goals. Importance plays a role only when a service class period is not meeting its goal. There are five levels of importance: lowest (5), low, medium, high, and highest (1).

Figure 2 on page 8 shows an example of the relationship between workloads and service classes. Work in the figure is represented by different size triangles. The IMS workload in the figure, represents all of the IMS work. There are three single-period service classes set up, each with a different importance, and a different response time goal.
Figure 3 shows an example of a workload that is organized by division. In the figure, work is represented by different shapes: circles, squares, and triangles. The workload, OFFICE, represents all of the work in the office division. There are three service classes set up, each a different kind of work in the OFFICE workload. IMSTEST represents the IMS test work, CICS represents all of the CICS work, and JES represents all of the batch work in the OFFICE workload. Each of the service classes has one period with a different response time goal assigned to it.

### Why Use Resource Groups?

Resource groups are a way of limiting or guaranteeing resource capacity. A resource group is a named amount of CPU capacity that you can assign to one or more service classes. For most systems, you can let workload management decide
how to manage the resources in the sysplex and not use resource groups. You set performance goals for work and let workload management adjust to meet the goals.

In some cases, however, you might want to use a resource group, for example, to limit the service that a certain service class can consume. Note that it is recommended that each resource group is assigned to only one service class. Using a resource group is appropriate, for example:

- If you have an SLA that charges for a fixed amount of CPU capacity and you want to ensure that no more is used.
- If you want to ensure that some work with discretionary goals receives some minimum amount of capacity.

Generally, it is recommended to evaluate whether using a resource group fulfills your requirements best, or whether you let workload management take care of managing the resources.

For a resource group, you specify either a minimum or a maximum amount of sysplex capacity in unweighted CPU service units per second, or both.

Figure 4 shows some examples of resource groups.

In the figure, the service class BATHOG is assigned to the resource group LIMIT. BATHOG might include work that consumes processing capacity in huge amounts, so you assign it to a resource group, and limit it to a maximum of 800 CPU service units per second. Also, the goal assigned to BATHOG is discretionary.

The service class TSOMED, on the other hand, is associated with the resource group PROTECT, because according to an SLA, it is contracted to be guaranteed a minimum amount of processing capacity. So the resource group PROTECT is assigned a minimum of 1000 CPU service units per second. Then, when there is sufficient TSOMED work in the system, and TSOMED is underachieving its goal, TSOMED is guaranteed at least 1000 CPU service units per second. Note that if TSOMED is overachieving its goal, then the minimum capacity setting has no effect. Workload management then processes work from both service classes.
making sure that TSOMED gets its minimum amount of capacity, and making sure BATHOG does not consume more than its assigned maximum. Keep in mind the service class goal: if BATHOG is assigned a stringent goal, the goal may never be achievable within the LIMIT resource group capacity. You should determine whether the resource group capacity fulfills your purpose, or the goal. Using both controls could introduce some conflict, and the resource group controls will prevail.

**Assigning Work to a Service Class**

Classification rules are the rules workload management uses to associate a performance goal and/or a resource group with work by associating incoming work with a service class. Optionally, classification rules can also associate incoming work with a report class, similar to a report performance group.

The classification rules for a subsystem are specified in terms of transaction qualifiers such as job name or transaction class. These qualifiers identify groups of transactions that should have the same performance goals and importance. The attributes of incoming work are compared to these qualifiers and, if there is a match, the rule is used to assign a service class to the work. A subsystem can also have a default class for work that does not match any of the rules.

Figure 5 shows how work classification rules associate a service class, and optionally a report class with incoming work.

 Optionally, classification rules can assign incoming work to a report class. You get reporting information in terms of workloads, service classes, and report classes. Use report classes for obtaining data on a subset of work within a service class, or for rolling up data across workloads.

**Why Use Application Environments?**

An *application environment* is a way to group similar server programs together and have workload management dynamically create and delete server address spaces as needed to handle the work. Each application environment typically represents a
named group of server functions that require access to the same application libraries. Depending on the subsystem's implementation of application environments, the scope of server address space management is either confined to a single system or is sysplex-wide. There is also an option to manually start and stop the server address spaces for an application environment if a customer has a special or temporary requirement to control the number of servers independently of workload management.

If you are using a subsystem that takes advantage of application environments, you need to refer to the subsystem documentation for guidance on how to use them for that subsystem. For a list of the IBM-supplied subsystems currently using application environments, see Chapter 13, “Defining Application Environments,” on page 113.

Figure 6 shows an example of how 2 application environments named AE1 and AE2 can be used to handle work requests from a subsystem. In this example, 3 types of work requests named X, Y, and Z are handled. X and Y might be two different kinds of payroll inquiries, and Z might be a loan inquiry.

The names X, Y, and Z are used by clients when making the work requests. The work manager subsystem contains a table or file that associates the work request names with an application environment name; in this example, X and Y are associated with application environment AE1, and Z with AE2. The application environment names AE1 and AE2 are specified to workload management in the service definition stored in the WLM couple data set, and are included in the active policy when a policy is activated.

Each application environment must be assigned a system procedure library (PROCLIB) member name that contains the JCL required to start server address spaces for the application environment. In the example, PROCS1 and PROCS2 are associated with application environments AE1 and AE2, respectively.
When the work manager subsystem receives a type Y work request from a client, the subsystem requests that workload management associate the request with AE1. Workload management determines if a server address space is available to handle the request, or if an address space needs to be created. If a server address space exists, workload management makes the request available for the server. If a server address space does not exist or if more are required, workload management starts a server address space using the startup JCL procedure named PROCS1 which is defined for AE1 in the active policy.

Workload management dynamically creates new server address spaces if they are needed to handle more incoming work and, for certain subsystems such as DB2, decreases the number of server address spaces if less capacity is needed. Refer to Chapter 13, “Defining Application Environments,” on page 113 for a description of how to define application environments to workload management.

**Why Use Scheduling Environments?**

*Scheduling environments* help ensure that units of work are sent to systems that have the appropriate resources to handle them. A scheduling environment is a list of resource names along with their required states. Resources can represent actual physical entities, such as a data base or a peripheral device, or they can represent intangible qualities such as a certain period of time (like second shift or weekend).

These resources are listed in the scheduling environment according to whether they must be set to ON or set to OFF. A unit of work can be assigned to a specific system only when all of the required states are satisfied. This function is commonly referred to as *resource affinity scheduling*.

*Figure 7 on page 13* shows a simple scheduling environment example. The arriving work units, X, Y, and Z, could be batch jobs submitted through JES2 or JES3. Each of the jobs had a scheduling environment associated with it at the time it was submitted (in this case the X and Y jobs are each associated with the A scheduling environment, and the Z job is associated with the B scheduling environment).
JES checks the scheduling environment associated with each arriving batch job and then assigns the work to a system that matches that scheduling environment. In the example, both the X and Y jobs require that both Resource P and Resource Q be set to ON, so those jobs can be initiated only on System 1 in the sysplex. The Z job requires that Resource P be set to ON and that Resource Q be set to OFF. So that job can be initiated only on System 2.

In a sysplex containing only one system, scheduling environments have some degree of usefulness, as JES will hold batch jobs until the required states become satisfied. In a multi-system sysplex, the full power of scheduling environments becomes apparent, as work is assigned only to those systems that have the correct resource states (the resource affinity) to handle that work.

Presently, JES2 and JES3 are the only participants that use scheduling environments, although the concepts could certainly apply to other types of work in the future.

Refer to Chapter 14, “Defining Scheduling Environments,” on page 125 for a description of how to define scheduling environments to workload management.

**Summary of Service Definition and Service Policy Concepts**

When you set up your service definition, you identify the workloads, the resource groups, the service classes, the service class periods, and goals based on your performance objectives. Then you define classification rules and one or more service policies. This information makes up the base service definition.
A two-step process is required before the sysplex starts using a new service
definition. First, you install the service definition onto the WLM couple data set.
Second, you activate one of the service policies from the definition.

With a service policy, you can override specific goals or resource groups in your
base service definition. In a typical scenario, you might define a base service
definition that is appropriate for your normal business hours. Because you need to
have at least one service policy defined, you might create an empty service policy
called NORMAL. While the NORMAL service policy is in effect, there would be no
overrides to the goals or resource groups in the base service definition. If you have
a special business need to change your goals for offshift processing, you might
then also create a service policy called OFFSHIFT. If you were to activate this
policy at the end of the business day (either by invoking the VARY
WLM,POLICY=policynname command or by using the “Activating a Service Policy”
panel in the ISPF application), then the goal overrides in the OFFSHIFT service
policy would be in effect until you switched back to NORMAL the next morning.

Chapter 5, “Defining Service Policies,” on page 35 tells you more about how to
define a policy, and also shows a few examples.

Note that you can override only goals, number and duration of periods, resource
group assignments and values. All of the workloads, service class names,
classification rules, scheduling environments, and application environments
defined in the service definition remain the same for any policy. If you need to
change any of these, you will need to change the base service definition, re-install
the service definition, and then activate a policy from that changed service
definition.

Note, also, that you need to define all of your policies at the outset, while you are
defining the rest of the service definition. Once the service definition is installed,
then you can switch from one defined policy to another. If you need to create a
new policy or change the overrides in an existing policy, you will need to re-install
the service definition with the new or changed policy definition before you can
activate the new policy.

Figure 8 on page 15 shows a service definition with two service policies.
The NORMAL service policy is empty — therefore the originally defined goals, resource group assignments, and resource group attributes remain unchanged when the NORMAL service policy is in effect. When the OFFSHIFT service is in effect, certain goals, resource group assignments, and resource group attributes are overridden.

Figure 8. Service Definition, Including Two Service Policies. The NORMAL service policy is empty — therefore the originally defined goals, resource group assignments, and resource group attributes remain unchanged when the NORMAL service policy is in effect. When the OFFSHIFT service is in effect, certain goals, resource group assignments, and resource group attributes are overridden.
Chapter 3. Workload Management Participants

You use the WLM ISPF administrative application to define your service definition. The administrative application requires the following products:
- TSO/E Version 2.5 plus SPEs, or higher
- ISPF 4.3, or higher

This information describes the work and reporting environments that support workload management.

Workload Management Work Environments

Cooperation between MVS and the subsystem work managers enable the sysplex-wide specification and reporting of goals defined for work in the service policy. You can define goals for the following kinds of work:
- IMS, if you have IMS/ESA Release 5 or higher
- CICS, if you have CICS/ESA 4.1 or higher
- z/OS UNIX System Services
- JES2
- JES3
- APPC
- TSO/E
- SOMobjects
- WebSphere® Application Server objects (CB)
- LSFM
- DDF, if you have DB2 V4.1, or higher
- DB2, if you have DB2 V 5, or higher
- IWEB, if you have Internet Connection Server V 2.2, or higher, Domino® Go Webserver, or IBM http Server for z/OS
- MQSeries® Workflow
- SYSH, if you need to manage non z/OS partitions
- NETV

Arriving work is associated with a service class, and therefore a goal and possibly a resource group. You get feedback from RMF™ as to how well the system is doing in meeting the goals. RMF provides sysplex-wide workload management reporting, providing sysplex-wide as well as single system feedback on the goals through the Post Processor and Monitor III realtime reports. The reports show kinds of delays seen by subsystem work managers such as CICS and IMS. SDSF displays workload management related information.
Workload management understands which address spaces (INITs, AORs, MPPs, BMPs, TORs, FORs) are involved in processing work within a service class, and matches the resources to meet the goal. Information as to how well each system is doing in processing towards a service class goal is recorded in SMF records on each system. For more information about workload management information in RMF reports, see z/OS RMF Report Analysis.

Subsystem Support for Goal Types and Multiple Periods

The types of goals a subsystem supports depends on the workload management services it uses. For the following subsystems that have address space-oriented transactions or use enclaves, you can specify any goal type and multiple periods:

- **Subsystems that have address space-oriented transactions:**
  - APPC
  - JES2
  - JES3
  - z/OS UNIX System Services
  - TSO/E

- **Subsystems that use enclaves:**
  - WebSphere Application Server
  - DB2
  - DDF
  - IWEB
  - MQSeries Workflow
  - LSFM
  - NETV
  - SOMobjects

**Note:** Enclaves are transactions that can span multiple dispatchable units in one or more address spaces, and in the case of multisystem enclaves, one or more address spaces on multiple systems in a parallel sysplex. See MVS Programming: Workload Management Services for more information on enclaves.
The CICS and IMS subsystems do not use enclaves, but use a different set of WLM services to support their transactions to WLM. Therefore, they support only response time goals, either percentile or average, and single period service classes.

**Subsystem-Specific Performance Hints**

Based on installation experiences, here are some subsystem-specific performance hints:

- **Watch out for increased CPU usage by the WLM address space due to a high CICS MAXTASK setting.** For CICS 4.1 and higher releases, WLM collects delay samples for each performance block. Because the number of performance blocks created is based on the MAXTASK value (a value of 100 means 100 performance blocks created per region), a MAXTASK value that is too high can cause a large sampling overhead when a CICS workload is switched to goal mode. If MAXTASK has been set to an arbitrarily high value, it should be reduced to a true “high water mark” value.

- **Watch out for work defaulting to SYSOTHER.** Work in subsystems that use enclaves (see “Subsystem Support for Goal Types and Multiple Periods” on page [18] for a list of these subsystems) can suffer performance degradation if left unclassified in the service definition. If you do not add classification rules for this work in your service definition, it will be assigned to the SYSOTHER service class, which has a discretionary goal. Using the WLM application, you need to add classification rules to assign the work to service classes with appropriate response time or velocity goals.

As a general rule, it’s a good idea to keep an eye on the SYSOTHER service class through RMF or another monitor. Any service accumulating in the SYSOTHER service class is a signal that you have unclassified work in your system.

For the latest information on these topics and others, see the WLM/SRM website at:


**Workload Balancing**

Workload management allocates resources to meet goals of the work that arrives. System programmers must use the existing methods of routing and scheduling work for subsystems except for those listed. For subsystems not exploiting workload balancing or routing services, if you want to balance your work across all MVS images in a sysplex, the system programmer must set the routing controls to either balance the arrival of work, or to ensure that all MVS images are equal candidates for processing work.

Examples of subsystems that can automatically balance work in a sysplex include:

- CPSM provides goal-oriented routing based on the goal defined for CICS work in the workload management service policy.
- DB2 V 4.1 provides automatic and dynamic work balancing in a sysplex for distributed data facility (DDF) work.
- DB2 V 5 provides additional automatic work balancing through its sysplex query parallelism function.
- SOMobjects uses application environments to help balance object class binding requests in a sysplex.
- CICS V 4.1, DB2 V 4.1, TSO/E V 2.5, and APPC cooperate with VTAM 4.4 and workload management in a sysplex to manage session placement. New sessions
for these subsystems are directed to the appropriate systems in the sysplex to balance work and meet performance goals.

- JES2 and JES3 provides automatic and dynamic placement of initiators for WLM-managed job classes. z/OS 1.4 together with JES2 1.4, and z/OS 1.5 together with JES3 1.5 provide Initiator Balancing, so that already available WLM managed initiators can be reduced on fully loaded systems and increased on low loaded systems to improve overall batch work performance and throughput over the sysplex.

- WebSphere Application Server cooperates with WLM in a sysplex to balance work among application control regions and to meet performance goals.

### Workload Management in a CPSM Environment

[Figure 10](#) shows workload management in a CICS with CPSM environment.

WLM recognizes that the terminal owning region (TOR) and the application owning region (AOR) on one or more systems are involved in processing CICS transactions. Using RMF Monitor I, you can get reporting information on the CICS response times, and on any execution delays experienced by a service class period for a single system or for the sysplex.

In a CPSM environment, WLM provides the CICS service class goal to CPSM. If the goal is an average response time goal and you specified the dynamic goal-oriented algorithm for CPSM, then CPSM uses the transaction's goal to help decide where to route the transaction for processing to meet the goal. If the goal is a percentile response time goal, CPSM reverts to its “shortest queue” algorithm since it only has average response time data available to it. Percentile goals are still preferred for any workload that can have a few unusually long transactions distorting the average response time. Following is a summary of the two CPSM algorithms:

**Shortest Queue**

Send the CICS transaction to the AOR that has the shortest queue of pending work, but prefer AORs on a local MVS image, and be sure to avoid “unhealthy” AORs.

**Goal**

Send the transaction to an AOR that has been successfully meeting the goal, but prefer AORs on a local image and be sure to avoid “unhealthy” AORs. If the goal is a percentile goal, use the shortest queue.

For more information, see CICSPlex SM Managing Workloads.
Workload Management in a DB2 Distributed Data Facility (DDF) Environment

The definition of a response time for the enclaves used to manage DDF transactions depends upon several parameters, including DB2 install parameters and the attributes used when binding the package or plan. For more information, see [DB2 Administration Guide](#).

Batch Workload Management

Workload management can dynamically manage the number of batch initiator address spaces in a JES2 or JES3 environment. You can selectively turn over control of the batch initiator management to WLM for one or more job classes. WLM will start new initiators, as needed, to meet the performance goals of this work.

By specifying or defaulting `MODE=JES` on the JES2 `JOBCLASS` statement or the JES3 `GROUP` statement, you indicate that the initiators for the job class should be JES-managed, as in the past. By specifying `MODE=WLM`, you keep the job class in JES-managed mode. (JES will manage the batch initiators for that job class, in the same way it has in prior releases.) By specifying `MODE=WLM`, you put that class into WLM-managed mode.

You can switch as many job classes to WLM-managed mode as you wish. You can easily switch any job class back to JES-managed mode by using the JES2 `STJOBCLASS` command or the JES3 `MODIFY` command.

Notes:

1. If you have velocity performance goals set for the work running on WLM-managed batch initiators, be aware that the initiation delays will be figured into the velocity formula. This will affect your velocity values and probably require some adjustment to your goals. See Chapter 8, “Defining Service Classes and Performance Goals,” on page 47 for information on defining velocity goals.

2. All jobs with the same service class should be managed by the same type of initiation. For example, if jobs in job classes A and B are classified to the HOTBATCH service class, and `JOBCLASS(A)` is `MODE=WLM`, while `JOBCLASS(B)` is `MODE=JES`, workload management will have a very difficult time managing the goals of the HOTBATCH service class without managing class B jobs.

3. You can use the JES2 `JOBCLASS` parameter to specify a default scheduling environment, thereby saving the effort of changing JCL jobcards or writing a specific JES2 exit to assign the scheduling environment.

4. If a WLM initiator is experiencing ABEND822s, there are two ways to recycle the initiator:
   - If you can determine the ASID of the initiator that is abending, you can stop it by issuing the command `P INIT,A=asid`. The initiator does not need to be idle at the time that you enter the command. If the initiator is busy processing a job, it will stop itself after the job finishes. WLM will automatically replace the initiator with a new one.
   - If you cannot determine the ASID, or if you want to recycle all initiators as part of a regular cleanup process, you can enter the commands `$P XEQ` and `$S XEQ`. The `$P XEQ` command causes all WLM initiators on that system to be “flagged” to terminate. The `$S XEQ` command enables WLM to start new initiators (without needing to wait for the old initiators to end). Beware that the `$P XEQ` command purges WLM's history which tells it how many
initiators are needed for each service class. It may take some time for WLM to build up the same number of initiators that existed before.

See the following JES2 and JES3 manuals for more information on WLM-managed JES2 job classes:
- z/OS JES2 Initialization and Tuning Guide
- z/OS JES2 Initialization and Tuning Reference
- z/OS JES2 Commands
- z/OS JES3 Initialization and Tuning Guide
- z/OS JES3 Initialization and Tuning Reference
- z/OS JES3 Commands

The following other functions exist to help you manage batch work in a JES environment:
- A new work qualifier, PRI, which allows you to use the job priority when defining work classification rules. See Chapter 9, “Defining Classification Rules,” on page 57.
- Scheduling environments, which allow you to define resource requirements for incoming work, ensuring that the work will be scheduled on a system within the sysplex only if the resource settings on that system satisfy those requirements. See Chapter 14, “Defining Scheduling Environments,” on page 125.

Multisystem Enclave Support

With multisystem enclave support, enclaves can run in multiple address spaces spanning multiple systems within a parallel sysplex. As in a single system enclave, the work will be reported on and managed as a single unit.

UNIX System Services Parallel Environment uses multisystem enclaves to run parallel jobs. With all tasks of the job running in the same enclave, WLM can manage all of the work to a single performance goal.

See Chapter 17, “Defining a Coupling Facility Structure for Multisystem Enclave Support,” on page 157 for more information on setting up the SYSZWLMM_WORKUNIT coupling facility structure, a prerequisite to multisystem enclave support.

See the “Creating and Using Enclaves” information in z/OS MVS Programming: Workload Management Services for more information on multisystem enclaves.

See z/OS UNIX System Services Parallel Environment: Operation and Use for more information on UNIX System Services Parallel Environment.

Intelligent Resource Director

The Intelligent Resource Director (IRD) extends the concept of goal-oriented resource management by allowing you to group logical partitions that are resident on the same physical server, and in the same sysplex, into an “LPAR cluster.” This gives Workload Management the ability to manage resources, both processor and DASD I/O, not just in one single image but across the entire cluster of logical partitions.

See Chapter 18, “The Intelligent Resource Director,” on page 161 for more information.
**HiperDispatch Mode**

This information briefly describes the HiperDispatch mode.

**Overview**

In addition to the performance improvements available with the IBM System z10™ processors, z/OS workload management and dispatching are enhanced to take advantage of the System z10 hardware design. A mode of dispatching called HiperDispatch provides additional processing efficiencies.

The HiperDispatch mode aligns work to a smaller subset of processors to maximize the benefits of the processor cache structures, and thereby, reduce the amount of CPU time required to execute work. Access to processors has changed with this mode, and as a result, LPAR weights prioritization of workloads via WLM policy definitions becomes more important.

**The Concept of HiperDispatch Mode**

Without HiperDispatch, for all levels of z/OS, a TCB or SRB may be dispatched on any logical processor of the type required (standard, zAAP or zIIP). A unit of work starts on one logical processor and subsequently may be dispatched on any other logical processor. The logical processors for one LPAR image will receive an equal share for equal access to the physical processors under PR/SM™ LPAR control. For example, if the weight of a logical partition with four logical processors results in a share of two physical processors, or 200%, the LPAR hypervisor will manage each of the four logical processors with a 50% share of a physical processor. All logical processors will be used if there is work available, and they typically have similar processing utilizations.

With HiperDispatch mode, work can be managed across fewer logical processors. A concept of maintaining a working set of processors required to handle the workload is introduced. In the previous example of a logical partition with a 200% processor share and four logical processors, two logical processors are sufficient to obtain the two physical processors worth of capacity specified by the weight; the other two logical processors allow the partition to access capacity available from other partitions with insufficient workload to consume their share. z/OS limits the number of active logical processors to the number needed based on partition weight settings, workload demand and available capacity. z/OS also takes into account the processor topology when dispatching work, and it works with enhanced PR/SM microcode to build a strong affinity between logical processors and physical processors in the processor configuration.

**Processor Categories**

The logical processors for a partition in HiperDispatch mode fall into one of the following categories:

- Some of the logical processors for a partition may receive a 100% processor share, meaning this logical processor receives an LPAR target for 100% share of a physical processor. This is viewed as having a high processor share. Typically, if a partition is large enough, most of the logical partition’s share will be allocated among logical processors with a 100% share. PR/SM LPAR establishes a strong affinity between the logical processor and a physical processor, and these processors provide optimal efficiencies in HiperDispatch mode.

- Other logical processors may have a medium amount of physical processor share. The logical processors would have a processor share greater than 0% and up to 100%. These medium logical processors have the remainder of the
partition’s shares after the allocation of the logical processors with the high
share. LPAR reserves at least a 50% physical processor share for the medium
processor assignments, assuming the logical partition is entitled to at least that
amount of service.

• Some logical processors may have a low amount, or 0%, of physical processor
share. These “discretionary” logical processors are not needed to allow the
partition to consume the physical processor resource associated with its weight.
These logical processors may be parked. In a parked state, discretionary
processors do not dispatch work; they are in a long term wait state. These
logical processors are parked when they are not needed to handle the partition’s
workload (not enough load) or are not useful because physical capacity does not
exist for PR/SM to dispatch (no time available from other logical partitions).

When a partition wants to consume more CPU than is guaranteed by its share
and other partitions are not consuming their full guaranteed share, a parked
processor can be unparked to start dispatching additional work into the
available CPU cycles not being used by other partitions. An unparked
discretionary processor may assist work running on the same processor type.

When examining an RMF CPU activity report in HiperDispatch mode, one may
now see very different processing utilizations across different logical processors of
a logical partition. For further information, refer to z/OS RMF Performance
Management Guide.

Setting of the HiperDispatch Mode in SYS1.PARMLIB

The HiperDispatch state of the system is determined by the number of logical
processors defined on an LPAR and the HIPERDISPATCH=YES\NO keyword
IEAOPTxx of SYS1.PARMLIB.

All partitions with more than 64 logical processors defined at IPL are forced to run
with HIPERDISPATCH=YES. LPARs with more than 64 logical processors defined
are also unable to switch into HIPERDISPATCH=NO after IPL.

For all partitions with less than 64 logical processors, HiperDispatch is enabled or
disabled by the HIPERDISPATCH=YES\NO keyword in parmlib member
IEAOPTxx. This parameter can be changed dynamically with the use of the SET
OPT command. This enables the operating system to choose the desired mode of
operation.

When a new hardware generation is installed, for any z/OS image(s) that are
running with HiperDispatch disabled, the system programmer should reevaluate
whether those z/OS image(s) should be migrated to running with HiperDispatch
enabled in the new environment.

On z10 on any z/OS release, HiperDispatch disabled is the default. However,
customers are encouraged to run with HiperDispatch enabled on z10 to take
advantage of the processing benefits.

Beginning with z/OS V1R13 on IBM zEnterprise 196 (z196), HiperDispatch enabled
is the default. With z/OS V1R13 running on a z196, z/OS partitions with share
greater than 1.5 physical processors will typically experience improved processor
efficiency with HiperDispatch enabled. z/OS partitions with share less than 1.5
physical processors typically do not receive a detectable performance improvement
with HiperDispatch enabled, but IBM recommends running those LPARs with
HiperDispatch enabled when the performance improvement is greater than or
equal to HiperDispatch disabled.
There are no new hardware controls or settings to enable use of HiperDispatch within a logical partition; however, the existing “control global performance data” security setting must be enabled on HMC for proper operation of HiperDispatch in a logical partition. HiperDispatch cannot effectively utilize vertical low processors when other partitions are active on the system and “global performance data” is not enabled.

For further information about the HIPERDISPATCH parameter, refer to z/OS MVS Initialization and Tuning Reference.

I/O Storage Management

Workload management can pass service class importance and goal information to the storage I/O priority manager in the IBM System Storage DS8000 series. The information enables the storage I/O priority manager to provide favored processing for I/O requests of important z/OS workloads that are not achieving their goals.

The storage I/O priority manager may throttle I/O requests to facilitate favored access to storage server resources for other I/O requests. The storage I/O priority manager analyzes the properties of the service class period associated with an I/O request and determines whether the I/O request should be favored, or throttled.

Handling Service Class Periods with a Response Time Goal

For service class periods with a response time goal, the goal achievement and specified importance are analyzed. Service class periods that exceed their goal may be throttled if there are service class periods that do not achieve their goal. Service class periods that miss their goal might only be throttled if there are service class periods with a higher importance that do not achieve their goal. In detail, service class periods are processed as follows:

- Service class periods with a response time goal and importance 1 that do not achieve their goal are favored. They are not throttled when they exceed their goal.
- Service class periods with a response time goal and importance 2 that do not achieve their goal are favored. They are not throttled when they exceed their goal, except when more important work is missing its goals.
- Service class periods with a response time goal and importance 3 that do not achieve their goal are favored. They might be moderately throttled when they exceed their goal or more important work is missing its goals.
- Service class periods with a response time goal and importance 4 that do not achieve their goal are moderately favored. They might be throttled when they exceed their goal or more important work is missing its goals.
- Service class periods with a response time goal and importance 5 may be throttled when they exceed their goal or more important work is missing its goals. They are not favored when they miss their goal.

The level of favored processing or throttling that the I/O requests of a service class period with a certain importance receive, depends on the goal achievement, that is, the performance index (PI), of the service class period. A service class period does not receive favored processing if its performance index is between 0.9 and 1.4. The more the PI of a service class period exceeds 1.4, the more it is favored: moderate, strong, or very strong. The more the PI of a service class period falls below 0.9, the more it may be throttled: moderate, strong, or very strong.
Table 1 shows the possible levels of favored processing, or throttling in relationship to the importance and goal achievement of a service class period with response time goal:

**Table 1. Levels of Processing**

<table>
<thead>
<tr>
<th>Importance</th>
<th>Service class period misses goal (PI is higher than or equals 1.4)</th>
<th>Service class period exceeds goal (PI is lower than or equals 0.9)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Favored processing - moderate, strong, or very strong</td>
<td>Not throttled</td>
</tr>
<tr>
<td>2</td>
<td>Favored processing - moderate, strong, or very strong</td>
<td>Not throttled, except when there are service class periods with higher importance that miss their goal</td>
</tr>
<tr>
<td>3</td>
<td>Favored processing - moderate or strong</td>
<td>Throttled - moderate</td>
</tr>
<tr>
<td>4</td>
<td>Favored processing - moderate</td>
<td>Throttled - moderate or strong</td>
</tr>
<tr>
<td>5</td>
<td>No favored processing</td>
<td>Throttled - moderate, strong or very strong</td>
</tr>
</tbody>
</table>

**Handling Service Class Periods with a Velocity Goal**

For service class periods with a velocity goal, the specified velocity goal and importance are taken into account.

Service class periods with a high importance (1 or 2) and a high velocity goal are most likely being favored. Service class periods with a low importance and a low velocity goal might be throttled.

Because the goal achievement of service class periods with a velocity goal is not taken into account, I/O requests of these service class periods might be throttled even when they miss their goal and might be favored even when they exceed their goal.

**Handling Other I/O Requests**

I/O requests associated with the system-provided service classes SYSTEM, SYSSTC, or SYSSTC1 - SYSSTC5 are not managed by the I/O priority manager.

I/O requests associated with service class periods that have a discretionary goal may be throttled, but will never be favored.

**Controlling the Information Passed to the I/O Manager**

Parameter STORAGESERVERMGT=YES|NO in IEAOPTxx of SYS1.PARMLIB controls whether service class importance and goal information is passed to the storage I/O priority manager.

STORAGESERVERMGT=YES specifies that SRM should pass service class importance and goal information to the storage I/O priority manager. The default is STORAGESERVERMGT=NO. Before specifying STORAGESERVERMGT=YES, verify that your IBM System Storage DS8000 model incorporates the I/O priority manager feature.

Throttle delays introduced by the storage I/O priority manager are reflected in control unit queue delays. Therefore, if STORAGESERVERMGT=YES is specified,
control unit queue delays are not considered when the execution velocity is calculated for service class periods with a velocity goal.

If you have significant control unit queue delays in your installation, you might have to adjust the velocity goal of service class periods when you specify STORAGESERVERMGT=YES.

For further information about the STORAGESERVERMGT parameter, refer to z/OS MVS Initialization and Tuning Reference.

Non z/OS Partition CPU Management

If you have logical partitions running LINUX or other non z/OS systems, you can manage CPU resources across these logical partitions in accordance with workload goals. Non z/OS partition CPU management does not support the management of partitions running z/OS systems.

Non z/OS partition CPU management allows WLM to shift weight from either z/OS or non z/OS partitions to another z/OS or non z/OS partition in the same LPAR cluster.

A new subsystem type SYSH is introduced for LINUX CPU management. SY and PX are the qualifiers which are valid for SYSH. No resource groups can be associated with service classes for SYSH. Also CPU protection cannot be assigned. SYSH supports velocity goals with single periods, but no discretionary goals.

In order to activate non z/OS CPU Management, see “Enabling Non z/OS CPU Management” on page 168.

Workload Management and Workload License Charges

As part of the z/OS support of Workload License Charges, you can set a defined capacity limit, also called a “soft cap,” for the work running in a logical partition. This defined capacity limit is measured in millions of service units per hour (MSUs). It allows for short-term spikes in the CPU usage, while managing to an overall, long-term rolling 4-hour average. It applies to all work running in the partition, regardless of the number of individual workloads the partition may contain.

See the PR/SM Planning Guide for more information on how to set a defined capacity limit.

WLM enforces the defined capacity limit by tracking the partition's CPU usage and continually averaging it over the past four hours. Spikes higher than the defined capacity limit are possible, as long as they are offset by low points that keep the 4-hour average at or below the limit. When this 4-hour average goes over the defined capacity limit, then WLM caps the partition. At that point, it can use no more than the defined capacity limit, until the average drops below the limit.

At IPL, WLM defaults to a 4-hour time interval that contains no partition CPU usage. This allows the defined capacity limit to be exceeded at IPL and therefore avoids capping during this time.

Consider the following example:
This partition has a 50 MSU defined capacity limit, as shown by the grey dashed line. The solid black line is the actual MSU consumption of the partition, and the dashed black line is the 4-hour MSU average. At IPL (09:00), the partition’s CPU usage starts with more than 100 MSUs. The 4-hour average, however, is below the defined capacity limit at that point, due to the fact that WLM is using the default 4-hour interval time containing no partition CPU usage. No action is taken. Just before 11:00, the 4-hour average hits the defined capacity limit. Now WLM caps the partition’s CPU usage at 50 MSUs.

WLM allows the CPU usage to remain at the defined capacity limit. Therefore, the 4-hour average can continue to go up. In this example, this occurs because the low usage numbers of the 4-hour default time interval at IPL are falling off the back end of the 4-hour “horizon”, and being replaced by the new usage numbers starting at IPL (09:00). This is a consequence of managing to the 4-hour average.

In summary, WLM caps a partition’s CPU usage only when the 4-hour average reaches the defined capacity limit. Before that point, you may see CPU usage spikes over the limit. You may see the 4-hour average go over the limit after capping, until the passage of time gradually brings the average back down to the limit.

**Defining the Capacity of a Partition**

WLM tells the PR/SM Hypervisor to cap the partition, as described in the previous example, in one of three possible ways, depending on the relative sizes of the defined capacity limit and the weight of the partition. The weight defined for a partition specifies the share of the partition of the available capacity of the machine. The following possible cap mechanisms are available:

- **Option 1:**
  WLM tells PR/SM to fully cap the partition at its weight if the capacity share based on the weight definition is equal to the defined capacity limit and the 4-hour rolling average reaches the defined capacity limit.

- **Option 2:**

---

*Figure 11. Workload Management Enforcing a Defined Capacity Limit*

This partition has a 50 MSU defined capacity limit, as shown by the grey dashed line. The solid black line is the actual MSU consumption of the partition, and the dashed black line is the 4-hour MSU average. At IPL (09:00), the partition’s CPU usage starts with more than 100 MSUs. The 4-hour average, however, is below the defined capacity limit at that point, due to the fact that WLM is using the default 4-hour interval time containing no partition CPU usage. No action is taken. Just before 11:00, the 4-hour average hits the defined capacity limit. Now WLM caps the partition’s CPU usage at 50 MSUs.

WLM allows the CPU usage to remain at the defined capacity limit. Therefore, the 4-hour average can continue to go up. In this example, this occurs because the low usage numbers of the 4-hour default time interval at IPL are falling off the back end of the 4-hour “horizon”, and being replaced by the new usage numbers starting at IPL (09:00). This is a consequence of managing to the 4-hour average.

In summary, WLM caps a partition’s CPU usage only when the 4-hour average reaches the defined capacity limit. Before that point, you may see CPU usage spikes over the limit. You may see the 4-hour average go over the limit after capping, until the passage of time gradually brings the average back down to the limit.

**Defining the Capacity of a Partition**

WLM tells the PR/SM Hypervisor to cap the partition, as described in the previous example, in one of three possible ways, depending on the relative sizes of the defined capacity limit and the weight of the partition. The weight defined for a partition specifies the share of the partition of the available capacity of the machine. The following possible cap mechanisms are available:

- **Option 1:**
  WLM tells PR/SM to fully cap the partition at its weight if the capacity share based on the weight definition is equal to the defined capacity limit and the 4-hour rolling average reaches the defined capacity limit.

- **Option 2:**
In case the capacity based on the weight is smaller than the defined capacity limit it is not possible to cap the partition at its weight for the complete time period until the 4-hour rolling average drops below the defined capacity limit because the partition would not be able to use the capacity it is entitled to.

WLM now defines a cap pattern which caps the partition at its weight and then removes the capping during another period of time. Over time this looks as if the partition is constantly capped at its defined capacity limit.

The cap pattern depends on the difference of the capacity based on the weight versus the defined capacity limit. If the capacity based on the weight is rather small compared to the defined capacity limit it looks as if the partition is capped very drastically during short periods of time. Therefore, for configurations requiring the capacity based on the weight to be smaller than the defined capacity limit, it is recommended to keep both definitions as close as possible.

**Option 3:**

In case the capacity based on the weight is greater than the defined capacity limit, WLM causes PR/SM Hypervisor to define a phantom weight, because it is not possible for WLM to cap the partition at its capacity based on the weight. A phantom weight pretends a utilization of the CEC for a particular partition which makes it possible to cap the partition at the defined capacity limit.

Option 1 and 3 are the recommended ways to specify defined capacity limits and weights because they provide a capping behavior which is very smooth. Option 3 has another implication which should be discussed in more detail. The phantom weight is calculated as follows:

\[
\text{Phantom Weight (P)} = \frac{\text{CEC Capacity}}{\text{Defined Capacity Limit (P)}} \times \text{Partition Weight (P)}
\]

where P is the partition for which the capacity should be defined.

The value specified for a phantom weight can amount to a maximum of up to a 1000 times the number of active partitions. This is a restriction of the PR/SM Hypervisor. With this maximum in mind it is not possible to specify a defined capacity limit which is very small compared to the capacity based on the weight definition.

**Example**

In the following example the CEC capacity is 500 MSUs. There are three partitions with the following definitions:

<table>
<thead>
<tr>
<th>Partition</th>
<th>Weight Definition</th>
<th>Defined Capacity Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>500</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>500</td>
<td></td>
</tr>
</tbody>
</table>

The sum of the weights is 1500. For partition 2 a very small defined capacity limit of 5 MSUs has been specified. Three partitions are defined; therefore the maximum
phantom weight which can be specified by WLM to cap partition 2 is limited to 3000. Based on the formula shown in Figure 12 on page 29, a phantom weight of 48500 would be required to perform the requested capping operation. This is not possible and will result in a situation where the partition is not capped at all, and ABEND code X'15F' with reason code 184 is issued.

To correct the definition, perform one of the following actions:

- Increase the defined capacity limit to a value which allows WLM to specify a phantom weight of less or equivalent to 3000. For the example at hand that would be a defined capacity limit of 56 or more if the weights and number of partitions remain unchanged.

- Adjust the weight of the partition. In order to cap the partition at 5 MSU it is necessary to define the weight for partition 2 less or equivalent to 45 assuming that the weight for the two other partitions should remain 500.

**Defining Group Capacity**

Group capacity limit is an extension of the defined capacity. It allows an installation to define a “soft cap” for multiple logical partitions of the same CEC all running z/OS V1R8 or higher. The group limit is a defined capacity (soft cap) for all partitions defined in the group. The capacity group is defined on the Hardware Management Console. Each capacity group has a name and a defined capacity which becomes effective to all partitions in the group.

See the *PR/SM Planning Guide* for more information on how to define a capacity group.

WLM uses the weight definitions of the partitions and their actual demand to decide how much CPU can be consumed by each partition in the group. In the following example a capacity group is defined which consists of three partitions MVS1, MVS2 and MVS3. The group limit is defined to 50 MSU and the weights of the partitions are shown in Table 3:

<table>
<thead>
<tr>
<th>Partition</th>
<th>Weight</th>
<th>Share [MSU]</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVS1</td>
<td>100</td>
<td>16,7</td>
</tr>
<tr>
<td>MVS2</td>
<td>50</td>
<td>8,3</td>
</tr>
<tr>
<td>MVS3</td>
<td>150</td>
<td>25</td>
</tr>
</tbody>
</table>

The total weight of all partitions in the group is 300. Based on the weight definitions, each partition gets an entitled share of the group capacity of 50 MSU. The entitled share is important to decide how much MSU can be used by each partition if the 4-hour rolling average of the group exceeds the group capacity limit. The share is also shown in Table 3.

[Figure 13 on page 31](#) shows an example how the partitions use their entitled capacity. At 07:00 p.m. all three partitions are started. In the beginning, only partition MVS1 and MVS2 use approximately 60 MSU. No work is running on partition MVS3. Therefore its measured consumption is very small. WLM starts to cap the partitions when the 4-hour rolling average for the combined usage exceeds the 50 MSU limit. This happens around 09:00 p.m. At that point, MVS1 is reduced to about 30 MSU and MVS2 to about 20 MSU. MVS3 still does not demand much CPU. Therefore the available MSU of the group can be consumed by MVS1 and MVS2.
Around 11:00 p.m. work is started on MVS3. A small spike can be observed when WLM recognizes that the third partition starts to demand its share of the group. After that spike MVS3 gets up to 25 MSU of the group because its weight is half of the group weight. MVS1 is reduced to 16.7 MSU and MVS2 to 8.3 MSU. Based on variation in the workload the actual consumption of the partitions can vary but the group limit of 50 MSU is always met on average.

The work on MVS3 stops around 04:00 p.m. At that point, a small negative spike can be observed and afterwards the capacity is consumed again only by the partitions MVS1 and MVS2.

Group capacity can be combined with all other existing management capabilities of the z/OS Workload Manager:
- Group capacity can be combined with an individually defined capacity for a partition. The partition will always honor the individually defined capacity.
- It is possible to define multiple capacity groups on a CEC. A partition can only belong to one capacity group at a given point in time.

Note that WLM only manages partitions which comply with the following rules within a group:
- The partitions must not be defined with dedicated processors.
- The partitions must run with shared CPUs and wait completion must be 'No'.
- The operating system must be z/OS V1R8, or higher.
- A hardware cap (that is “Initial Capping” checked on the Change Logical Partitions Controls panel) which limits the partition to its weight is not allowed for a partition being managed in a capacity group.

All partitions which do not conform to these rules are not considered part of the group. WLM will dynamically remove such partitions from the group and manage the remaining partitions towards the group limit.
Group capacity functions together with IRD weight management while the partitions in the group are not subject to capping. No weight moves take place for partitions as long as the group is being capped.

Group capacity does not work when z/OS is running as a VM guest system.

### Workload Management with Other Products

For subsystem work managers, if you do not have the product or product release supporting workload management, you can define goals only for the subsystem regions. Since subsystem regions are treated as batch jobs or started tasks, a velocity goal is most appropriate. All started tasks are managed on the system where they are started.

For example, if you have a CICS release that is not at least at the CICS/ESA 4.1 level, you could define velocity goals for the TORS and AORS. You could not manage CICS transactions to response time goals.

For reporting or monitoring products, you should check whether they support workload management. Vendor information should explain whether or not they report on the workload management activity.
Chapter 4. Setting up a Service Definition

This information describes how to set up a service definition based on your performance objectives. A service definition contains all of the information necessary for workload management processing. Based on this information, you should be able to set up a preliminary service definition using the worksheets provided. Then, you can enter your service definition into the ISPF administrative application with ease.

The service definition is the way to express your installation's business goals to your sysplex. In order to do this, you must understand your installation's business environment from the following areas:

- What are your installation's revenue-earning workloads?
- Is there a business priority to the workloads?
- Do you understand what kind of service users expect and the time-frame they expect it in?
- Do you understand the service you can deliver?

Even if your installation does not currently have an SLA, or other written performance objectives, users most often have service expectations.

- How can you use monitoring and performance products to gather information?

Specifying a Service Definition

You define a minimum of one policy in a service definition. A service definition contains workloads, service classes, and classification rules. Optionally, it contains resource groups, report classes, and application environments. A service definition also includes one or more policies. A policy is a set of overrides to the goals and resource group limits in the service definition. In a service definition, you also specify whether you want workload management to manage I/O priorities based on your service class goals. A service definition has an identifying name and description.

There is one notepad available for the service definition. You can update the notepad whenever you are creating or modifying any of the other parts of a service definition, such as the policy, workload, service class, resource group, and classification rules. All of the information in a service definition is available to reporting products, except for the notepad information and the classification rules. The notepad information is provided for a history log, and change management.

<table>
<thead>
<tr>
<th>Name</th>
<th>Service definition name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Description of the service definition</td>
</tr>
<tr>
<td>Name (required)</td>
<td>Eight character identifier of the service definition.</td>
</tr>
<tr>
<td>Description (optional)</td>
<td>An area of 32 characters to describe the service definition. For example, you could include the time period this service definition is intended to cover.</td>
</tr>
</tbody>
</table>
Storing Service Definitions

You can work in the ISPF administrative application with one service definition at a time. In order to make the service definition accessible to all systems in the sysplex, you store the service definition on a WLM couple data set. Only one service definition can be installed on the WLM couple data set at a time.

If you want to work on more than one service definition at a time, you can keep each in a distinct MVS partitioned data set (PDS), or in an MVS sequential data set (PS). As an MVS service definition data set, the service definition is subject to all the same functions as an MVS data set. You can restrict access to the service definition data set, send it, and copy it, as you can any MVS data set.

The service definition must be installed on a WLM couple data set, and a service policy activated. Only service policies in the service definition installed on the WLM couple data set can be activated. A WLM couple data set can have automatic backup. For more information about allocating and using a WLM couple data set, see Chapter 16, “Workload Management Migration,” on page 141.

Defining the Parts of a Service Definition

This information explains how to set up a service definition by defining each of its parts: policies, workloads, resource groups, application environments, service classes, classification rules, and report classes. When you set up your service definition, you should define its parts in the following order:

1. Service policy

2. Workloads

3. Resource groups

4. Service Classes
   Chapter 8, “Defining Service Classes and Performance Goals,” on page 47.

5. Service policy overrides
   “Using Policy Overrides” on page 36.

6. Classification rules
   Chapter 9, “Defining Classification Rules,” on page 57.

7. Report classes

8. Service coefficients and options
   Chapter 11, “Defining Service Coefficients and Options,” on page 93. Service definition options include I/O priority management.

9. Application environments

10. Scheduling environments
    Chapter 14, “Defining Scheduling Environments,” on page 125.
Chapter 5. Defining Service Policies

A service policy is a named collection of service class and resource group specification overrides. When a policy is put into effect, the overrides are merged with the service class and resource group specifications in the service definition. A policy override is a way to change a goal or resource group capacity without having to redefine all of your service classes and resource groups.

See “Summary of Service Definition and Service Policy Concepts” on page 13 for an overview of the relationship between a service definition and a service policy, and Figure 8 on page 15 for a visual overview of how service policy overrides work. Note that in an ideal scenario, you would only have to define your service definition once. As part of that service definition, you would predefine multiple policies to meet varying performance goals or business needs. Once the service definition is installed, you would then activate one policy at a time, and then, when appropriate, switch to another. Note that you must define at least one service policy, and you can define up to 99.

When you are creating your service definition, you may choose to define one empty “default” policy with no overrides at all. Next, create your workloads and service classes. Then determine how and when your service classes may have different goals at different times. Define additional policies with the appropriate overrides for these time periods.

### Defining Service Policies

<table>
<thead>
<tr>
<th>Name</th>
<th>Service policy name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Description of the service policy</td>
</tr>
<tr>
<td>Policy Override</td>
<td>Changing a service class goal or resource group</td>
</tr>
</tbody>
</table>

**Name (required)**

Eight characters identifying the service policy. Every service policy name must be unique in a service definition. The service policy is activated by name in one of the following ways:

- An operator command from the operator console.
- A utility function from the workload management ISPF application.

You can display the name of the active service policy with an operator command, or by viewing a performance monitor, such as RMF.

**Description (optional)**

An area of 32 characters describing the service policy. The descriptive text is available to performance monitors for reporting.

**Policy Override (optional)**

A way to change a performance goal, a service class-resource group assignment, or a resource group capacity for a service policy. For more information about defining policy overrides, see “Using Policy Overrides” on page 36.
Examples of Service Policies

- Daytime policy
  Name = DAYTIME
  Description = Policy from 9:00 am to 5:00 pm
- Policy for national holidays
  Name = HOLIDAY
  Description = Policy for Arbor day
- Weekend policy
  Name = WEEKEND
  Description = Policy for Sat and Sun

Using Policy Overrides

Once you have defined your service classes, you can determine whether any of your service class goals change at different times. If they do, you can define a policy override. With an override, you can change one or more of the following for a service policy:
- A goal for a service class period
- Number and duration of periods
- A service class - resource group assignment
- Resource group attributes
- CPU protection.

Example 1: Policy Overrides

In this example, the service class BATPIG is in the BATCH workload. It is associated with the resource group LIMIT. Suppose the LIMIT resource group is assigned some maximum capacity. BATPIG is assigned a discretionary goal. Since it is a discretionary goal, it does not have an assigned importance. In the Weekend policy, however, both the goal and the resource group association is overridden. The resource group association is overridden, so that in the Weekend policy, BATPIG is not assigned to a resource group. It is instead assigned a response time goal of 1 hour, with an importance of 5.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy: Standard</td>
<td>Policy: Weekend (overrides)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service Class..... BATPIG</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description........ All batch CPU hogs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Workload....... BATCH</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Resource Group. LIMIT</td>
<td>LIMIT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Period</td>
<td>Goal.... discretionary</td>
<td>1 hour AVG</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Import.. n/a</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>
**Example 2: Policy Overrides**

In this example, the CICSHIGH service class is in the CICS workload. In the Standard policy, it is assigned a response time goal of 1 second average, with an importance of 1. It is not assigned a resource group, because there is no business need to limit or guarantee capacity. For the weekend policy, however, it has an overridden goal. Because of a contract with CICS users, the agreed response time for weekends is 2 seconds average, with an importance of 2.

<table>
<thead>
<tr>
<th>Service Class</th>
<th>Description</th>
<th>Workload</th>
<th>Resource Group</th>
<th>Period</th>
<th>Goal</th>
<th>Import</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICSHIGH</td>
<td>All short CICS transactions</td>
<td>CICS</td>
<td></td>
<td>1</td>
<td>1 sec avg</td>
<td>1</td>
</tr>
</tbody>
</table>

**Example 3: Policy Overrides**

In this example, the resource group DEPT58 is associated with 3 service classes: 58CICS, 58TSO, and 58BATCH. Since the department is willing to pay for more capacity on the weekends, the minimum is overridden for the weekend policy. It is 1500 CPU service units per second. So in the weekend policy, the service classes 58CICS, 58TSO, and 58BATCH have a minimum of 1500 CPU service units guaranteed.

<table>
<thead>
<tr>
<th>Resource Group</th>
<th>Description</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEPT58</td>
<td>Contracted capacity for dept. 58</td>
<td>1000</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Service Class</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>58CICS</td>
<td>1500</td>
<td></td>
</tr>
<tr>
<td>58TSO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>58BATCH</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chapter 6. Defining Workloads

A workload is a named collection of work to be reported as a unit. You can arrange workloads by subsystem (CICS, IMS), by major application (Production, Batch, Office) or by line of business (ATM, Inventory, department). Logically, a workload is a collection of service classes.

### Defining Workloads

<table>
<thead>
<tr>
<th>Name</th>
<th>Workload name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Description of workload</td>
</tr>
</tbody>
</table>

**Name (required)**

Eight characters identifying the workload. Every workload name must be unique for all defined workloads in a service definition.

**Description (optional)**

An area of 32 characters describing the workload. The descriptive text is available to performance monitors for reporting.

---

**Defining a Departmental Workload**

In order to set up a departmental workload that crosses subsystem boundaries, you must keep in mind how you can assign the work to service classes. You should review Chapter 9, “Defining Classification Rules,” on page 57 and find out the best way you can assign your work to service classes. The following table shows examples of what would be required for some of the subsystems:

- **TSO**
  - You must set up TSO user IDs or account numbers according to department structure, so that the user IDs correspond to a specific department.

- **JES**
  - You must have unique batch classes or account numbers by department.

- **CICS**
  - You must have unique CICS regions for each department.

- **IMS**
  - You must have a separate IMS/VS resource lock manager (IRLM), IMS control region, and IMS message processing region (MPR) for each workload.

For more information, see Chapter 9, “Defining Classification Rules,” on page 57.
Examples of Workloads

- By subsystem
  
  **Name**  IMS
  
  **Description**
  All work in classes IMS1 IMSA IMSS TIMS IMSV

- By department/location
  
  **Name**  DEVELOP
  
  **Description**
  All work in classes STSO IMSB and LINKA
Chapter 7. Defining Resource Groups

A resource group is an amount of processor capacity. It is optional. Unless you have some special need to limit or protect processor capacity for a group of work, you should skip defining resource groups and let workload management manage all of the processor resource to meet performance goals. You use a resource group to:

- Limit the amount of processing capacity available to one or more service classes
- Set a minimum processing capacity for one or more service classes in the event that the work is not achieving its goals
- Define a minimum and maximum amount of capacity sysplex-wide, or on system-level

You can specify a minimum and maximum amount of capacity to a resource group. You can assign only one resource group to a service class. You can assign multiple service classes to the same resource group. You can define up to 32 resource groups per service definition.

Keep in mind your service class goals when you assign a service class to a resource group. Given the combination of the goals, the importance level, and the resource capacity, some goals may not be achievable when capacity is restricted.

If work in a resource group is consuming more resources than the specified maximum capacity, the system caps the associated work accordingly to slow down the rate of resource consumption. The system may use several mechanisms to slow down the rate of resource consumption, including swapping the address spaces, changing its dispatching priority, and capping the amount of service that can be consumed. Reporting information reflects that the service class may not be achieving its goals because of the resource group capping.

By setting a minimum processing capacity, you create an overriding mechanism to circumvent the normal rules of importance. If the work in a resource group is not meeting its goals, then workload management will attempt to provide the defined minimum amount of CPU resource to that resource group.

<table>
<thead>
<tr>
<th>Defining Resource Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
</tr>
<tr>
<td><strong>Description</strong></td>
</tr>
<tr>
<td><strong>Resource Group Type</strong></td>
</tr>
<tr>
<td><strong>Capacity Maximum</strong></td>
</tr>
<tr>
<td><strong>Capacity Minimum</strong></td>
</tr>
</tbody>
</table>
Name
Eight characters that identify the name of the resource group. Each resource
group must be unique within a service definition.

Description
Up to 32 characters that describe the resource group.

Resource Group Type
Resource groups allow to define a guaranteed maximum and minimum CPU
consumption for work on the sysplex and on each individual member of the
sysplex. This allows to:
• Prioritize work on a system-level basis
• Control the minimum and maximum resource consumption

The following types of resource groups are valid:

Resource Group Type 1
The capacity is specified in unweighted CPU service units per second, the
value must be between 0 and 99999999.

Minimum and maximum capacity applies sysplex-wide, that is, WLM
ensures that the limits are met within the sysplex.

The table in Appendix B, “CPU Capacity Table,” on page 241 shows the
service units per second by CPU model.

Resource Group Type 2
The capacity is specified as a percentage of the LPAR share, the value must
be between 0 and 99, and the sum of all minimum LPAR share percentages
for all resource groups of this type should not exceed 99.

Minimum and maximum capacity has a system scope, that is, WLM
ensures that the limits are met on each system within the sysplex. Refer to
“Calculating an LPAR Share - Example 1” on page 43 for a scenario
showing how to calculate an LPAR share when using resource group type
2.

Resource Group Type 3
The capacity is specified as a number of general purpose processors (CPs),
a number of 100 represents the capacity of 1 CP. The number should be
between 0 and 999999.

Minimum and maximum capacity has a system scope, that is WLM
ensures that the limits are met on each system within the sysplex.

Capacity
Identifies the amount of available capacity you want workload management to
allocate to the resource group. Capacity includes cycles in both TCB and SRB
mode. The table in Appendix B, “CPU Capacity Table,” on page 241 shows the
service units per second by CPU model. Resource group minimum can equal
resource group maximum.

Maximum
CPU service that this resource group may use. Maximum specified for this
resource group applies to all service classes in that resource group
combined. Maximum is enforced. There is no default maximum value.

Minimum
CPU service that should be available for this resource group when work in
the group is missing its goals. The default is 0. If a resource group is not
meeting its minimum capacity and work in that resource group is missing
its goal, workload management will attempt to give CPU resource to that work, even if the action causes more important work (outside the resource group) to miss its goal. If there is discretionary work in a resource group that is not meeting its minimum capacity, WLM will attempt to give the discretionary work more CPU resource if that action does not cause other work to miss its goal.

The minimum capacity setting has no effect when work in a resource group is meeting its goals.

**Note:** You cannot assign a resource group to service classes representing transaction-oriented work, such as CICS or IMS transactions. The ISPF application notifies you with an error message if you attempt to define one. If you want to assign a minimum or a maximum capacity to CICS or IMS work, you can do so by assigning a resource group to their regions. For example, suppose you have three service classes representing your CICS works: CICSTRN, CICSAORS, and CICSTORS. CICSTRN represents all of your online CICS transactions, and has one period with a short response time goal. CICSAORS and CICSTORS represent all of your CICS AORs and TORs, respectively, that process the online transactions. To assign a maximum capacity to your CICSTRN work, define a resource group, and assign it to the regions. So you assign the resource group to the CICSAORS and CICSTORS service classes.

### Calculating an LPAR Share - Example 1

The following example illustrates how the capping works for a resource group type 2 and how to calculate this. For this resource group the minimum and maximum capacity is defined as a percentage of the share for the logical partition:

**LPAR share** The LPAR share is defined as the percentage of the weight definition for the logical partition to the sum of all weights for all active partitions on the CEC.

In this example a resource group type 2 ELPMAX is defined to cap CPU-intensive work on the systems in the sysplex environment. The aim is to limit the consumption of the work to 60% of the LPAR share. The sysplex consists of 2 systems: WLM1 and WLM2 which share the CEC with other VM and MVS systems:

<table>
<thead>
<tr>
<th>Table 4. LPAR Configuration - Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Partition</strong></td>
</tr>
<tr>
<td>WLM1</td>
</tr>
<tr>
<td>WLM2</td>
</tr>
<tr>
<td>VMA</td>
</tr>
<tr>
<td>MVSA</td>
</tr>
</tbody>
</table>

The CEC is a zSeries® 990, Model 306. For the value service units per second (SU/s) of 18626.3, refer to Appendix B, “CPU Capacity Table,” on page 241. With 6 processors the total SU/s is 111758. This value is called CEC-capacity. Based on the CEC-capacity value, it is possible to calculate the LPAR share capacity, which is the CEC-capacity multiplied with the LPAR share of each system in the sysplex. For
system WLM1 the LPAR share is 7.8%. The LPAR share capacity for WLM1 results to 8717 SU/s. For ELPMAX a maximum limit of 60% is defined. On WLM1 this results to 5230 SU/s.

Table 5. ELPMAX in Sysplex WLMPLEX- Example

<table>
<thead>
<tr>
<th>Partition</th>
<th>LPAR Share</th>
<th>Share Capacity (SU/s)</th>
<th>RG Limit: 60%</th>
<th>Logical Processors (LCPs)</th>
<th>SU/s based on LCPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>WLM1</td>
<td>7.8%</td>
<td>8717</td>
<td>5230 SU/s</td>
<td>2</td>
<td>37252.6</td>
</tr>
<tr>
<td>WLM2</td>
<td>13.2%</td>
<td>14752</td>
<td>8851 SU/s</td>
<td>2</td>
<td>37252.6</td>
</tr>
</tbody>
</table>

Note that ELPMAX is still defined sysplex-wide but the capacity definition depends on the logical and physical configuration of the systems. On WLM1 work in ELPMAX is entitled to consume 5230 SU/s and on WLM2 to 8851 SU/s. Furthermore, the following two factors are also important for the entitled capacity:

- The SU/s based on the logical processor configuration of the LPAR. If this SU/s for the logical processor configuration is smaller than the share of the logical partition, we will calculate the entitlement for ELPMAX based on the logical processor configuration. In our example there are 2 logical processors defined per partition. This results to 37252.6 SU/s and this value is bigger than the SU/s based on the LPAR share.
- If there is a defined capacity and the partition is capped to that defined capacity, the entitlement for ELPMAX is based on that defined capacity. In the current example it is assumed that there is no defined capacity.

Figure 14 shows how work is capped in resource group ELPMAX on system WLM1. For this reason we associate a service class with ELPMAX and submit work which consumes a lot of CPU service:

Figure 14. Resource Group Overview - Example

It can be observed that the system needs a short period to apply the correct number of capping slices for the work units associated with the resource group.
After this ramp up period, the resource group is capped slightly below the entitled capacity. The reason for that is, that the work is being capped by slices which depend on the capacity of the partition. Therefore, the capacity used can never match the entitlement exactly.

The following shows the complete calculations for applying and interpreting the results of a resource group type 2:

\[
\begin{align*}
R[SUs] & = \text{Softcap} \cdot RG_{\text{Limit}[]} \quad \text{if softcap < ShareCapacity or LCPCapacity} \\
& = \text{LCPCapacity} \cdot RG_{\text{Limit}[]} \quad \text{if LCPCapacity < ShareCapacity or softcap} \\
& = \text{ShareCapacity} \cdot RG_{\text{Limit}[]} \quad \text{if ShareCapacity < LCPCapacity or softcap}
\end{align*}
\]

\[
\begin{align*}
\text{ShareCapacity} & = \text{CECCapacity} \cdot \text{LPARShare} \\
\text{CECCapacity} & = \text{Capacity based on shared physical processors for the CEC} \\
\text{LPARShare} & = \frac{\text{Weight(Current Partition)}}{\sum \text{Weight(i)}}
\end{align*}
\]

\[
\text{LCPCapacity} = \text{Capacity based on shared processors available on the LPAR}
\]

Figure 15. Working With A Resource Type 2 - Sample Calculation

\(R[SUs]\) is the result entitlement for the resource group which depends on the LPAR share, LCP capacity and whether there is a defined capacity which is active.

### Specifying the Capacity as a Number of CPs - Example 2

For resource group type 3, the capacity is defined as a number of general purpose processors (CPs). The following example illustrates how the capping works for a resource group type 3 and how to calculate this. This means, finding out to how many service units per second (SU/s) the defined capacity corresponds.

The CEC is a zSeries 990, Model 316. The WLM service definition specifies a resource group RGT3 (= type 3) with a maximum/minimum value of 250. For resource group type 3 a number of 100 represents the capacity of 1 CP. The LPAR has 6 processors assigned on the hardware console and none of them have been varied offline from the MVS console. To calculate the capping value, do the following:

1. For the correct SU/s value for the resource group, refer to Appendix B, “CPU Capacity Table,” on page 241. In the table for z990s, find the row that represents the model with the number of online processors that your LPAR has. In the current example, this is Model 306, because your LPAR has 6 online processors.

   **Note:** The higher number in this row compared to the value for Model 316 accounts for the lower MP factor that your LPAR has because it has only 6 processors.

2. So, the value you pick is 18 626.3.
3. Multiply 18 626.3 by 250/100.
4. The result is: on that system, the maximum capacity for RGT3 is 46 565 SU/s.
The calculation is done individually for each LPAR. If the LPARs in the sysplex have a different number of processors assigned to them - one can still calculate the correct capping value. This means, the same resource group may represent different SU/s on different LPARs.

An advantage of using resource group type 3 is that it dynamically adjusts to the processor capacity when the work is run on another hardware.
Chapter 8. Defining Service Classes and Performance Goals

A service class is a named group of work within a workload with the following similar performance characteristics:

- Performance goals
- Resource requirements
- Business importance to the installation

Workload management manages a service class period as a single entity when allocating resources to meet performance goals. A service class can be associated with only one workload. You can define up to 100 service classes.

You can assign the following kinds of performance goals to service classes: average response time, response time with percentile, velocity, and discretionary. You assign an importance level to the performance goal. Importance indicates how vital it is to the installation that the performance goal be met relative to other goals.

Because some work has variable resource requirements, workload management provides performance periods where you specify a series of varying goals and importances. You can define up to eight performance periods for each service class. You can also assign a service class to a resource group if its CPU service must be either protected or limited.

This information explains the parts of a service class, how to define performance goals, and how to use performance periods.
Defining Service Classes and Performance Goals

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Service class name</td>
</tr>
<tr>
<td>Description</td>
<td>Service class description</td>
</tr>
<tr>
<td>Workload</td>
<td>The name of the workload associated with this service class.</td>
</tr>
<tr>
<td>Resource Group</td>
<td>The name of the resource group associated with the work in this service class.</td>
</tr>
<tr>
<td>Performance Period</td>
<td>One goal per period</td>
</tr>
<tr>
<td>Duration</td>
<td>Number of service units for this performance period.</td>
</tr>
<tr>
<td>Average Response Time</td>
<td>Average response time for transactions completing within the period in terms of hours, minutes, and seconds. Decimal points are accepted. Response time varies from 15 milliseconds to 24 hours.</td>
</tr>
<tr>
<td>Response Time and Percentile</td>
<td>A percentile of work to be completed in the specified amount of time. Percentile boundaries vary from 1 to 99. Amount of time is in hours, minutes, or seconds. Decimal points are accepted. Response time ranges from 15 milliseconds to 24 hours.</td>
</tr>
<tr>
<td>Velocity</td>
<td>Measure of how fast work should run when ready, without being delayed for WLM-managed resources. Velocity ranges from 1 to 99.</td>
</tr>
<tr>
<td>Discretionary</td>
<td>Workload management defined goal. Work is run as system resources are available.</td>
</tr>
<tr>
<td>Importance</td>
<td>How important it is to the installation that the goal be achieved.</td>
</tr>
<tr>
<td>CPU Protection</td>
<td>Whether long-term CPU protection should be assigned to this service class.</td>
</tr>
</tbody>
</table>

Name (required)
Eight characters describing the service class. Service class names must be unique within a service definition.

Description (optional)
An area of 32 characters describing the service class. The descriptive text is available to performance monitors for reporting.

Workload (required)
The name of the workload associated with the service class. You can associate only one workload per service class in a service definition. The workload must have been previously defined.

Resource Group (optional)
The resource group name associated with this service class. You can assign
only one resource group per service class in a service policy. You can override
the resource group assigned to a service class in each service policy. For more
information about resource groups, see Chapter 7, “Defining Resource Groups,”
on page 41.

Performance Period
A performance goal, importance, and duration for a service class. You set up
multiple performance periods for work that has changing performance
requirements as work consumes more and more resources. You can specify up
to eight performance periods.

Duration
Specifies the length of the period in service units. For a definition of service
units, see Chapter 11, “Defining Service Coefficients and Options,” on page 93.
If the work included in this service class period does not complete when the
number of service units have been used, the work moves into the next
performance period. You do not specify a duration on the last defined period.

Response Time
The expected amount of time required to complete the work submitted under
the service class, in milliseconds, seconds, minutes and hours. Specify either an
average response time, or response time with a percentile. Percentile is the
percentage of work in that period that should complete within the response
time. Percentile must be a whole number. You must specify a system response
time goal, not “end-to-end”. That is, workload management does not control
all aspects of system performance, so response time scope is confined to the
time workload management has control of the work. This time includes the
time the work is using or waiting for CPU, storage, or I/O service.

Note: Workload management does not delay work, or limit it, to achieve the
response time goal when extra processing capacity exists.

Velocity
A measure of how fast work should run when ready, without being delayed
for WLM-managed resources. Velocity is a percentage from 1 to 99. See
“Velocity Formula” on page 50 for a description of the calculations needed to
determine velocity.

Discretionary
Workload management defined goal. Associate this goal with work for which
you do not have a specific performance goal. Work with a discretionary goal is
run when excess resources are available.

Importance
The relative importance of the service class goal. Importance is a reflection of
how important it is that the service class goal be achieved. Workload
management uses importance only when work is not meeting its goal.
Importance indicates the order in which work should receive resources when
work is not achieving its goal. Importance is required for all goal types except
discretionary. Importance applies on a performance period level and you can
change importance from period to period. Importance is in five levels: 1 to 5, 1
being the highest importance.

CPU Protection
By specifying YES in the “CPU Critical” field when defining a service class,
you ensure that work of lower importance will always have a lower dispatch
priority. See Chapter 12, “Defining Special Protection Options for Critical
Work,” on page 99 for more information.
Velocity Formula

The formula for velocity is:

\[
\text{velocity} = \frac{\text{using samples}}{\text{using samples} + \text{delay samples}} \times 100
\]

where:

**using samples**
- The number of samples of work using the processor
- The number of samples of work using non-paging DASD I/O resources (in a state of device connect).\(^1\)

The I/O samples are derived from actual time measurements.

**delay samples**
- The number of samples of work delayed for the processor
- The number of samples of work delayed for storage
- The number of samples of work delayed for non-paging DASD I/O.\(^1\)

Work delayed for storage includes:
- Paging
- Swapping
- Multiprogramming level (MPL)
- Server address space creation delays
- Initiation delays for batch jobs in WLM-managed job classes.

MPL is the SRM-controlled function that adjusts the number of address spaces allowed to be in central storage and ready to be dispatched.

I/O delays include:
- IOS queue
- Subchannel pending
- Control unit queue delays

The samples for subchannel pending and control unit queue delay are derived from actual time measurements.

---

**Defining Performance Goals**

This section explains how to define performance goals in your service definition. If you have an SLA today, you should consider a few things:

- Does it contain end-to-end response time?
  
  If it does, then you need to keep in mind that workload management processes towards system response times, and make the adjustment when you define the performance goal. Section "Determining System Response Time Goals" on page 51 explains how you can determine the system response times of work.

- For what type of workloads do you need a different goal?

---

\(^1\) If I/O priority management is off, these samples are not included.
You may have some throughput type goals, which you need to convert into either response time goals, or velocity goals.

**Determining System Response Time Goals**

Goal mode introduces several changes in the definition of a work request's response time. The changes more accurately reflect end-user expectations of response time.

The number of batch transactions equals number of jobs. Defining a response time goal may not be appropriate for some types of batch work, such as jobs with very long execution times. Work that is appropriate for a response time goal should have at least three transaction completions per 20 minutes of elapsed time. If there are too few completions, use a velocity or discretionary goal.

TYPRUN=HOLD and TYPRUN=JCLHOLD times are *not* included in batch response times.
Examples

Examples of Service Classes with Response Time Goals

- A service class representing TSO/E work with multiple periods.

<table>
<thead>
<tr>
<th>Service Class</th>
<th>TSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period 1</td>
<td></td>
</tr>
<tr>
<td>Response Time</td>
<td>85% 0.5 second</td>
</tr>
<tr>
<td>Importance</td>
<td>1</td>
</tr>
<tr>
<td>Duration</td>
<td>400 Service Units</td>
</tr>
<tr>
<td>Period 2</td>
<td></td>
</tr>
<tr>
<td>Response Time</td>
<td>80% 1 second</td>
</tr>
<tr>
<td>Importance</td>
<td>3</td>
</tr>
<tr>
<td>Duration</td>
<td>1000 Service Units</td>
</tr>
<tr>
<td>Period 3</td>
<td></td>
</tr>
<tr>
<td>Response Time</td>
<td>60% 15 second</td>
</tr>
<tr>
<td>Importance</td>
<td>4</td>
</tr>
</tbody>
</table>

Note that the percentile in period 1 and 2 refer to the transactions ending in each period, not the total TSO/E transactions.

- A service class representing CICS transactions.

<table>
<thead>
<tr>
<th>Service Class</th>
<th>CICSHOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period 1</td>
<td></td>
</tr>
<tr>
<td>Response Time</td>
<td>0.5 second AVG</td>
</tr>
<tr>
<td>Importance</td>
<td>1</td>
</tr>
</tbody>
</table>

- A service class representing IMS transactions.

<table>
<thead>
<tr>
<th>Service Class</th>
<th>IMSCAT1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response Time</td>
<td>95% .3 Second</td>
</tr>
<tr>
<td>Importance</td>
<td>1</td>
</tr>
</tbody>
</table>

- A service class representing IMS transactions.

<table>
<thead>
<tr>
<th>Service Class</th>
<th>OIMSCAT3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response Time</td>
<td>5 sec AVG</td>
</tr>
<tr>
<td>Importance</td>
<td>3</td>
</tr>
</tbody>
</table>

Defining Velocity Goals

This section describes where to find information to set a velocity goal, and what kind of work is appropriate for velocity goals. Velocity goals define the acceptable amount of delay for work when work is ready to run. Velocity goals are intended for subsystems which use address spaces or enclaves to represent individual work requests. Velocity goals are not supported for work in the IMS and CICS subsystem work environments because velocity data is accounted to the region, not to the individual transaction. Velocity is a goal to consider for long-running jobs.
For a service class with multiple periods, you cannot switch from a velocity goal in one period to a response time goal in a later period. See "Subsystem Support for Goal Types and Multiple Periods" on page 18 for a list of subsystems for which you can specify multiple periods.

**Adjusting Velocity Goals Based on Samples Included in Velocity Calculation**

You can adjust your velocity goals based on whether or not the following samples are to be included in the velocity calculation:

- I/O samples (included in the velocity calculation if I/O priority management is turned on)
- Initiation delay samples (included in the velocity calculation if you have WLM-managed batch initiators).

In the RMF Monitor I workload activity report, there are two fields, **I/O PRTY** and **INIT MGMT**, which indicate the following:

**I/O PRTY**

If you have I/O priority management turned off, then the **I/O PRTY** value shows you what your velocity would be if you were to turn I/O priority management on.

**INIT MGMT**

If you are not currently using WLM-managed batch initiators, then the **INIT MGMT** value shows you what your velocity would be if you turned over control of all batch initiators in this service class to WLM.

**Note:** For both the **I/O PRTY** and **INIT MGMT** fields, it is assumed that the other setting is unchanged. For example, the **INIT MGMT** field assumes that your current I/O priority management setting remains the same.

These fields may help you to adjust a current velocity goal in anticipation of including these samples.

**Using Velocity Goals for Started Tasks**

Velocity goals are the most appropriate goal for started tasks and long running work. Instead of figuring out a specific velocity goal for your started tasks, you should divide your started tasks into a high, a medium, and a low importance service class, and define a velocity that suffices for each category.

You can also take advantage of the system supplied service classes for started tasks: SYSTEM and SYSSTC. Workload manager recognizes special system address spaces (like GRS, SMF, CATALOG, MASTER, RASP, XCFCAS, SMXC, CONSOLE, IOSAS, WLM), puts them into the SYSTEM service class, and treats them accordingly. Address spaces in the SYSSTC service class are kept at a very high dispatching priority.

**Note:** You can also assign address spaces to the SYSTEM and SYSSTC service classes as part of your work classification rules. See "System-Provided Service Classes" on page 82.

For information about how to define service classes and associated classification rules for started tasks, see "Using the System-Supplied Service Classes" on page 87.
Velocity is also appropriate for the “server” started tasks, that is, the address spaces that do work on behalf of a transaction manager or resource manager, such as CICS AOR, or an IMS control region. Since the server address spaces are processing work that also has an assigned performance goal, the velocity goal that you assign to servers applies only during address space startup. Then workload management manages resources to meet the goals defined for the work the servers are processing, and not towards the goals defined for the servers.

If you have a version of a work manager such as CICS and IMS that does not support workload management, you cannot define a goal to the work manager’s transactions, but you can define a velocity goal for its server address spaces.

**Using Discretionary Goals**

With discretionary goals, workload management decides how best to run this work. Since workload management’s prime responsibility is matching resources to work, discretionary goals are used best for the work for which you do not have a specific performance goal. For a service class with multiple performance periods, you can specify discretionary only as the goal in the last performance period.

Discretionary work is run using any system resources not required to meet the goals of other work. If certain types of other work are overachieving their goals, that work may be “capped” so that the resources may be diverted to run discretionary work. See “Migration Considerations for Discretionary Goal Management” on page 154 for more information on the types of work that are eligible for resource donation, and how you may want to adjust those goals.

### Examples of Service Classes with Discretionary Goals

- **Discretionary goal as last period goal**

<table>
<thead>
<tr>
<th>Service Class</th>
<th>DEVBATCH</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Period</strong></td>
<td>1</td>
</tr>
<tr>
<td>Response Time</td>
<td>80% 1 minute</td>
</tr>
<tr>
<td>Importance</td>
<td>2</td>
</tr>
<tr>
<td>Duration</td>
<td>2000 Service Units</td>
</tr>
<tr>
<td><strong>Period</strong></td>
<td>2</td>
</tr>
<tr>
<td>Response Time</td>
<td>80% 5 minutes</td>
</tr>
<tr>
<td>Importance</td>
<td>3</td>
</tr>
<tr>
<td>Duration</td>
<td>10000 Service Units</td>
</tr>
<tr>
<td><strong>Period</strong></td>
<td>3</td>
</tr>
<tr>
<td>Discretionary</td>
<td></td>
</tr>
</tbody>
</table>

- **Discretionary goal for leftover work**

<table>
<thead>
<tr>
<th>Service Class</th>
<th>ASDBATCH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discretionary</td>
<td></td>
</tr>
</tbody>
</table>
Using Performance Periods

Performance periods are available for work that has variable resource requirements and for which your goals change as the work uses more resources. You specify a goal, an importance, and a duration for a performance period. Duration is the amount of service that period should consume before going on the next goal. Duration is specified in service units. For more information about defining durations, see Chapter 11, “Defining Service Coefficients and Options,” on page 93.

You can define multiple performance periods for work in subsystems which use address spaces or enclaves to represent individual work requests. For a list of subsystems for which you can specify multiple periods, see “Subsystem Support for Goal Types and Multiple Periods” on page 18.

Multiple periods are not supported for work in the IMS and CICS subsystem work environments because service units are accumulated to the address space, not the individual transactions. So, the system cannot track a duration for those transactions.

Multiple periods are also not supported for work in the SYSH subsystem.

Defining Goals Appropriate for Performance Periods

As you go from one performance period to the next, you can change the type of goal. Goals should become less stringent going from one period to the next. A prime example would be changing to a velocity or discretionary type goal in the last period.

Using Importance Levels in Performance Periods

Importance levels should stay the same or decrease as the transactions move from one performance period to the next. Remember that importance applies only if a goal is not being met during the duration of the period.

<table>
<thead>
<tr>
<th>Examples of Multiple Performance Period Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decreasing stringency of goal and decreasing importance from one period to the next</td>
</tr>
<tr>
<td>Service Class = BATCHX</td>
</tr>
<tr>
<td>Period 1</td>
</tr>
<tr>
<td>Velocity = 50</td>
</tr>
<tr>
<td>Importance = 3</td>
</tr>
<tr>
<td>Duration = 2500 SU</td>
</tr>
<tr>
<td>Period 2</td>
</tr>
<tr>
<td>Velocity = 15</td>
</tr>
<tr>
<td>Importance = 5</td>
</tr>
</tbody>
</table>

Chapter 8. Defining Service Classes and Performance Goals
Chapter 9. Defining Classification Rules

Classification rules are the rules you define to categorize work into service classes, and optionally report classes, based on work qualifiers. A work qualifier is what identifies a work request to the system. The first qualifier is the subsystem type that receives the work request.

There is one set of classification rules in the service definition for a sysplex. They are the same regardless of what service policy is in effect; a policy cannot override classification rules. You should define classification rules after you have defined service classes, and ensure that every service class has a corresponding rule.

The full list of work qualifiers and their abbreviations is:

<table>
<thead>
<tr>
<th>Qualifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI</td>
<td>Accounting information</td>
</tr>
<tr>
<td>CI</td>
<td>Correlation information</td>
</tr>
<tr>
<td>CN</td>
<td>Collection name</td>
</tr>
<tr>
<td>CT</td>
<td>Connection type</td>
</tr>
<tr>
<td>ESC</td>
<td>zEnterprise service class name from a Unified Resource Manager performance policy</td>
</tr>
<tr>
<td>LU</td>
<td>Logical Unit name</td>
</tr>
<tr>
<td>NET</td>
<td>Netid</td>
</tr>
<tr>
<td>PC</td>
<td>Process name</td>
</tr>
<tr>
<td>PF</td>
<td>Perform</td>
</tr>
<tr>
<td>PK</td>
<td>Package name</td>
</tr>
<tr>
<td>PN</td>
<td>Plan name</td>
</tr>
<tr>
<td>PR</td>
<td>Procedure name</td>
</tr>
<tr>
<td>PRI</td>
<td>Priority</td>
</tr>
<tr>
<td>PX</td>
<td>Sysplex name</td>
</tr>
<tr>
<td>SE</td>
<td>Scheduling environment name</td>
</tr>
<tr>
<td>SI</td>
<td>Subsystem instance</td>
</tr>
<tr>
<td>SPM</td>
<td>Subsystem parameter</td>
</tr>
<tr>
<td>SSC</td>
<td>Subsystem collection name</td>
</tr>
<tr>
<td>SY</td>
<td>System name</td>
</tr>
<tr>
<td>TC</td>
<td>Transaction class/job class</td>
</tr>
<tr>
<td>TN</td>
<td>Transaction name/job name</td>
</tr>
<tr>
<td>UI</td>
<td>Userid</td>
</tr>
</tbody>
</table>

Notes:
1. Not all work qualifiers are valid for every subsystem type; they are subsystem dependent. For details on which qualifiers are valid for which subsystems, see Table 9 on page 74.
2. For many of these qualifiers, you can specify classification groups by adding a G to the type abbreviation. (For example, a transaction name group would be TNG.) See “Using Groups” on page 86 for more information.

A singular classification rule is a work qualifier and an associated service class or report class. The following example is a classification rule:
Example of a Classification Rule

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Fold qualifier names?</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMS</td>
<td>Y (Y or N)</td>
<td>IMS medium interactive</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Action Type</th>
<th>Name Start</th>
<th>Service Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>-----</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

**Defaults:** IMSMED

**Note:** The “Fold qualifier names” option, set to the default Y, means that the qualifier names will be folded to uppercase as soon as you type them and press Enter. If you set this option to N, the qualifier names will remain in the case they are typed in. Leave this option set to Y unless you know that you need mixed case qualifier names in your classification rules.

This example shows that all work coming into any IMS subsystem is associated with service class IMSMED. Service class IMSMED is the default service class for the IMS subsystem type. You can also assign a default report class to a subsystem type.

Since you might not want all work coming into a subsystem assigned to the same service class, or the same report class, you can specify multiple classification rules.

**Figure 16** shows two classification rules. In the example, the incoming work request has work qualifiers of subsystem type, job name, job class, accounting information, and user ID.

---

In the example, the service administrator set up classification rules to assign all work coming into JES into service class BATCHA, unless the work had a user ID of BOND, in which case, it should be assigned to service class BATCHB. For JES classification, you do not need to specify JES2 or JES3.

For example, if you want all CICS work to go into service class CICSB except the following:

- You want work originating from LU name LONDON to run in service class CICSD
- You want work originating from LU name PARIS to run in service class CICSA, unless:
• The work is from the PAYROLL application, in which case you want it to go into service class CICSC.

You could specify the following classification rules:

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>LU Name</th>
<th>Transaction Name</th>
<th>Service Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICS</td>
<td>LONDON</td>
<td></td>
<td>CICSB</td>
</tr>
<tr>
<td></td>
<td>PARIS</td>
<td>PAYROLL</td>
<td>CICSA</td>
</tr>
</tbody>
</table>

The previous example has two classification rules with level 1 qualifiers: LU name LONDON and LU name PARIS. Under PARIS there is a level 2 qualifier with transaction name PAYROLL. The PAYROLL qualifier applies only to transactions associated with the level 1 qualifier of PARIS.

In this example, if a work request comes in from an LU other than London or Paris, then it is assigned CICSB. If another work request comes in from Paris, and is from the payroll application, it is assigned to the service class CICSC. If a work request is from the payroll application, but came in from a system in London, then it is assigned to the service class CICSD.

The order of the nesting and the order of the level 1 qualifiers, determine the hierarchy of the classification rules. The application supports eight characters for each rule. For more information about defining the hierarchy of the classification rules, see "Defining the Order of Classification Rules" on page 78.

Defining Classification Rules for Each Subsystem

Work qualifiers depend on the subsystem that first receives the work request. When you are defining the rules, start with the service classes you have defined, and look at the type of work they represent. Determine which subsystem type or types process the work in each service class. Then understand which work qualifiers your installation could use for setting up the rules. It may be that your installation follows certain naming conventions for the qualifiers for accounting purposes. These naming conventions could help you to filter work into service classes. Also, understand which work qualifiers are available for each subsystem type. You can then decide which qualifiers you can use for each service class.

The following table shows the IBM-supplied subsystem types that workload management supports, the kind of work they run, whether they use address space-oriented transactions or enclaves (see special note for CICS and IMS), and where to go for more information. (Unless otherwise noted, look for "Workload Manager" in each book.) A comparison of the various transaction types is shown in Table 8 on page 64.

Note: Enclaves are transactions that can span multiple dispatchable units in one or more address spaces. See z/OS MVS Programming: Workload Management Services for more information on enclaves.
### Table 6. IBM-Defined Subsystem Types

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Work Description</th>
<th>Enclave, Address Space, or LPAR</th>
<th>For More Information, See:</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASCH</td>
<td>The work requests include all APPC transaction programs scheduled by the IBM-supplied APPC/MVS transaction scheduler.</td>
<td>Address Space</td>
<td><em>z/OS MVS Planning: APPC/MVS Management</em></td>
</tr>
<tr>
<td>CB</td>
<td>The work requests include all WebSphere Application Server client object method requests.</td>
<td>Enclave</td>
<td><em>The online information included with the WebSphere Application Server system management user interface</em></td>
</tr>
</tbody>
</table>
| CICS           | The work requests include all transactions processed by CICS Version 4, and higher. | See note. | *CICS/ESA Performance Guide*  
*CICS/ESA Dynamic Transaction Routing in a CICSplex* |
| DB2            | The work requests include only the queries that DB2 has created by splitting a single, larger query and distributed to remote systems in a sysplex. The local piece of a split query, and any other DB2 work, is classified according to the subsystem type of the originator of the request (for example, DDF, TSO, or JES). | Enclave | *DB2 Data Sharing: Planning and Administration* |
| DDF            | The work requests include all DB2 distributed data facility (DB2 Version 4 and higher) work requests. | Enclave | *DB2 Data Sharing: Planning and Administration* |
| EWLM           | Work requests include DB2 distributed data facility (DDF) requests that originate from an ensemble, through virtual servers that are classified within a Unified Resource Manager performance policy. | Enclave | *“Defining z/OS WLM Classification Rules for Performance Policies from the Unified Resource Manager” on page 76* |
| IMS            | The work requests include all messages processed by IMS Version 5 and higher. | See note. | *IMS/ESA Administration Guide: System* |
| IWEB           | The work requests include all requests from the world-wide-web being serviced by the Internet Connection Server (ICS), Domino Go Webserver, or IBM http Server for z/OS. These requests also include those handled by the Secure Sockets Layer (SSL). This also includes transactions handled by the Fast Response Cache Accelerator. | Enclave | *z/OS HTTP Server Planning, Installing, and Using* |
| JES            | The work requests include all jobs that JES2 or JES3 initiates. | Address Space | *z/OS JES2 Initialization and Tuning Guide*  
*z/OS JES3 Initialization and Tuning Guide* |
<p>| LDAP           | The work requests include all work processed by the z/OS LDAP server | Enclave | <em>z/OS IBM Tivoli Directory Server Administration and Use for z/OS</em> |
| LSFM           | The work requests include all work from LAN Server for MVS. | Enclave | |
| MQ             | The work requests include MQSeries Workflow work such as new client server requests, activity executions, activity responses, and subprocess requests. | Enclave | |</p>
<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Work Description</th>
<th>Enclave, Address Space, or LPAR</th>
<th>For More Information, See:</th>
</tr>
</thead>
</table>
| NETV          | The work requests include NetView® network management subtasks and system automation (SA) subtasks created by Tivoli® NetView for z/OS. | Enclave | • The Tivoli NetView for z/OS Tuning Guide  
• Tivoli NetView for z/OS Installation  
• Tivoli NetView for z/OS Administration Reference  
• APAR OW54858 |
| OMVS          | The work requests include work processed in z/OS UNIX System Services forked children address spaces. (Work that comes from an enclave is managed to the goals of the originating subsystem.) | Address Space | • z/OS UNIX System Services Planning |
| SOM           | The work requests include all SOM client object class binding requests. | Enclave | • z/OS SOMobjects Configuration and Administration Guide |
| STC           | The work requests include all work initiated by the START and MOUNT commands. STC also includes system component address spaces such as the TRACE and PC/AUTH address spaces. | Address Space | • “Using the System-Supplied Service Classes” on page 87 |
| TCP           | The work requests include work processed by the z/OS Communications Server. | Enclave | • z/OS Communications Server: IP Configuration Guide |
| TSO           | The work requests include all commands issued from foreground TSO sessions. | Address Space | • z/OS MVS Initialization and Tuning Guide  
(Look for the discussion of TSO/E transactions in the “System Resources Manager” information.) |
| SYSH          | Identifies non z/OS partitions (e.g. LINUX partition) in the LPAR cluster which needs to be managed by WLM according to business goals set for the partition. | LPAR | • “Non z/OS Partition CPU Management” on page 27 and  
Chapter 18, “The Intelligent Resource Director,” on page 161 |

**Important Note on CICS and IMS Transactions**

CICS and IMS do not use enclaves, but use a different set of WLM services to provide transaction management.

CICS and IMS transactions can be assigned only response time goals (either percentile or average) within single period service classes. If you do not define any goals at all for CICS or IMS work, then the work will be managed to the velocity goals of the address spaces. Once you have defined a transaction goal for CICS or IMS work, then all subsequent work will be managed to those transaction goals, not to the velocity goals of the address spaces.

For example, you may initially be managing all CICS work to the velocity goals of the CICS address space. If you define a response time goal for a CICS transaction, you will be required to declare a default goal as part of that definition. Now all CICS transactions will be managed to those response time goals, even if they must accept the default.
**Important Note on NETV Subsystem**

Make sure to add the subsystem type NETV to your service definition (option 6 in ISPF application).

Tivoli NetView optionally allows you to let WLM manage NetView subtask performance in relation to other tasks and applications running on the system or sysplex. If enabled, NetView creates enclaves during subtask initialization and calls WLM to classify a subtask to the appropriate service class.

When a user decides to separate the management of NetView's network and system automation (SA) subtasks, NetView creates z/OS enclaves to manage those two sets of subtasks so that users can assign different performance goals to the enclaves. "Network" subtasks include all those not connected with system automation.

These two types of NetView enclaves should be classified to service classes with velocity goals. The goals should have approximately the same velocity value, but the goal assigned to NetView system automation enclaves should be more important than the goal assigned to any NetView network enclaves. There is no need to define a separate service class for NetView, if existing service classes in your service definition satisfy these conditions. For example, if SA z/OS or other system automation is used, a goal of Velocity = 50 and an Importance of 1 could be assigned. For non-system automation NetView subtasks, a goal of Velocity = 30 and an Importance of 2 could be assigned to give preference to the system automation NetView subtasks.

If the NetView WLM support is enabled, the absence of classification rules for subsystem type NETV will result in the NetView enclaves being classified to service class SYSOTHER.

Note that the WLM ISPF application does not validate the classification attributes used in the classification rules for subsystem type NETV.

Table 7 on page 63 shows some selected non-IBM subsystems, with information on how customers can define classification rules for these WLM exploiters. The information here may be incomplete and may not reflect recent changes to the exploiting subsystems. Always check the related subsystem information for the latest updates.
Table 7. Non-IBM Subsystems

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Work Description (with Allowable Work Qualifiers, Goals, and Periods)</th>
<th>For More Information, See:</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP R/3</td>
<td>Each SAP R/3 work process has an associated type, corresponding to the allowable transaction name qualifiers. The ICLI server maps each work process type to one of eight pre-existing enclaves, each with its own performance goal. These enclaves are started at ICLI server startup time, and are deleted when the ICLI server is shut down. Therefore, only velocity goals are appropriate, with single periods.</td>
<td>• SAP R/3 on DB2 for OS/390: Planning Guide</td>
</tr>
</tbody>
</table>

**Allowable Work Qualifiers:**
- Transaction Name — must be one of the following:
  - DIALOG
  - BATCH
  - UPDATE
  - UPDATE2
  - GENERIC
  - SPOOL
  - ENQUEUE
  - UNKNOWN
- Userid — the user ID of the user who started the ICLI server

When defining the classification rules for SAP, use the Userid (UI) qualifier with the user who started the ICLI server, and then the Transaction Name (TN) qualifiers (with the exact names shown) as the subrules. Here is an example:

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>. . . SAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type Name Service Report</td>
<td>DEFAULT: DISCRET ______</td>
</tr>
<tr>
<td>1 UI ICLIRUN</td>
<td>______</td>
</tr>
<tr>
<td>2 TN GENERIC</td>
<td>LOWVEL ______</td>
</tr>
<tr>
<td>2 TN DIALOG</td>
<td>DIALOG ______</td>
</tr>
<tr>
<td>2 TN UPDATE</td>
<td>UPDATE ______</td>
</tr>
<tr>
<td>2 TN UPDATE2</td>
<td>HIGHVEL ______</td>
</tr>
<tr>
<td>2 TN SPOOL</td>
<td>LOWVEL ______</td>
</tr>
<tr>
<td>2 TN BATCH</td>
<td>BATCH ______</td>
</tr>
</tbody>
</table>

**Note:** In this example, ENQUEUE and UNKNOWN work defaults to the DISCRET service class.

If you have a subsystem not included in either of these tables, you should check its documentation for the kind of work requests supported by workload management and the applicable work qualifiers.
The following table summarizes the key differences among the service classes for enclave transactions, address space-oriented transactions, and IMS/CICS transactions.

<table>
<thead>
<tr>
<th>Transaction Type</th>
<th>Allowable Goal Types</th>
<th>Allowable Number of Periods</th>
<th>RMF (or Other Monitor) Reporting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address Space-Oriented</td>
<td>• Response Time</td>
<td>Multiple</td>
<td>• IOC, CPU, MSO, and SRB service consumption reported</td>
</tr>
<tr>
<td></td>
<td>• Execution Velocity</td>
<td></td>
<td>• Execution delays reported</td>
</tr>
<tr>
<td></td>
<td>• Discretionary</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enclave</td>
<td>• Response Time</td>
<td>Multiple</td>
<td>• CPU service consumption reported</td>
</tr>
<tr>
<td></td>
<td>• Execution Velocity</td>
<td></td>
<td>• Execution delays reported</td>
</tr>
<tr>
<td></td>
<td>• Discretionary</td>
<td></td>
<td>• “Served by” reported for enclaves using TCBs</td>
</tr>
<tr>
<td>CICS/IMS</td>
<td>• Response Time</td>
<td>1</td>
<td>• No service consumption reported (reported under regions)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• No execution delays reported (reported under regions)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• “Service Classes Being Served” reported (for service classes assigned to the server address spaces)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• “Response Time Breakdown in Percentage” reported</td>
</tr>
</tbody>
</table>

The ISPF application provides these subsystem types as a selection list on the classification rules panel. You can add any additional subsystem type if it supports workload management on the same panel.

**Defining Work Qualifiers**

The name field for work qualifiers is 8 characters long. You can use nesting for the work qualifiers that run longer than 8 characters. These are accounting information, correlation information, collection name, procedure name, and subsystem parameter. See “Organizing Work for Classification” on page 84 for how to nest using the start position.

You can use masking and wild card notation to group qualifiers that share a common substring. For accounting information and the subsystem parameter, you can use a start position to indicate how far to index into the character string. When no start parameter is specified, WLM matches the name field for work qualifiers according to the number of characters specified.

**Accounting Information**

<table>
<thead>
<tr>
<th>ASCH</th>
<th>The information passed on the JOB statement.</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2</td>
<td>Accounting information is associated with the originator of the query; for example, the accounting information from TSO, JES, or DDF.</td>
</tr>
<tr>
<td>DDF</td>
<td>Accounting information is the value of the DB2 accounting string associated with the DDF server thread.</td>
</tr>
<tr>
<td>JES</td>
<td>The information passed on the JOB statement, not the EXEC statement.</td>
</tr>
</tbody>
</table>

**Note:** WLM/SRM ignores this if the JES2 init deck has a JOBDEF statement with parm ACCTFLD=OPTIONAL specified.
OMVS  Accounting data is normally inherited from the parent process of a 
z/OS UNIX System Services address space. In addition, when a 
daemon creates a process for another user, accounting data is taken 
from the WORKATTR of the RACF® user profile. A user can also 
assign accounting data by setting the _BPX_ACCT_DATA 
environment variable or by passing accounting data on the 
interface to the _spawn service. For more information about z/OS 
UNIX System Services accounting information, see [z/OS UNIX System Services Planning](#).

STC  The information passed on the JOB statement.

TSO  The accounting information specified by the TSO/E user when 
logging on to the system.

Because JCL supports 143 characters in accounting information, and the application 
allows only eight characters per rule, the application allows “nesting” for 
accounting information. See “Using Inheritance in Classification Rules” on page 79 
for more information.

---

**Example of Nesting Accounting Information**

In this example, you can “nest” accounting information (AI) in the 
classification rules.

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>JES (Required)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>All batch rules</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>Service</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AI</td>
<td>43876AAA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>AI</td>
<td>DEPT58*</td>
<td>9</td>
<td>BATHBEST</td>
</tr>
</tbody>
</table>

This example shows the classification rules for the JES subsystem. You can 
classify with more than the allowed 8 characters by nesting accounting 
information. In the example, all work with accounting information 
‘43876AAADEPT58’ starting in position 1 for 14 characters is associated with 
service class BATHBEST.

---

**Collection Name**

**CB**  The logical server group name defined using the WebSphere 
Application Server system management utility. This represents a 
set of WebSphere Application Server objects that are grouped 
together and run in a logical server. For more information, see the 
online information included with the WebSphere Application 
Server system management user interface.

**DB2**  The collection name associated with the originator of the query; for 
example, the collection name from DDF.

**DDF**  The DB2 collection name of the first SQL package accessed by the 
distributed relational database architecture (DRDA®) requestor in 
the work request.

**SOM**  The logical server name defined using the SOM REGIMPL utility 
(defined to REGIMPL as the application alias). This represents a set
of SOM objects that are grouped together and run in a logical server. For more information, see z/OS SOMobjects Configuration and Administration Guide.

**Connection Type**

**DB2** Connection type associated with the originator of the query; for example, the connection type from DDF.

**DDF** The DB2 connection type of the DDF server thread. The thread contains the value 'DIST' indicating it is a server.

**Correlation Information**

**DB2** Correlation ID associated with the originator of the query; for example, the correlation ID from DDF.

**DDF** The DB2 correlation ID of the DDF server thread.

**LU name and Netid**

LU name and netid are used mostly for qualifying CICS, DB2, DDF, and IMS work. If you want to filter on the fully qualified name, you can use the LU name value of up to 8 characters, and then define a sub-rule of netid for up to 8 characters.

**CICS** Only the LU name is available for CICS. It is the 8-byte NETNAME of the principal facility of the transaction instance. For details of the value of this parameter for non-VTAM terminals, and for transaction-routed transactions see CICS/ESA Performance Guide.

**DB2** The LU name and netid associated with the originator of the query; for example, the LU name and netid from DDF. For more information about the format of the LU name, see DB2 SQL Reference.

**DDF** The VTAM LU name and netid of the system that issued the structured query language (SQL) request. For more information about the format of the LU name, see DB2 SQL Reference.

**IMS** If a transaction comes in from an LU 6.2 device, you can specify both the LU name and the netid. Otherwise, you can specify only the 8 byte LU name.

**NETV** The LU name and the netid associated with the NetView subtask.

**Package Name**

**DB2** Package name of the originator of the query; for example, the package name from DDF.

**DDF** The name of the first DB2 package accessed by the DRDA requestor in the work request.

**Perform**

**DB2** The performance group number for the thread associated with the query.

**JES** The performance group number specified using the PERFORM keyword on the JCL JOB statement.

**STC** One of the following:
  * The performance group number specified using the PERFORM keyword on the START command.
• The performance group number specified using the PERFORM keyword on the JCL JOB statement.

**TSO**
The performance group number specified on the logon panel.

**Plan Name**

**DB2**
Plan name associated with the originator of the query; for example, the plan name from DDF.

**DDF**
The DB2 plan name associated with the DB2 server thread. For DB2 private protocol requestors and DB2 Version 3 or higher DRDA requestors, this is the DB2 plan name of the requesting application. For non-DB2 requestors and other DRDA requesters, this is not applicable.

**Priority**

**DB2**
Priority associated with the originator of the query; for example, the priority from a batch job.

**JES**
A value between 0 and 15, the priority associated with the batch job submitted through JES2 or JES3.

**Note:** For JES work, 15 is the highest priority and 0 is the lowest. (Contrast with MQ work, where 0 is the highest priority and 9 is the lowest.)

**MQ**
A value between 0 and 9, the priority associated with the Websphere MQ Workflow message.

**Note:** For MQ work, 0 is the highest priority and 9 is the lowest. (Contrast with JES work, where 15 is the highest priority and 0 is the lowest.)

**NETV**
Indicates a number between 1 and 9 that defines the dispatching priority of this task in relation to other subtasks running in this NetView program. For more information, see *Tivoli NetView Administration Reference*.

When you use priority as a work qualifier, you can use operators such as greater-than ('>') and less-than ('<') to group a range of priorities into one service or report class.

---

**Priority Example**

To put priority 8 and higher work into service class BATCH020, and put all other work into service class BATCH005, you would code the following:

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PRI</td>
<td>&gt;=8</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DEFAULTS: BATCH005</td>
<td>BATCH020</td>
</tr>
</tbody>
</table>

---

**Procedure Name**

**DB2**
Procedure name associated with the originator of the query; for example, the DB2 stored procedure name from DDF.
If the first SQL statement issued by the DDF client is a CALL statement, this field contains the unqualified name of the DB2 stored procedure. In all other cases, this field contains blanks.

**Process Name**

**DB2**  
A client application name

**DDF**  
A client application name

**MQ**  
The MQWIH_ServiceName from the message's work information header.

**Scheduling Environment Name**

**JES**  
The scheduling environment name assigned to the job.

**DB2**  
Scheduling environment name associated with the originator of the query.

**DDF**  
The user ID for the client. This can be different from the authorization ID used to connect to DB2. This information is for identification only, and is not used for any authorization.

**Subsystem Collection Name**

**JES**  
The XCF group name.

**DB2**  
Subsystem collection name associated with the originator of the query.

**DDF**  
The subsystem collection name is the name assigned by the subsystem to related groups of its work. For example, DB2 data sharing group name.

**Subsystem Instance**

You can use subsystem instance to isolate multiple instances of a subsystem. For example, use subsystem instance if you have a CICS production system as well as a CICS test system.

**CB**  
The WebSphere Application Server specific short name.

**CICS**  
The VTAM applid for the subsystem instance. For more information, see CICS/ESA Dynamic Transaction Routing in a CICSpexion.

**DB2**  
The subsystem type associated with the originator of the query; for example:
  - TSO for requests from TSO/E
  - JES for requests from batch jobs
  - DDF for requests from DDF

**DDF**  
The DB2 server's MVS subsystem name. For more information about the name, see DB2 Administration Guide

**IMS**  
The IMS subsystem name, as defined on the IMSID positional parameter in the IMS DFSMPR procedure. It is a 1- to 4-character value that uniquely identifies the control region. The generation default is IMSA. For more information, see IMS/ESA System Definition Reference

**IWEB**  
The subsystem name from the application environment definition. (Note that this is identical to bits 0-7 of the Subsystem Parameter qualifier for IWEB).
JES  The JES2 or JES3 subsystem name from the IEFSSNx parmlib member.

LDAP The z/OS LDAP server’s job name. Needed to distinguish between different LDAP servers.

LSFM The procname of address space in which LAN Server for MVS is running.

MQ The MQSeries Workflow subsystem name.

NETV The NetView WLM subsystem name as coded in CNMSTYLE. This is usually the NetView domain name. For more information, see Tivoli NetView Administration Reference.

TCP The TCP/IP jobname. For further information, refer to z/OS Communication Server IP Configuration Guide.

Subsystem Parameter
If you have a vendor or home-grown subsystem type that has a qualifier other than the IBM supported ones, it could use the subsystem parameter. You should check your subsystem documentation to determine whether your subsystem supports the subsystem parameter, and in what parameter format.

Because the subsystem parameter is up to 256 characters long, you can nest to use more than the limit of eight characters. Nesting is allowed only for qualifiers longer than 8 characters. These are: accounting information, correlation information, connection type, and subsystem parameter.

DB2 The subsystem parameter, if any, associated with the originator of the query.

DDF The subsystem parameter. This qualifier has a maximum length of 255 bytes. The first 16 bytes contain the client's user ID. The next 18 bytes contain the client's workstation name. The remaining 221 bytes are reserved.

Note the following:
- If the length of the client's user ID is less than 16 bytes, use blanks after the user ID to pad the length.
- If the length of the client's workstation is less than 18 bytes, use blanks after the workstation name to pad the length.

IWEB A 47-byte string formatted as follows:

1-8 Subsystem name
9 Blank
10-24 Source IP address
25 Blank
26-40 Target IP address
41 Blank
42-47 Target port

For more information, see Internet Connection Server User's Guide.

MQ The 32-byte application_environment_name from the APPLICID attribute of the process definition associated with the WLM-managed queue.

SOM A 246-byte string consisting of two 123-byte fields:
- Field 1 — class name
- Field 2 — method name
For more information, see z/OS SOMobjects Configuration and Administration Guide.

**STC**
Indicates the system-provided service class name that will be assigned if a started task created with the high dispatching priority, privileged, or system task attribute is not assigned to a service class.

Values:
- **SYSTEM** — Started task was created with high dispatching priority attribute.
- **SYSSTC** — Started task is privileged or is a system task.
- (blank) — Started task was not created with the high dispatching attribute, is not privileged, and is not a system task.

**Sysplex Name**
For all subsystem types, use the sysplex name qualifier if you have a common service definition in multiple sysplexes, and need to assign different service classes or report classes based on the specific sysplex in which the work is running.

**System Name**
The system name qualifier is supported for address spaces whose execution system is known at classification time. Note that JES is not eligible for this qualifier, as the system on which classification occurs may not be the system on which the job is run. Subsystem-defined transactions (CICS/IMS) and enclave-based transactions are not bound to an execution system at classification time, and are therefore not eligible either.

- **ASCH**
  The name of the execution system.
- **OMVS**
  The name of the execution system.
- **STC**
  The name of the execution system.
- **TSO**
  The name of the execution system.
- **SYSH**
  The name of the execution system.

**Transaction Class/Job Class**

- **ASCH**
  The job class used for work selection.
- **CB**
  Name resulting from mapping the URI to a name. For more information, see WebSphere Application Server for z/OS: Installation and Customization.
- **DB2**
  The job or transaction class associated with the originator of the query; for example, the job class from JES.
- **IMS**
  The CLASS keyword on the PGMTYPE=parameter in the APPLCNT macro. For more information, see IMS/ESA System Definition Reference.
- **IWEB**
  The arbitrary class name specified in the APPLENV directive in the Webserver's administrative file. Using the filtering function in the webserver, you can assign transactions to transaction classes based on the requested URL. The transaction classes can then be assigned unique service classes using this Transaction Class qualifier. Note that this is probably the most useful qualifier for IWEB work, because of its flexibility. For more information, see z/OS HTTP Server Planning, Installing, and Using.
Note that for the function Fast Response Cache Accelerator for high performance handling of cached static web pages, you must classify work as described in the following paragraph. Otherwise, it will be assigned the default service class for IWEB work.

The transactions handled by the Cache Accelerator are all joined to a single, long-lived enclave. This enclave should be assigned a unique transaction class (as specified on the Webserver FRCAWLMParms directive). This transaction class should then be assigned to a service class with a single period and a velocity goal in the service policy under the IWEB subsystem type. Neither response time goals nor multiple periods are appropriate for this work, as WLM is not aware of the individual Cache Accelerator requests. (Because each individual transaction is so trivial, it would cost more resource to manage them than to just process them.) In RMF reports, you will see zero ended transactions for the Cache Accelerator service class (assuming you have no other work running in this service class), but you will see some amount of accumulated service for this single enclave.

**JES**
The job class used for work selection.

**MQ**
A value of either ONLINE, meaning that the server is immediately available, or BACKLOG, meaning that the message was queued pending availability of a server.

**NETV**
The NetView subtask type. Valid types are: AOST, DST, HCT, MNT, NNT, OPT, OST, and PPT. For more information, see Tivoli NetView Customization: Using Assembler.

**Transaction Name/Job Name**

**ASCH**
The jobname in the JCL JOB statement in the APPC/MVS transaction program (TP) profile.

**CB**
The method name, for example, GET, POST, or DELETE. For more information, see WebSphere Application Server for z/OS: Installation and Customization.

**CICS**
A parameter on many CICS commands. It is often referred to as the CICS transaction identifier, or tranid. For more information, see CICS/ESA Resource Definition Guide.

**DB2**
The transaction or job name associated with the originator of the query; for example, the job name from JES.

**IMS**
The CODE= parameter on the IMS TRANSACT macro. For more information, see IMS/ESA System Definition Reference.

**IWEB**
The method name, for example, GET, HEAD, POST, DELETE, or PUT.

**JES**
The jobname of the JES managed job. For example, you may run a CICS region as a batch job in your installation. You would define it in classification rules as a transaction name in the JES subsystem type.

**LDAP**
The z/OS LDAP server’s enclave transaction name:
- Any transaction name that is also defined in the configuration file of the directory server.
- GENERAL for all LDAP work that is not assigned a user-defined exception class.
For further information, refer to *z/OS IBM Tivoli Directory Server Administration and Use for z/OS*.

**LSFM**
One of the following:
- LSFMMMTX - multi-media transactions
- LSFMFITX - file transactions
- LSFMAMTX - administration transactions
- LSFCMCTX - communication transactions

**MQ**
The MQWIH_ServiceStep value from the message's work information header.

**NETV**
The NetView subtask module name. For more information, see the NetView TASK statement description in the *Tivoli NetView Administration Reference*.

**OMVS**
The jobname for the z/OS UNIX System Services address space. By default, fork and spawn set jobname values to the user ID with a number (1-9) appended. However, daemons or users with appropriate privileges can set the _BPX_JOBNAME environment variable to change the jobname for forked or spawned children. In this way, servers and daemons in z/OS UNIX System Services address spaces can easily be assigned performance attributes different than other z/OS UNIX System Services address spaces.

**STC**
One of the following:
- The name as specified on the JOBNAME= parameter of the START command
- The name specified on the MOUNT command
- The system address space name
- The name on the JOB statement.

For example, you may run your IMS regions as started tasks in your installation. You would define these as transaction names in your STC subsystem type in classification rules. However, if IMS V5.1 is present, the STC rule for IMS is ignored and the rules for the IMS subsystem type are used when IMS becomes a server.

**TCP**
The z/OS Communications Server transaction name associated with data traffic being processed by z/OS Communications Server on an independent enclave that has been active for a relatively long period of time. Currently the only transaction name supported by z/OS Communications Server is TCPENC01 for IPSec traffic. For further information, refer to *z/OS Communication Server IP Configuration Guide*.

**Userid**

**ASCH**
The user ID of the user requesting the APPC/MVS service.

**CB**
The user ID of the user requesting the WebSphere Application Server service.

**CICS**
The user ID specified at LOGON time, which is the RACF (or other access control facility) defined resource. For more information about CICS user IDs, see *CICS/ESA CICS-RACF Security Guide*.

**DB2**
The user ID associated with the originator of the query; for example, the user ID from TSO, JES, or DDF.

**DDF**
The DDF server thread’s primary AUTHID, after inbound name translation.
IMS  The user ID specified at LOGON time, which is the RACF (or other access control facility) defined resource.

IWEB  The user ID of the web server address space (not the original requestor's user ID). Note that because this user ID will generally be the same for all transactions, using this qualifier for IWEB work will have limited usefulness.

JES  The user ID specified on the JOB statement on the RACF USER keyword.

MQ  The first 8 bytes of the 12-byte message header field MQMD_USERIDENTIFIER.

NETV  The NetView subtask ID. For NetView OST and NNT subtasks this is the NetView operator ID. For more information, see *Tivoli NetView User's Guide*. For AOST (automation subtasks) this is the NetView operator ID on the AUTOTASK command that started the AOST. For more information, see *Tivoli NetView Command Reference*. For NetView DST and OPT subtasks this is the taskname on the NetView TASK statement. For more information, see the *Tivoli NetView Administration Reference*. The MNT subtask ID is MNT. The PPT subtask ID is the NetView domain ID concatenated with the characters "PPT". The HCT subtask ID is the same as its LU name which is specified on the HARDCOPY statement. For more information, see *Tivoli NetView Administration Reference*.

OMVS  The RACF user ID associated with the address space. This user ID is either inherited from the parent process or assigned by a daemon process (for example, the rlogin or telnet daemon). For more information about z/OS UNIX System Services user IDs, see *z/OS UNIX System Services Planning*.

SOM  The user ID of the user requesting the SOM service.

STC  The user ID assigned to the started task by RACF (or other access control facility).

TSO  The user ID specified at LOGON time, which is the RACF (or other access control facility) defined user profile.

**zEnterprise Service Class Name (ESC) from a Unified Resource Manager Performance Policy**

EWLM  The subsystem type associated with work requests that originate from virtual servers in an ensemble workload. This subsystem type and work qualifier enable administrators to assign zEnterprise service classes to WLM service classes and report classes.

Table 9 shows which work qualifiers are supported by each IBM-defined subsystem.
Table 9. Work Qualifiers Supported by Each IBM-Defined Subsystem Type

<table>
<thead>
<tr>
<th>A</th>
<th>S</th>
<th>C</th>
<th>C</th>
<th>D</th>
<th>D</th>
<th>E</th>
<th>W</th>
<th>W</th>
<th>E</th>
<th>J</th>
<th>L</th>
<th>L</th>
<th>M</th>
<th>N</th>
<th>O</th>
<th>S</th>
<th>S</th>
<th>T</th>
<th>T</th>
<th>S</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>CH</td>
<td>CB</td>
<td>CS</td>
<td>DB</td>
<td>DF</td>
<td>LM</td>
<td>IM</td>
<td>MS</td>
<td>IM</td>
<td>ES</td>
<td>SP</td>
<td>FM</td>
<td>MQ</td>
<td>TV</td>
<td>VM</td>
<td>OM</td>
<td>SM</td>
<td>MS</td>
<td>ST</td>
<td>CP</td>
<td>TS</td>
<td>SY</td>
</tr>
<tr>
<td>Accounting Information</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>Collection Name</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>Connection Type</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>Correlation Information</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>zEnterprise Service Class</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>LU Name</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>Netid</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>Package Name</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>Perform</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>Plan Name</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>Priority</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>Procedure Name</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>Process Name</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>Scheduling Environment Name</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>Subsystem Collection Name</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>Subsystem Instance</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>Subsystem Parameter</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>Sysplex Name</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>System Name</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>-----------------------------</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>Transaction Name / Job Name</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>Userid</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>
</tbody>
</table>
For information about which qualifiers are used by other subsystems supporting workload management, see their supporting documentation.

**Defining z/OS WLM Classification Rules for Performance Policies from the Unified Resource Manager**

For virtual servers that run z/OS, you can map service classes from a Unified Resource Manager performance policy to z/OS WLM service classes to achieve end-to-end goal-based performance management for multitier applications. Through classification rules for the IBM-defined EWLM subsystem type, z/OS WLM administrators can assign work requests that originate from virtual servers in an ensemble workload to service classes and report classes on z/OS.

The EWLM subsystem work requests include DB2 distributed data facility (DDF) requests that originate from an ensemble, through virtual servers (or hosts) that are classified within a Unified Resource Manager performance policy. Unified Resource Manager performance policies have service classes that specify either discretionary or velocity performance goals that the Unified Resource Manager uses to manage virtual servers in the ensemble workload. When you correlate these zEnterprise service classes to WLM service classes, you enable z/OS WLM to manage the work differently than it does for local z/OS work:

- z/OS WLM assigns the incoming work request to the appropriate WLM service class immediately, instead of going through the z/OS classification process.
- z/OS WLM can manage the work processed by a z/OS subsystem, such as DDF, according to different performance goals than those used for local z/OS work.
- z/OS WLM also can assign the work requests originating from virtual servers in the ensemble to different report classes than those used for local z/OS work.

The configuration tasks necessary for mapping zEnterprise service classes to WLM services classes are described in the following. Ensemble workload administrators, performance management administrators, and z/OS WLM administrators collaborate to complete these tasks.

For the ensemble, perform the following steps:

1. Create a workload and add the z/OS virtual servers to it. Also enable guest platform management provider (GPMP) support for each z/OS virtual server.
2. Define policies, service classes, and classification rules for the workload, and schedule activation of the appropriate policy. Note that z/OS WLM restricts the length of service class names to 32 characters, so you need to observe this restriction when you define the names of zEnterprise service classes that you want to map to WLM service classes.
3. Configure and activate a guest platform management provider on virtual servers that support the tiers of a multitier application. Also enable application response measurement (ARM) for the IBM middleware running on those virtual servers.
   Depending on the configuration requirements of the multitier applications that your company uses, you might need to configure guest platform management providers and enable ARM for middleware not only on the virtual servers running z/OS, but also on z/VM® guests and virtual servers on a POWER® blade.

For z/OS images running on virtual servers, perform the following steps:

1. For each Unified Resource Manager workload that contains virtual servers running z/OS:
- Decide whether corresponding WLM service classes are required for the service classes within the Unified Resource Manager performance policy. If so, define the new WLM service classes that set only response time goals for Unified Resource Manager work requests. These response time goals apply to all DB2 distributed data facility (DDF) requests that originate from ensemble virtual servers that are classified within a performance policy from the Unified Resource Manager.

- Optionally, define one WLM service class as the default service class for work associated with a Unified Resource Manager workload policy.

2. To map zEnterprise service classes to their corresponding WLM service classes, modify the classification rules for the EWLM subsystem type. Use only the ESC work qualifier in the classification rules.

3. Optionally, specify WLM report classes to monitor the performance of Unified Resource Manager work requests separately from local z/OS work.

To map specific zEnterprise service classes to WLM service classes, you need to know the names of the zEnterprise service classes and the performance goals that they set. For example, suppose that a Unified Resource Manager performance policy defines performance goals for Internet sales. Within that policy is a service class named WebSales, containing classification rules that identify ensemble virtual servers that send work requests to z/OS images. For those virtual servers, the WebSales service class sets the business importance as high and the velocity goal as fast.

On z/OS, suppose that the z/OS WLM administrator defines two new WLM service classes for the work requests that arrive from z/OS virtual servers in the ensemble: ZMGRDEFA is the default service class for most work and GPFASTHI is the corresponding service class for the WebSales service class. Through these two WLM service classes, the z/OS WLM administrator defines different response time goals for the work requests that arrive from virtual servers in the same ensemble as the z/OS virtual servers.

To map the new WLM service classes to the zEnterprise service classes, the z/OS WLM administrator modifies the classification rules for the EWLM subsystem type. Figure 17 illustrates a sample WLM ISPF panel through which you map zEnterprise service classes to the corresponding WLM service classes:

![Figure 17. Sample z/OS WLM Classification Rules Panel for Mapping Unified Resource Manager performance policies](image)

On the left side of the panel, note the values under the Qualifier heading:
- In the Type column, ESC is the required work qualifier that you must use for the EWLM subsystem type.
- In the Name column, the names of several zEnterprise service classes are displayed. WebSales is at the top of the list. The entry for StoreSales illustrates how to enter a zEnterprise service class name that is longer than eight characters.

On the right side of the panel, note the values under the Class heading:
- In the Service column, the names of WLM service classes are displayed. ZMGRDEFA is the default service class for z/OS WLM to use for work requests that are associated with a zEnterprise service class other than those listed on the left side of the panel. GPFASTHI is listed across from its corresponding zEnterprise service class, WebSales.
- In the Report column, the name of a default report class is displayed. This name identifies the WLM report class that you can use to monitor performance data for work requests associated with all zEnterprise service classes.

When you use the ESC work qualifier for classification rules, you must observe the following restrictions:
- The ESC work qualifier is valid only for the EWLM subsystem type.
- A subrule must use the same work qualifier as its parent classification rule: ESC.
- A classification rule does not require a corresponding service class, and the EWLM subsystem type does not require a default service class. If processing of the classification rule does not result in the assignment of a valid WLM service class, the usual subsystem type specific WLM classification rules are applied to assign a WLM service class.
- The name of each specified zEnterprise service class cannot be longer than 32 characters.
- zEnterprise service class names are mixed case (fold qualifier name = N) and WLM uses case-sensitive names during the classification process.
- WLM uses classification rules starting from the top of the list as shown on the modify subsystem panel, so WLM uses the first correlation it finds for a zEnterprise service class.
- The selected WLM service class must have a response time goal; a velocity or discretionary goal is not supported.
- The selected WLM service class cannot have a multi-period goal.
- The selected WLM service class cannot belong to a resource group.

**Defining the Order of Classification Rules**

When the subsystem receives a work request, the system searches the classification rules for a matching qualifier and its service class or report class. Because a piece of work can have more than one work qualifier associated with it, it may match more than one classification rule. Therefore, the order in which you specify the classification rules determines which service classes are assigned.

Prior to z/OS V1R3, with the IEAICSxx, the system used a set order for searching the work qualifiers. With workload management, there is no set order. You define it. Only the subsystem type must be the first level qualifier for classification rules. You determine the rest of search order by the order in which you specify the classification rules. You can use a different hierarchy for each subsystem.
Example of Defining an Order of Rules

Suppose you are defining your JES classification rules. For all work requests coming into the JES2 subsystem instance, you want to assign work with user ID MNGRBIG and jobname PERFEVAL to service class BATSLOW. All other work with user ID beginning with MNGR should be assigned to service class BATHOT. So in this case, the hierarchy for the JES subsystem is:

1. Subsystem type, because it is always the first
2. Subsystem instance
3. Userid
4. Jobname

Defining a Subsystem Service Class Default

A service class default is the service class that is assigned if no other classification rule matches for that subsystem type. If you want to assign any work in a subsystem type to a service class, then you must assign a default service class for that subsystem — except for STC. You are not required to assign a default service class for STC, even if you assign started tasks to different service classes.

Optionally, you can assign a default report class for the subsystem type. If you want to assign work running in a subsystem to report classes, then you do not have to assign a default service class for that subsystem.

Using Inheritance in Classification Rules

Keep in mind that if you leave out the service class or report class on a classification rule, the work inherits the service class or report class from the higher level rule.

You can use this to your advantage by using the qualifier that applies to the most service classes in the subsystem as your first level qualifier. You can then list the exceptions in the subsequent levels.
### Example of Using Inheritance

For example, for CICS, the available qualifiers include subsystem instance, user ID, transaction name, and LU name.

**Subsystem Type**: CICS  
**Description**: CICS subsystem

<table>
<thead>
<tr>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
</table>
| 1 UI ATMA | 2 TN  
| CICSB | CASHA |
| 2 TN DEPOSIT | 3 LU WALLST |
| CICSA | BIGDEP |
| 1 UI ATMC | 2 TN  
| CICSC | CASHC |
| 2 TN DEPOSIT | 3 LU WALLST |
| CICSA | BIGDEP |

In this example, the installation set up their user IDs for their CICS work according to the ATM setup that they have. Since all of their interactive work is related to the ATMs, they chose user ID as their first level qualifier. Then, they wanted to separate out their cash transactions from their deposit transactions for reporting purposes, so they set up a report class for each.

The transactions do not have a service class explicitly assigned to them, so they inherit the service class from the rule one level before.

In addition, for the deposit transactions, they wanted to separate out those deposits coming from ATMs on Wall St location, because that area had been having some service troubles. So they defined a report class at a level 3 under the DEPOSIT transactions for each ATM user ID.

You cannot nest most qualifier types within themselves. For example, if you choose user ID as a first level, you cannot use user ID as a second level, or sub-rule qualifier. Nesting is allowed only for qualifiers longer than 8 characters. These are: accounting information, correlation information, collection name, procedure name, and subsystem parameter. See “Organizing Work for Classification” on page 84 for how to nest using the start position.

Keep in mind that the system sequentially checks for a match against each level one rule. When it finds a match, it continues just in that family through the level two rules for the first match. Similarly, if a match is found at any given level, then its sub-rules are searched for further qualifier matches. The last matching rule is the one that determines the service class and report class to be assigned.
Example 1: Using the Order of Rules

Suppose you have defined the following classification rules for your IMS work:

<table>
<thead>
<tr>
<th>Class</th>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFAULTS: IMSB</td>
<td>IMSA</td>
<td>IMSC</td>
</tr>
</tbody>
</table>

1 Tran Name 5128
2 LU Name BERMU

Suppose the following kind of work requests enter the system:

<table>
<thead>
<tr>
<th>Transaction</th>
<th>Tran Name</th>
<th>LU Name</th>
<th>Tran Name</th>
<th>LU Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5128</td>
<td>BERMU</td>
<td>6666</td>
<td>BERMU</td>
</tr>
<tr>
<td>2</td>
<td>5128</td>
<td>BERMU</td>
<td>5128</td>
<td>CANCUN</td>
</tr>
</tbody>
</table>

- Transaction 1 is assigned service class IMSC, because the transaction 5128 is from the BERMU LU name.
- Transaction 2 is assigned service class IMSB, the subsystem default, because it is not transaction 5128, and therefore the system never checks any sub-rules.
- Transaction 3 is assigned service class IMSA, because it is associated with LU name CANCUN.
- If you specified the classification rules as:

1 LU Name BERMU IMSC
2 Tran Name 5128 IMSA

Then all work from LU BERMU is assigned in service class IMSC, except work with Tran name 5128.
**Example 2: Using the Order of Rules**

Suppose you have defined the following rules for your IMS work:

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Service</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Report</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEFAULTS: PRDIMSR</td>
</tr>
<tr>
<td>1 SI</td>
<td>IMST</td>
<td>TRNIMSR</td>
</tr>
<tr>
<td>2 TC</td>
<td>15</td>
<td>TRNIMSNR</td>
</tr>
<tr>
<td>1 TC</td>
<td>15</td>
<td>PRDIMSNR</td>
</tr>
<tr>
<td>1 SI</td>
<td>IMSM</td>
<td>MDLIMSR</td>
</tr>
<tr>
<td>2 TC</td>
<td>15</td>
<td>MDLIMSNR</td>
</tr>
</tbody>
</table>

- If a work request in transaction class (TC) 15 enters the system from subsystem instance (SI) IMST, it is assigned service class TRNIMSNR.
- If a work request in transaction class (TC) 15 enters the system from subsystem instance (SI) IMSM, it is assigned service class PRDIMSNR and not MDLIMSNR as you might expect.

This is because the level 1 classification rule

1 TC 15 PRDIMSNR

comes first and the more explicit classification rule comes second:

1 SI IMSM MDLIMSR
2 TC 15 MDLIMSNR

The system stopped on the first level one match that it encountered. You can re-order the rules so that this does not occur. Put the most explicit rule first and the more general rule second as shown here:

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Service</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Report</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEFAULTS: PRDIMSR</td>
</tr>
<tr>
<td>1 SI</td>
<td>IMST</td>
<td>TRNIMSR</td>
</tr>
<tr>
<td>2 TC</td>
<td>15</td>
<td>TRNIMSNR</td>
</tr>
<tr>
<td>1 TC</td>
<td>15</td>
<td>PRDIMSNR</td>
</tr>
<tr>
<td>1 SI</td>
<td>IMSM</td>
<td>MDLIMSR</td>
</tr>
<tr>
<td>2 TC</td>
<td>15</td>
<td>MDLIMSNR</td>
</tr>
<tr>
<td>1 TC</td>
<td>15</td>
<td>PRDIMSNR</td>
</tr>
</tbody>
</table>

**System-Provided Service Classes**

If some work comes into a system for which there is no associated service class defined in the classification rules, workload management assigns it to a default service class. There are several such default service classes:

**SYSTEM**

For all system address spaces designated ‘high dispatching priority’ (X’FF’) address spaces. The high dispatching priority address spaces include: MASTER, TRACE, GRS, DUMPSRV, SMF, CATALOG, RASP, XCFAS, SMXC, CONSOLE, IOSAS, and others. For a list of the high dispatching priority address spaces in your installation, see the RMF Monitor II report and look for the x’FF’ dispatching priority.

You do not need to set up service classes for these system address spaces. Workload management recognizes these as special system address spaces and treats them accordingly.

If for some reason you do want to control these address spaces, you can do the following:

- Define a service class for them
• Set up a classification rule in the STC subsystem type which assigns the address space to a service class other than the default STC service class.

Note: To make sure that the system runs smoothly, certain address spaces cannot be freely assigned to all service classes. The following address spaces are always classified into service class SYSTEM, independently of the user defined classification rules:

*MASTER*
INIT
WLM
XCFAS
GRS
CONSOLE
IEFSCHAS
IXGLOGR
SMF
CATALOG
SMSPDSE
SMSPDSE1

When you assign a service class other than SYSTEM to a started task eligible for the SYSTEM service class, it loses the high dispatching priority attribute and runs at the dispatching priority of the assigned service class period. The high dispatching priority attribute can be restored by one of the following methods:

• You can use the RESET command to change the started task's service class to SYSTEM.
• You can change the classification rules to explicitly classify the started task to SYSTEM and activate a policy.

You can also assign work to the SYSTEM service class as part of your work classification rules. You can only do this, however, for classification rules in the STC subsystem type, and only for address spaces that are designated as “high dispatching priority” address spaces.

For more information about using SYSTEM in classification rules for started tasks, see “Using the System-Supplied Service Classes” on page 87.

SYSSTC2

For all started tasks not otherwise associated with a service class. Workload management treats work in SYSSTC just below special system address spaces in terms of dispatching.

You can also assign work to the SYSSTC service class as part of your work classification rules. You can do this for classification rules in the following subsystem types:

• ASCH
• JES
• OMVS (z/OS UNIX System Services)
• STC
• TSO

2. When NetView subtasks are WLM managed, refer also to Table 6 on page 60.
Some address spaces normally created when running MVS are neither high dispatching priority, privileged, nor a system task, such as NETVIEW. These address spaces must be explicitly assigned to a service class such as SYSSTC.

For more information about using SYSSTC in classification rules for started tasks, see "Using the System-Supplied Service Classes" on page 87.

**SYSSTC1 - SYSSTC5**

The service classes SYSSTC1, SYSSTC2, SYSSTC3, SYSSTC4, and SYSSTC5 are provided for future z/OS support. Service class SYSSTC and the SYSSTCx service classes are congruent as far as management of work is concerned. Work assigned to any of these service classes is managed identically to work assigned to any other. Currently, there is no technical reason to choose SYSSTCx as an alternative to SYSSTC. Many displays, for example, SDSF displays, provide the service class assigned to an address space. It is possible that one might assign SYSSTCx to convey some meaning that would be similar to a report class.

**SYSOTHER**

For all other work not associated with a service class. This is intended as a ‘catcher’ for all work whose subsystem type has no classification. It is assigned a discretionary goal.

---

**Organizing Work for Classification**

There are some ways you can organize your qualifiers for easier classification. You can use masking or wildcard notation as a way of grouping work to the same service class or report class. Or, you could set up a qualifier group when the qualifier is a standard 8 character value.

If you have more than five rules at a given level for the same classification qualifier within a subsystem type, there may be performance implications. Qualifier groups are quicker to check than a single rule, so it may make sense to use them for performance sensitive subsystems like CICS and IMS.

For qualifiers longer than 8 characters, you can use the start position. Accounting information, correlation information, connection type, procedure name, and the subsystem parameter are greater than 8 characters. The following sections explain each kind of notation.

**Using the Start Position**

For work qualifiers longer than 8 characters, you can use a start position to indicate how far to index into the character string for a match. For example, you can assign all TSO/E users in a department to the same service class, assuming you follow a naming convention for accounting information for the department.

The ISPF administrative application provides the Start field where you can specify the starting position. The name field for a work qualifier is 8 characters long. If you are matching on a string less than 8 characters, you must use wildcard notation (asterisk) at the end of the string. Otherwise, the qualifier is padded with blanks to be 8 characters, and the blanks are used when making a match. When no start parameter is specified, WLM matches the name field for work qualifiers according to the number of characters specified.
Examples of Using the Start Position

- Assume you want to associate all JES2 work from department IRS with the service class JESFAST. You assigned the default for JES2 work as service class JESMED. If all JES2 accounting information from department IRS have the characters 'DIRS' starting in the seventeenth position, you enter a rule with qualifier DIRS* to match on just the four characters. If you want to filter out those jobs with the eight characters 'DIRS ' starting in the seventeenth position, you need another rule with qualifier DIRS to assign those jobs to JESMED. The example shows the rules:

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Description</th>
<th>All JES2 service classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type Name Start Service Report</td>
<td>DEFAULT: JESMED</td>
<td>-------Qualifier------------ -------Class--------</td>
</tr>
<tr>
<td>1 AI DIRS 17 JESMED</td>
<td>1 AI DIRS* 17 JESFAST</td>
<td></td>
</tr>
</tbody>
</table>

In this case, all jobs that have accounting information with the eight characters 'DIRS ' starting in the seventeenth position are assigned to JESMED. All other jobs that have the four characters 'DIRS' starting in the seventeenth position are assigned to JESFAST. All other work coming into JES is assigned to service class JESMED.

Using Masking Notation

You can use masking notation to replace a single character within a qualifier. This allows any character to match the position in the rule. Use a % in the position where the character would be. You can use multiple masks successively for multiple character replacement. If you specify a mask at the end of a character string, it could match on a null value OR a single character.

Examples of Masking Notation

For example, suppose you have a naming convention for all users in your IS services department that all user IDs start with DEPT58, followed by a letter A-F (depending on which division), ended by an I. Suppose also, you would like to bill your IS services department separately. You could use masking notation in setting up the classification rules as shown below.

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Description</th>
<th>All JES2 service classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type Name Start Service Report</td>
<td>DEFAULTS: BATREG</td>
<td>-------Qualifier------------ -------Class--------</td>
</tr>
<tr>
<td>1 UI DEPT58%I</td>
<td>1 UI DEPT58</td>
<td></td>
</tr>
</tbody>
</table>

In the example, all work in this subsystem is associated with service class BATREG, and all work from the IS services department is associated with the report class DEPT58.

Using Wildcarding Notation

You can also use wildcard notation for multiple character replacement in a character string. The wildcard character is an asterisk (*). You can use the wildcard character as the last position of a character string, or by itself. If a character string contains an asterisk in a position other than the last, it is treated as a character; for example,
if you specify a character string of CI*S, the third character in a matching character string must have an asterisk (*) as the third character. An asterisk by itself indicates a match for all characters.

--- Examples of Wildcard Notation ---

For example, suppose your installation has a naming convention for your CICS AORS and TORS. You can use the following wildcard notation in your CICS classification rules. Note that the subsystem instance of CI*S is not wildcard notation, a matching subsystem instance must be CI*S.

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 TN</td>
<td>TOR*</td>
<td>___</td>
<td>CICSTC1_</td>
<td></td>
</tr>
<tr>
<td>1 TN</td>
<td>AOR*</td>
<td>___</td>
<td>CICSTC3_</td>
<td></td>
</tr>
<tr>
<td>1 SI</td>
<td>CI*S</td>
<td>___</td>
<td>CICSTEST_</td>
<td></td>
</tr>
</tbody>
</table>

--- Important Note ---

Be careful when putting specific definitions below wildcards which might cause an unwanted early match. In the example, the rule for TOR11 is useless, because a TOR11 transaction will match the TOR* rule before it.

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 TN</td>
<td>TOR*</td>
<td>___</td>
<td>CICSTC1_</td>
<td></td>
</tr>
<tr>
<td>1 TN</td>
<td>TOR11</td>
<td>___</td>
<td>CICSTC4_</td>
<td></td>
</tr>
</tbody>
</table>

--- Using Groups ---

Groups are available for grouping together work qualifiers to make classification simpler. You can create groups to collect together work when you don't have a standard naming convention that allows masking or wildcarding. A group is a collection of the same work qualifiers. For example, you may want to create a group of started tasks because you want to assign them all to the same service class.

Groups are allowed for the following work qualifiers:
- Connection type
- LU name
- Netid
- Package name
- Perform
- Plan name
- Subsystem instance
- System Name
- Transaction class
- Transaction name
- Userid

Group types are specified by adding G to the type abbreviation. For example, a transaction name group is indicated as TNG.
Qualifier groups of more than 5 members are quicker to check than single instances in the classification rules. So if you have, for example, a long list of CICS or IMS transaction names that you want to group in a service class or report class, consider setting up a group.

--- Examples of Groups ---

If you want to assign a large number of CICS transactions to the same service class, you can create a transaction name group (TNG). You name the group, for example CICSCONV, and list all the transaction names you want included in the group.

```
Qualifier type ..........: Transaction Name
Group name .............: CICSCONV (required)
Description .............: CICS Conversational Group
```

```
--Qualifier--
Name
CDBC
CDBI
CDBM
CEBR
CECI
CECS
CEDA
CEDB
```

```
Qualifier type ..........: Transaction Name
Group name .............: CICSLONG (required)
Description .............: CICS Long-Running Transactions
```

```
--Qualifier--
Name
CDB0
CSGX
CSNC
CSNE
CSSX
```

Then you use those group names in the classification rules, as shown in this panel:

```
Subsystem Type ........ : CICS
Description ............: CICS transactions

-------Qualifier--------- -------Class---------
Type Name Start Service Report
1 TNG CICSCONV ___ CICSCONV ________
1 TNG CICSLONG ___ CICSLONG ________
```

--- Using the System-Supplied Service Classes ---

You can also take advantage of the system-supplied service classes to simplify the process of defining service classes and classification rules for started tasks.

Use the system-provided service classes SYSTEM, and SYSSTC for your STC service classes. WLM recognizes certain system address spaces when they are created (like GRS, SMF, CATALOG, MASTER, RASP, XCFAS, SMXC, CONSOLE, IOSAS, WLM), puts them into the SYSTEM service class, and treats them accordingly. If a started task is not assigned to a service class, WLM manages the started task in the SYSSTC service class. Started tasks in SYSSTC are assigned a
high dispatching priority. This is appropriate for started tasks such as JES and VTAM. Not all started tasks are appropriate for SYSSTC, as a CPU-intensive started task could use a large amount of processor cycles. However, if your processor is lightly loaded, or in a 6-way, 8-way, or 10-way MP, SYSSTC might be appropriate, because that one task may not affect the ability of the remaining processors to manage the important work with goals.

Note: *MASTER*, INIT and WLM always run in the SYSTEM service class and cannot be reassigned via the service definition.

The following example implements started task classification with these assumptions:

- Any started tasks not explicitly classified are given low priority.
- Started tasks are defined in three transaction name groups: HI_STC, MED_STC, and LOW_STC.
- System defaults are used for MVS-owned address spaces.
- Separate reporting is used for Master, DUMPSRV, and GRS.

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Description</th>
<th>All started tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEFAULTS: DISC__</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 TN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 TN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 TN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 TNG</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 TNG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 TNG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 TNG</td>
</tr>
</tbody>
</table>

The first rule assigns *MASTER* to report class MASTER. (Note that “SYSTEM” is shown in the service class column. In this case, it is not a true assignment as *MASTER* must always run in the SYSTEM service class anyway.) *MASTER* could also be classified using the SPM SYSTEM rule, but since separate reporting is desired for *MASTER*, it is classified separately. GRS and DUMPSRV are handled similarly.

The SPM rules assign started tasks created with the high dispatching priority attribute to SYSTEM, and other privileged or system tasks to SYSSTC. This allows you to let MVS manage started tasks that it recognizes as special.

Notes:

1. Note that explicitly defining the SPM rules, as in the example at hand, is optional. If they were removed from the example, then the high dispatching priority work would still be assigned to SYSTEM, and the other privileged or system tasks would still be assigned to SYSSTC, as those are the defaults. The reason for explicitly defining them here with the SPM rules is to protect yourself from inadvertently assigning them elsewhere in the rules that follow the SPM rules.

2. Note, also, that the placement of these SPM rules is crucial. If they had been declared first, then the three TN rules intended to assign *MASTER*, DUMPSRV, and GRS to report classes would have never been reached.
System tasks are those given the privileged and/or system task attribute in the IBM-supplied program properties table or in the SCHEDxx parmlib member. See the SCHEDxx chapter in *z/OS MVS Initialization and Tuning Reference* for a list of system tasks. An example for such a system task is DB's subsystem parameter module, the program DSNYASCP that is defined in the IBM-supplied PPT (IEFSDPPT) with the system task attribute. Without the fourth classification rule, the SPM rules would enforce all DB2 subsystem address spaces that execute this program (MSTR, DBM1, DIST, SPAS) to be classified into SYSSTC.

A transaction name group is defined to match all DB2 subsystem regions with 4-character subsystem names not already classified. Such a group for DB2 V4 could be defined as follows:

- **Qualifier type:** Transaction Name
- **Group name:** DB2
- **Description:** All non-IRLM DB2 regions
- **Fold qualifier names?** Y (Y or N)

```
<table>
<thead>
<tr>
<th>Qualified Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>%%%%%DBM1</td>
</tr>
<tr>
<td>%%%%%MSTR</td>
</tr>
<tr>
<td>%%%%%DIST</td>
</tr>
<tr>
<td>%%%%%SPAS</td>
</tr>
</tbody>
</table>
```

**Note:** The “Fold qualifier names” option, set to the default Y, means that the qualifier names will be folded to uppercase as soon as you type them and press Enter. If you set this option to N, the qualifier names will remain in the case they are typed in. Leave this option set to Y unless you know that you need mixed case qualifier names in your classification rules.

Started tasks in the HI_STC transaction name group are run in service class SYSSTC, MED_STC in VEL35I3, and LOW_STC in VEL15I5. The goals on the service classes use importance to guarantee that started tasks of low value will be sacrificed if necessary to keep medium-value started tasks meeting goals. Note that this does not guarantee any relationship between the dispatching priorities that will be observed for service classes VEL35I3 and VEL15I5.

Started tasks which do not match any of the classification rules are assigned the default service class DISC. If the default service class were left blank, these started tasks would be assigned to SYSSTC.
Chapter 10. Defining Report Classes

Optionally, classification rules can assign incoming work to a report class. Workload management provides data for reporting on all of the service definition terms on a service class period and workload basis. Report classes can be used to report on a subset of transactions running in a single service class but also to combine transactions running in different service classes within one report class. In the first case, a report class is called homogeneous, in the second case it is called heterogeneous. Heterogeneous report classes can cause incorrect performance data, since the data collected is based on different goals, importance, or duration. WLM allows a caller of the workload reporting service IWMRCOLL to detect whether a report class is homogeneous or heterogeneous for a given report interval. See z/OS MVS Programming: Workload Management Services for more information on how a reporting product can receive an indication whether a report class is homogeneous or heterogeneous.

The data available for report classes include:
- Number of transactions completed
- Average response times
- Resource usage data
- State samples
- Response time distribution buckets
- Work manager delay data

Each item in the list is related to a report class period. The reporting products are advised to report response time distributions only for homogenous report class periods, where only 1 service class contributed data to the report class.

For CICS and IMS workloads, resource usage data and state samples are reported with the service classes for the regions, not the service classes assigned to the transactions.

Note that a report class can have as many periods as the largest service class within the same service definition has periods. Not all of them may be used, though. Report classes inherit periods automatically based on the transaction’s service class period that is contributing to the report class. A reporting product can determine whether a report class is homogeneous within a given reporting interval and is advised to report the data as one period if the report class is heterogeneous. Based on its capabilities and options, a reporting product can report the data of a homogeneous report class as one period or as multiple periods.

For subsystem STC initiator INITs should never be in a report class.

You can assign up to a maximum of 2047 report classes, with no more than one report class per work request or transaction.

<table>
<thead>
<tr>
<th>Defining Report Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
</tr>
<tr>
<td><strong>Description</strong></td>
</tr>
</tbody>
</table>
### Example 1: Defining Report Classes

Suppose you have defined the following rules for your CICS work:

<table>
<thead>
<tr>
<th>Subsystem Type: CICS</th>
<th>Description: CICS subsystem</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICSB</td>
<td>ATMA</td>
</tr>
<tr>
<td>CICSA</td>
<td>CASHA</td>
</tr>
<tr>
<td>DEPOSITA</td>
<td>BIGDEP</td>
</tr>
<tr>
<td>ATMC</td>
<td>CICSC</td>
</tr>
<tr>
<td>CASHC</td>
<td>DEPOSITC</td>
</tr>
<tr>
<td>BIGDEP</td>
<td>BIGDEP</td>
</tr>
</tbody>
</table>

In this example, the cash transactions are separated from their deposit transactions for reporting purposes. Report class ATMA therefore, does not include all transactions with userid ATMA, because it does not include the cash or deposit transactions.
Chapter 11. Defining Service Coefficients and Options

Firstly, this information explains how service can be calculated in WLM (see "Calculating the Amount of Service Consumed").

Next, this information describes the options that must be specified for workload management. They are the following:

- How the system calculates the amount of resources that work consumes with service coefficients (see "Service Definition Coefficients" on page 94).
- Whether workload management is to dynamically set I/O priorities based on performance goals (see "Specifying I/O Priority Management" on page 96).
- Whether workload management is to dynamically reassign parallel access volume alias addresses based on performance goals (see "Specifying Dynamic Alias Management" on page 96).

All of these options can be set using the Service Coefficient/Service Definition Options panel in the WLM ISPF application, as shown in "Working with Service Coefficients and Options" on page 207.

Calculating the Amount of Service Consumed

One of the basic functions of WLM/SRM is to monitor the dynamic performance characteristics of all address spaces under its control to ensure distribution of system resources as intended by the installation.

A fundamental aspect of these performance characteristics is the rate at which an address space is receiving service relative to other address spaces competing for resources within the same domain.

The amount of service consumed by an address space is computed by the following formula:

\[
\text{service} = (\text{CPU} \times \text{CPU Service Units}) + (\text{SRB} \times \text{SRB Service Units}) + (\text{IOC} \times \text{I/O Service Units}) + (\text{MSO} \times \text{Storage Service Units})
\]

Figure 18. Formula for Calculating Service Consumption

where CPU, IOC, MSO, and SRB are installation defined service definition coefficients and:

**CPU Service Units =**

- task (TCB) execution time, multiplied by an SRM constant which is CPU model dependent. Included in the execution time is the time used by the address space while executing in cross-memory mode (that is, during either secondary addressing mode or a cross-memory call). This execution time is not counted for the address space that is the target of the cross-memory reference.

**SRB Service Units =**

- service request block (SRB) execution time for both local and global SRBs, multiplied by an SRM constant which is CPU model dependent. Included in the execution time is the time used by the address space while executing in cross-memory mode (that is, during either secondary addressing mode
or a cross-memory call). This execution time is not counted for the address space that is the target of the cross-memory reference.

**I/O Service Units** =
measurement of data set I/O activity and JES spool reads and writes for all data sets associated with the address space. SRM calculates I/O service using I/O block (EXCP) counts. When an address space executes in cross-memory mode (that is, during either secondary addressing mode or a cross-memory call), the EXCP counts or the DCTI will be included in the I/O service total. This I/O service is not counted for the address space that is the target of the cross-memory reference.

**Storage Service Units** =
(central page frames) x (CPU service units) x 1/50, where 1/50 is a scaling factor designed to bring the storage service component in line with the CPU component. NOT included in the storage service unit calculation are the central storage page frames used by an address space while referencing the private virtual storage of another address space through a cross service (that is, through secondary addressing or a cross-memory call). These frames are counted for the address space whose virtual storage is being referenced.

---

**Service Definition Coefficients**

The amount of system resources an address space or enclave consumes is measured in service units. Service units are calculated based on the CPU, SRB, I/O, and storage (MSO) service an address space consumes.

Service units are the basis for period switching within a service class that has multiple periods. The duration of a service class period is specified in terms of service units. When an address space or enclave running in the service class period has consumed the amount of service specified by the duration, workload management moves it to the next period. The work is managed to the goal and importance of the new period.

Because not all kinds of services are equal in every installation, you can assign additional weight to one kind of service over another. This weight is called a service coefficient.

**Changing Your Coefficient Values**

Prior to z/OS V1R3, you could use the same coefficients as used in the IEAIPSxx parmlib member. Then you could directly compare RMF data, and determine your durations properly.

However, if you plan to use workload management, it is probably a good time for you to rethink your coefficients. The current defaults are inflated, given the size and processing capability of processors. Processors can consume much higher amounts of service, and as a result, service unit consumption numbers are very high. These high numbers can cause problems if they reach the point where they wrap in the SMF fields. If they wrap, you may see abnormally large transaction counts, and last period work may be restarted in the first period.

It is possible for you to make them smaller, yet still maintain the same relationship between the coefficient values. Consider changing your definitions to the following:

**CPU** 1
If you do decide to change the coefficients, you must re-calculate your durations and accounting procedures.

HINT: If you want to gather storage service information by service class, but don't want it affecting your durations or accounting procedures, use an MSO coefficient of 0.0001. This results in very low MSO service unit numbers, but still allows you to obtain storage service information through RMF.

Since changing the coefficients affects durations and accounting values, the defaults are meant to be consistent with settings seen in the field today. If you do not define the service coefficients, the defaults are:

<table>
<thead>
<tr>
<th>Service</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>10.0</td>
</tr>
<tr>
<td>IOC</td>
<td>5.0</td>
</tr>
<tr>
<td>MSO</td>
<td>0.0</td>
</tr>
<tr>
<td>SRB</td>
<td>10.0</td>
</tr>
</tbody>
</table>

**Using the Storage (MSO) Coefficient for Calculations**

The MSO service definition coefficient is externalized in the following SMF records:

- **SMF Type 30**
  - Performance Section, Field SMF30MSC
- **SMF Type 72 Subtype 3**
  - Workload Manager Control Section, Field R723MMSO

The externalized value is the value specified in the WLM administrative application, scaled up by 10 000. For example, if you specify MSO = 0.0001 the externalized value is 1; if you specify MSO = 1.0 the externalized value is 10 000.

The idea of storage service units is to account for central storage being held while CPU cycles are being used. The basic unit of measure is a page frame (4 096 bytes) held for one CPU service unit. To make MSO roughly commensurate with CPU service units, the raw number is divided by 50 to yield MSO service units.

By scaling the MSO value with 4 096 and dividing it by 50, the internal value used by SRM becomes slightly less precise than the input value which is externalized in the SMF records. The following formula is used for scaling:

\[
\frac{\text{Input MSO Coefficient scaled by } 10000 \times 4096}{1000} + 1
\]

*Figure 19. MSO Coefficient Formula*

The result is truncated to the nearest integer value. Therefore, input values between 0.0001 and 0.0122 all result in a value of 1. An MSO coefficient of 1 results in a value of 82 which is used by SRM to calculate MSO service.
To use the MSO service definition coefficient for your own calculations, apply the same scaling in order to understand the storage service units shown in the SMF 30 and SMF Type 72 records.

Specifying I/O Priority Management

I/O priority queueing is used to control non-paging DASD I/O requests that are queued because the device is busy. You can optionally have the system manage I/O priorities in the sysplex based on service class goals.

The default for I/O priority management is no, which sets I/O priorities equal to dispatching priorities. If you specify yes, workload management sets I/O priorities in the sysplex based on goals.

WLM dynamically adjusts the I/O priority based on how well each service class is meeting its goals and whether the device can contribute to meeting the goal. The system does not micro-manage the I/O priorities, and changes a service class period’s I/O priority infrequently.

When I/O priority management is on, I/O samples are used in the velocity formula. See “Velocity Formula” on page 50 for more information.

Considerations for I/O Priority Management

If you specify I/O priority management, workload management dynamically sets I/O priorities based on goals and I/O activity, and includes the I/O information when calculating execution velocity. So you might see some changes in your velocity values. The recommended setting for I/O priority management is YES.

The new DASD I/O using and DASD I/O delay samples are reported even when I/O priority management is turned off. This allows you to calculate the new velocity values to plan for velocity changes.

Specifying Dynamic Alias Management

Workload Management Considerations

As part of the Enterprise Storage Subsystem’s implementation of parallel access volumes, the concept of base addresses versus alias addresses is introduced. While the base address is the actual unit address of a given volume, there can be many alias addresses assigned to a base address, and any or all of those alias addresses can be reassigned to a different base address. With dynamic alias management, WLM can automatically perform those alias address reassignments to help work meet its goals and to minimize IOS queueing.

Note that to be able to move aliases to other bases, the aliases first have to have been initially assigned to bases via the ESS specialist, and the bases have to have come online at some point.

When you specify yes for this value on the Service Coefficient/Service Definition Options panel, you enable dynamic alias management globally throughout the sysplex. WLM will keep track of the devices used by different workloads and broadcast this information to other systems in the sysplex. If WLM determines that a workload is not meeting its goal due to IOS queue time, then WLM attempts to
find alias devices that can be moved to help that workload achieve its goal. Even if all work is meeting its goals, WLM will attempt to move aliases to the busiest devices to minimize overall queueing.

In addition to automatically managing aliases, WLM ensures during system initialization that a minimum number of aliases is assigned to parallel access volumes with page data sets. The minimum number of aliases for a volume is:

\[ 2 \times \text{number of page datasets} - 1 \]

This allows each page data set to have two I/Os active at a time, and ensures that paging intensive activities such as system dumping are not delayed by IOS queueing. This automatic enforcement of the minimum aliases only happens if dynamic parallel access volumes management is active for the device.

**IMPORTANT:** If you enable dynamic alias management, you must also enable I/O priority management. So you need to specify `yes` for both of these options on the panel. If I/O priority management is set to `no`, you will get only the efficiency part of dynamic alias management and not the goal-oriented management. This means that WLM will make alias moves that minimize overall IOS queueing, but these moves will not take service class goals into consideration.

**HCD Considerations**

While you can globally enable or disable dynamic alias management on the WLM ISPF panel, you can also individually enable or disable dynamic alias management on a given device via HCD. You can do this by specifying `WLMPAV=YES` or `NO` in that device’s HCD definition.

Note, however, that there is no consistency checking for dynamic alias management between different systems in a sysplex. If at least one system in the sysplex specifies `WLMPAV=YES` for a device, then dynamic alias tuning will be enabled on that device for all systems in the sysplex, even if other systems have specified `WLMPAV=NO`. It is recommended **not** to use dynamic alias management for a device unless all systems sharing that device have dynamic alias management enabled. Otherwise, WLM will be attempting to manage alias assignments without taking into account the activity from the non-participating systems.

Note, also, that you can specify `WLMPAV=YES` or `NO` on both base and alias devices. The WLMPAV settings on an alias device, however, is only meaningful when the alias device is bound to a base device that is offline, as follows:

- If the base device is *offline*, then only alias devices with WLMPAV set to YES will be reassigned to other base devices.

  The WLMPAV setting on the base device itself is irrelevant when the base device is offline, for either “giving” or “receiving” aliases. (Even if WLMPAV was set to YES on the base device, it cannot have new aliases assigned to it, as it is offline.)

- For any base device that is offline to one or more systems in the sysplex and online to others, the WLMPAV keyword in HCD needs to be set to NO for the base device and its aliases. You need to statically assign the desired number of aliases for the base device via the ESS Specialist. If you try to use dynamic alias management for such a device, WLM will make unpredictable alias moves.

- If the base device is *online*, then the WLMPAV settings on the aliases are ignored, as follows:
  - If WLMPAV is set to YES on the base device, then the aliases can be reassigned regardless of their WLMPAV settings.
- If WLMPAV is set to NO on the base device, then no aliases can be reassigned, regardless of their WLMPAV settings.

For a WLMPAV=YES base device, the aliases initially assigned to it should be allowed to default to YES. The only situation where you might want to change an alias to WLMPAV=NO is if the alias is initially assigned to a WLMPAV=NO base device. Because the base is set to NO, the aliases initially assigned to it will not be moved to other bases by WLM. Then, because the aliases are set to NO, if the base is ever varied offline, the aliases remain assigned to that base and cannot be reassigned by WLM to other bases. Certain combinations of WLMPAV settings are not recommended, as described in table "Table 10".

### Table 10. Effects of WLMPAV Settings on Base and Alias Devices

<table>
<thead>
<tr>
<th>Base Device WLMPAV Setting</th>
<th>Alias Device WLMPAV Setting</th>
<th>Effects/Recommendations</th>
</tr>
</thead>
</table>
| YES                        | YES                         | - If base is online: Base is WLM-managed. Aliases can be freely moved to and from the base device by WLM.  
- If base is offline: Aliases become unbound and are available to WLM to assign to other WLM-managed bases. |
| YES                        | NO                          | Not recommended. If base is WLM-managed, then it is not predictable which aliases will remain bound to that base when the base goes offline. If the base device is set to YES, then you should set the aliases to YES as well. (See previous option.) |
| NO                         | YES                         | Not recommended. If the base is not WLM-managed, then you risk losing all of its aliases when the device goes offline. (See next option.) |
| NO                         | NO                          | - If base is online: Base is not WLM-managed. The initial aliases assigned to this base remain there.  
- If base is offline: Aliases remain bound to the offline base device and are not available to WLM for reassignment. When the base comes back online, it retains its initial alias assignments. |

In order for dynamic alias management to be most effective, you should try to spread out your aliases in the initial definition. If one base device has several alias devices while other base devices have none, it will take more time for WLM to reassign the aliases appropriately. Ideally, you should have at least two aliases assigned to each base at the outset.

For further information on HCD definitions, refer to [z/OS HCD Planning].
Chapter 12. Defining Special Protection Options for Critical Work

Several options are available to help performance administrators protect critical work. Although applicable to several other subsystem types, CICS and IMS work will particularly benefit from the enhancements described in this section:

- Long-term storage protection
- Long-term CPU protection
- Modifications of transaction response time management

These three options are described, and then illustrated by an example in “Sample Scenarios” on page 102.

Important

The use of these options limits WLM’s ability to manage the system. This may affect system performance and/or reduce the system’s overall throughput.

Long-Term Storage Protection

When you assign long-term storage protection to critical work, WLM restricts storage donations to other work. This option can be useful for work that needs to retain storage during long periods of inactivity because it cannot afford paging delays when it becomes active again. With long-term storage protection assigned, this work will lose storage only to other work of equal or greater importance that needs the storage to meet performance goals.

You assign long-term storage protection with the “Storage Critical” option, found by scrolling right on the “Modify Rules for the Subsystem Type” panel:

![Modify Rules for the Subsystem Type Row 1 to 2 of Command](image)

You can assign long-term storage protection to all types of address spaces using classification rules for subsystem types ASCH, JES, OMVS, STC, and TSO. By specifying YES in the “Storage Critical” field for a classification rule, you assign storage protection to all address spaces that match that classification rule. An address space must be in a service class that meets two requirements, however, before it can be storage-protected:
• The service class must have a single period.
• The service class must have either a velocity goal, or a response time goal of over 20 seconds.

**Note:** These 2 requirements only apply to the address spaces classified under subsystem types ASCH, JES, OMVS, STC, and TSO.

For CICS and IMS work, you can assign long-term storage protection by specifying YES in the “Storage Critical” field in the rules for specific transactions. Once you specify YES for one transaction in a CICS/IMS service class, *all CICS/IMS transactions in that service class* will be storage-protected. If a CICS or IMS region is managed as a server by WLM (managed to the response time goals of the transactions it serves) and *any* of the transaction service classes it serves is assigned storage protection, then the CICS/IMS region itself is automatically storage-protected by WLM.

As an alternative to assigning storage protection based on specific transaction service classes, you can instead choose to assign storage protection to the region in which the transactions run. You do this by adding or modifying the STC or JES classification rule that assigns the service class to the region.

### Long-Term CPU Protection

When you assign long-term CPU protection to critical work, you ensure that less important work will generally have a lower dispatch priority. (There are some rare exceptions, such as when other work is promoted because it is holding an enqueue for which there is contention.) This protection can be valuable for work which is extremely CPU-sensitive, such as certain CICS and IMS transactions.

Use the “Cpu Critical” option on the “Modify a Service Class” panel to assign long-term CPU protection to a specific service class:

![Create a Service Class](image)

*Figure 21. Specifying the “Cpu Critical” Option*

You can assign CPU protection to service classes handling address space-oriented work, enclave work, or CICS/IMS transactions, but the service class must have only one period, and it cannot have a discretionary goal. If a CICS or IMS region is managed as a server by WLM (managed to the response time goals of the transactions it serves) and *any* of the transaction service classes it serves is assigned CPU protection, then the CICS/IMS region itself is automatically CPU-protected by WLM.
Modifications of Transaction Response Time Management

Use the “Manage Region Using Goals Of:” field in the “Modify Rules for the Subsystem Type” panel to declare that a specific CICS/IMS region will not be managed to the response times of the CICS/IMS transactions that it processes. Note that other regions are not affected by what is in this column, and that this option can only be used in STC and JES classification rules:

If you specify TRANSACTION in this field (the default), the region will be managed as a CICS/IMS transaction server by WLM. If you specify REGION in this field, the region will be managed to the performance goal of the service class assigned to that region (address space). In other words, it will not be managed as a CICS/IMS transaction server by WLM.

If you specify BOTH in this field, the region will also be managed to the performance goal of the service class assigned to that region, but it will nevertheless track all transaction completions so that WLM can still manage the CICS service classes with response time goals. Option BOTH should only be used for CICS TORs. All AORs should remain at the default TRANSACTION.

If you specify TRANSACTION or BOTH, RMF will report performance information as follows:

- Response time data is reported in the WLMGL SCPER report for the service class in which those transactions are running.
- Response time data is also reported in the WLMGL RCLASS report for the report class in which those transactions are running.
- The service classes served by the region are reported in the WLMGL SCLASS report for the service class in which the region is running.

If you specify REGION, only the information in the SCPER report will change. Transaction response times reported by these regions will not be reported in any service class. This response time data will still be reported in the RCLASS report for the transaction report class, and the service classes served by the region will still be reported in the SCLASS report for the service class in which the region is running. In both of these cases, this information will be useful if you are migrating the CICS/IMS region to transaction response time management, and need both a transaction response time benchmark and a list of the service classes the region is serving.
Sample Scenarios

The following scenarios illustrate different configurations and how they would benefit from these options:

Scenario 1

In this scenario, you wish to assign storage protection and/or CPU protection to address spaces. Suppose you have the following classification rules:

![Diagram showing address spaces HAMLET1, HAMLET2, and HAMLET3 with their respective service classes and storage protection settings.]

Figure 23. Scenario 1: Address Spaces

In this scenario, you wish to assign storage protection and/or CPU protection to address spaces. Suppose you have the following classification rules:

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Fold qualifier names?</th>
<th>IBM-defined subsystem type</th>
</tr>
</thead>
<tbody>
<tr>
<td>JES</td>
<td>Y (Y or N)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>Service</th>
<th>Report</th>
<th>Critical</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TN</td>
<td>HAMLET1</td>
<td>YES</td>
<td>PRODBAT</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>TN</td>
<td>HAMLET*</td>
<td></td>
<td>PRODBAT</td>
<td>NO</td>
<td></td>
</tr>
</tbody>
</table>

Note: Many of the “panels” shown in this section are actually composites of the information displayed after scrolling right from the “Modify Classification Rules” panels.

To assign storage protection to the address space HAMLET1, change the “Storage Critical” value to field to YES in the classification rule for HAMLET1:

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Fold qualifier names?</th>
<th>IBM-defined subsystem type</th>
</tr>
</thead>
<tbody>
<tr>
<td>JES</td>
<td>Y (Y or N)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>Service</th>
<th>Report</th>
<th>Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TN</td>
<td>HAMLET1</td>
<td></td>
<td>PRODBAT</td>
<td></td>
<td>YES</td>
</tr>
<tr>
<td>1</td>
<td>TN</td>
<td>HAMLET*</td>
<td></td>
<td>PRODBAT</td>
<td></td>
<td>NO</td>
</tr>
</tbody>
</table>
To assign storage protection to the address spaces HAMLET2 and HAMLET3 (but not HAMLET1), change the “Storage Critical” value to field to YES in the wild card rule for HAMLET*:

<table>
<thead>
<tr>
<th>Subsystem Type .: JES</th>
<th>Fold qualifier names? Y (Y or N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description . . . . .</td>
<td>IBM-defined subsystem type</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Qualifier</th>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>Service</th>
<th>Report</th>
<th>Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>HAMLET1</td>
<td>TN</td>
<td>___</td>
<td>PRODBAT</td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HAMLET*</td>
<td>TN</td>
<td>___</td>
<td>PRODBAT</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note that in this instance, HAMLET1 will *not* be protected, as it will match the HAMLET1 rule first. To protect *all* of the address spaces, you would have to specify YES in both rules.

The default values (with no CPU protection assigned) in the service class definitions would be:

<table>
<thead>
<tr>
<th>Service Class Name . . . . : PRODBAT (Required)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description . . . . . . . . . . . . . . . . . .</td>
</tr>
<tr>
<td>Workload Name . . . . . . . . . . . . . . . . .</td>
</tr>
<tr>
<td>Base Resource Group . . . . . . . . . . . . .</td>
</tr>
<tr>
<td>Cpu Critical . . . . . . . . . . . . . . . . .</td>
</tr>
</tbody>
</table>

To assign CPU protection to all of these address spaces, change the value in the “Cpu Critical” field in the service class definition for PRODBAT:

<table>
<thead>
<tr>
<th>Service Class Name . . . . : PRODBAT (Required)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description . . . . . . . . . . . . . . . . . .</td>
</tr>
<tr>
<td>Workload Name . . . . . . . . . . . . . . . . .</td>
</tr>
<tr>
<td>Base Resource Group . . . . . . . . . . . . .</td>
</tr>
<tr>
<td>Cpu Critical . . . . . . . . . . . . . . . . .</td>
</tr>
</tbody>
</table>
Scenario 2

Suppose you have the following CICS classification rules:

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Fold qualifier names?</th>
<th>IBM-defined subsystem type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICS</td>
<td>Y (Y or N)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Description</th>
<th>--------------</th>
<th>--------------</th>
<th>--------------</th>
<th>--------------</th>
<th>--------------</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Type</td>
<td>Name</td>
<td>Start</td>
<td>Service</td>
<td>Report</td>
</tr>
<tr>
<td>1</td>
<td>TN</td>
<td>AA1</td>
<td>___</td>
<td>TRXAA</td>
<td>AA1RPT</td>
</tr>
<tr>
<td>1</td>
<td>TN</td>
<td>AA2</td>
<td>___</td>
<td>TRXAA</td>
<td>AA2RPT</td>
</tr>
<tr>
<td>1</td>
<td>TN</td>
<td>BB</td>
<td>___</td>
<td>TRXBB</td>
<td>______</td>
</tr>
<tr>
<td>1</td>
<td>TN</td>
<td>CC</td>
<td>___</td>
<td>TRXCC</td>
<td>______</td>
</tr>
</tbody>
</table>

Suppose you have the following STC classification rules to classify the CICS regions:

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Fold qualifier names?</th>
<th>IBM-defined subsystem type</th>
</tr>
</thead>
<tbody>
<tr>
<td>STC</td>
<td>Y (Y or N)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Description</th>
<th>--------------</th>
<th>--------------</th>
<th>--------------</th>
<th>--------------</th>
<th>--------------</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Type</td>
<td>Name</td>
<td>Start</td>
<td>Service</td>
<td>Report</td>
</tr>
<tr>
<td>1</td>
<td>TN</td>
<td>CICSREGP</td>
<td>___</td>
<td>PRODRGNS</td>
<td>______</td>
</tr>
<tr>
<td>1</td>
<td>TN</td>
<td>CICSREGT</td>
<td>___</td>
<td>TESTRGNS</td>
<td>______</td>
</tr>
</tbody>
</table>

Suppose the transaction service classes are defined as follows. (Only TRXAA is shown here. The definitions for TRXBB and TRXCC would also have NO specified in the “Cpu Critical” field.)
Suppose the regions’ service classes are defined as follows (PRODRGNS shown here, TESTRGN would look the same):

In this scenario, assume that the regions are running normal, non-conversational transactions. Response time goals are appropriate, and there is enough activity so that WLM can manage the regions as servers virtually all of the time. Transaction AA1 is very important to the business, and you wish to give it both storage and CPU protection.

In this case, protection on a transaction service class level is sufficient. This approach allows you to focus on protecting specific transactions rather than the regions that process them. The protection will be inherited by any regions in which the transactions run, as long as WLM is allowed to manage the region to the transactions’ goals. (“Scenario 5” on page 108 shows what happens when WLM is not allowed to manage the region to the transactions’ goals.)

Assign storage protection to transaction AA1 using the CICS classification rule:

Transaction service class TRXAA runs only in the CICSREGP region, and not in CICSREGT; therefore, CICSREGP will inherit the storage protection, and CICSREGT will not inherit the storage protection.

Assign CPU protection to the transaction service class TRXAA in the service class definition:

Chapter 12. Defining Special Protection Options for Critical Work 105
Any region serving any TRXAA transactions, even one serving AA2 only, inherits CPU protection. As was true for storage protection, CICSREGT will not inherit CPU protection because it does not serve transaction service class TRXAA.

Reporting products which display data about the regions themselves will not show that storage and CPU protection was specified, but will show that they were protected while serving the transactions. (See “Reporting” on page 110.) Service class reports will show the storage and CPU protection assigned to the TRXAA transaction service class.

**Scenario 3**

In this scenario, again using the CICS regions shown in Figure 24 on page 104, assume that the regions are running non-conversational transactions, but with periods of low activity during which WLM may stop managing them as servers. During this time, it is more likely that the regions’ pages will be stolen by competing workloads. In this scenario, assume that transaction BB needs storage and CPU protection.

Protection on a transaction service class level is once again useful, ensuring that the transactions will be protected wherever they run. The regions themselves should also be protected, as WLM may not manage them as servers during the low activity periods.

Assign storage protection to transaction BB using the CICS classification rules:

<table>
<thead>
<tr>
<th>Subsystem Type : CICS</th>
<th>Fold qualifier names?</th>
<th>Y (Y or N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description : IBM-defined subsystem type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Action</td>
<td>Type</td>
<td>Name</td>
</tr>
<tr>
<td>____</td>
<td>1 TN</td>
<td>AA1</td>
</tr>
<tr>
<td>____</td>
<td>1 TN</td>
<td>AA2</td>
</tr>
<tr>
<td>____</td>
<td>1 TN</td>
<td>BB</td>
</tr>
<tr>
<td>____</td>
<td>1 TN</td>
<td>CC</td>
</tr>
</tbody>
</table>

Also, assign storage protection to the regions themselves using the STC classification rules:

<table>
<thead>
<tr>
<th>Subsystem Type : STC</th>
<th>Fold qualifier names?</th>
<th>Y (Y or N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description : IBM-defined subsystem type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Action</td>
<td>Type</td>
<td>Name</td>
</tr>
<tr>
<td>____</td>
<td>1 TN</td>
<td>CICSREGP</td>
</tr>
<tr>
<td>____</td>
<td>1 TN</td>
<td>CICSREGT</td>
</tr>
</tbody>
</table>

Assign CPU protection to the transaction service classes TRXBB (the “Cpu Critical” field in the TRXAA and TRXCC service class definitions would remain NO):
And also to the regions themselves (PRODRGNS shown here, TESTRGNS would also specify YES in the “Cpu Critical” field):

<table>
<thead>
<tr>
<th>Service Class Name</th>
<th>PRODRGNS (Required)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>CICS Regions</td>
</tr>
<tr>
<td>Workload Name</td>
<td>STC (name or ?)</td>
</tr>
<tr>
<td>Base Resource Group</td>
<td>(name or ?)</td>
</tr>
<tr>
<td>Cpu Critical</td>
<td>YES (YES or NO)</td>
</tr>
</tbody>
</table>

Note that since both CICSREGP and CICSREGT run transaction BB, both regions must be protected.

Reporting products which display data about the regions will show that both CPU and storage protection was specified. (See “Reporting” on page 110.) While the regions are serving transactions, protection will occur if either the regions themselves or any of their served transaction service classes are protected.

**Scenario 4**

In this scenario, again using the CICS regions shown in Figure 24 on page 104, assume that the regions are running conversational transactions, and response time goals are not appropriate. By exempling the regions from management to the transaction response time goals, the regions will instead be managed according to the goal of the service class assigned to those regions. (If either storage or CPU protection is needed, that goal must be a velocity goal, since discretionary goals are not eligible for storage or CPU protection.) In this scenario, assume that only the production region CICSREGP needs protection.

Assign storage protection to the CICSREGP region. Also, in the same panel, exempt both regions from management to the transaction response time goals:

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>STC</th>
<th>Fold qualifier names?</th>
<th>Y (Y or N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>IBM-defined subsystem type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Action Type</td>
<td>CICSREGP</td>
<td>PRODRGNS</td>
<td>YES</td>
</tr>
<tr>
<td>Action Type</td>
<td>TESTRGNS</td>
<td>YES</td>
<td>NO</td>
</tr>
</tbody>
</table>

Assign CPU protection to the PRODRGNS service class (the “Cpu Critical” field in the TESTRGNS service class definitions would remain NO):

<table>
<thead>
<tr>
<th>Service Class Name</th>
<th>PRODRGNS (Required)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>CICS Regions</td>
</tr>
<tr>
<td>Workload Name</td>
<td>STC (name or ?)</td>
</tr>
<tr>
<td>Base Resource Group</td>
<td>(name or ?)</td>
</tr>
<tr>
<td>Cpu Critical</td>
<td>YES (YES or NO)</td>
</tr>
</tbody>
</table>

Reporting products which display data about the regions will show that CPU or storage protection was specified based on the regions’ storage protection value and the CPU protection value of the regions’ service classes. (See “Reporting” on page 110.)
Scenario 5

This scenario is similar to “Scenario 2” on page 104, but here you’ll see what happens when WLM is not allowed to manage one of the regions to the transactions’ goals, and how this will prevent protection of a transaction. In this case, assume that it is transaction BB that you wish to give both storage and CPU protection, and assume that at the same time you have exempted region CICSREGT from management to the transaction response time goals.

You’ve assigned the storage protection to transaction BB using the CICS classification rule:

```
Subsystem Type. : CICS  Fold qualifier names? Y (Y or N)
Description . . . IBM-defined subsystem type
Action Type Name Start Service Report Storage Critical
____ 1 TN AA1 ___ TRXAA AA1RPT NO
____ 1 TN AA2 ___ TRXAA AA2RPT NO
____ 1 TN BB ___ TRXBB ______ YES
____ 1 TN CC ___ TRXCC ______ NO
```

You’ve also assigned CPU protection to the transaction service class TRXBB in the service class definition:

```
Service Class Name . . . . TRXBB (Required)
Description . . . . . . . . CICS Transactions
Workload Name . . . . . . CICS (name or ?)
Base Resource Group . . . (name or ?)
Cpu Critical . . . . . . . YES (YES or NO)
```

In the classification rule for CICSREGT, you have exempted the region from management to the transaction response time goals:

```
Subsystem Type . : STC  Fold qualifier names? Y (Y or N)
Description . . . IBM-defined subsystem type
Action Type Name Start Service Report Storage Critical Manage Region Using Goals Of:
____ 1 TN CICSREGP ___ PRODRGNS ___ YES TRANSACTION
____ 1 TN CICSREGT ___ TESTRGSN ___ YES REGION
```

As illustrated in Figure 24 on page 104, transaction BB runs in both regions, CICSREGP and CICSREGT. Since WLM will not manage region CICSREGT to transaction response times, it will not inherit storage or CPU protection from the TRXBB transaction service class. Transaction BB will therefore not run with storage or CPU protection in region CICSREGT.

Scenario 6

In this scenario, assume that the CICS regions adhere to a Work Manager/Consumer model. This means that the TOR acts as a work receiver of requests from the work originator and as a sender of results back to the work originator. The TOR just distributes the work to consumer processes in the AORs which start application programs to perform the function behind the work requests.
Suppose you have the following STC classification rules to classify the CICS regions:

The TOR typically only requires short access to resources, but it also needs very fast access in order to avoid being a bottleneck. The AORs typically run more resource intensive programs which do not require the same fast access to the resources.

In this setup, when contention increases on the system, WLM has no way to give faster access to TORs than to AORs, if no other workload like batch work with lower service goals exists that can be demoted. At higher utilization levels, typically above 85%, a noticeable queue delay within the TORs can be recognized. This reduces the end-to-end response times of the CICS transactions and the throughput of the CICS work.

One could avoid this problem by exempting the CICS regions from being managed towards response time goals:
The negative effect of this solution is that CICS transaction statistics are no longer available for managing the CICS work and only through classes for reporting purposes. This workaround in fact reduces WLM’s capabilities to a reporting only function for CICS.

The best solution to solve this issue is to use option BOTH for the TOR:

```plaintext
Subsystem Type : STC    Fold qualifier names? Y (Y or N)
Description . . . IBM-defined subsystem type

<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>Service</th>
<th>Report</th>
<th>Critical</th>
<th>Using Goals Of</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TN</td>
<td>TOR</td>
<td>VEL80I1</td>
<td>NO</td>
<td>REGION</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>TN</td>
<td>AOR*</td>
<td>VEL60I2</td>
<td>NO</td>
<td>REGION</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

This option allows WLM to manage a region by the goals of the regions but also ensures that the region tracks all transaction completions correctly so that WLM can still manage the CICS service classes with response time goals. The new option should only be used for CICS TORs. All AORs should remain at the default (TRANSACTION). In addition the service class for the CICS TORs should be defined with a higher importance than the service classes for the CICS transactions.

The result is that the CICS TORs now can access the resources they need as fast as possible. Because TORs consume typically only between 5 to 10% of the total CICS resource consumption the goals of the CICS service classes still manage the important parts of the CICS workloads and ensure that work is managed towards response times.

**Reporting**

Because storage protection can be implicitly applied to an entire transaction service class, and because WLM may or may not be honoring a customer’s storage or CPU protection assignment at any given time (for example, due to a RESET), there are seven different “states” that can be reported:

- Storage protection has been explicitly assigned on a classification rule
- Storage protection has been implicitly assigned to a CICS/IMS transaction service class (because it was assigned to at least one transaction in that service class)
- Storage protection is currently being honored
- CPU protection has been explicitly assigned in a service class definition
- CPU protection is currently being honored
- Exemption from transaction response time management has been explicitly assigned on a classification rule.
- Management of a region using the goal of both region and transaction service classes has been explicitly assigned on a classification rule.
These states are reported in SMF type 30 and type 79.1 records. States that apply to an entire service class are also reported in SMF 72.3 records.

**Option Summary**

The following table summarizes the effects of the storage protection, CPU protection, and exemption from transaction response time management options:

*Table 11. Summary of Storage Protection, CPU Protection, and Exemption from Transaction Response Time Management Options*

<table>
<thead>
<tr>
<th>When you...</th>
<th>WLM...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assign CPU protection to a service class used to manage address spaces and/or enclaves.</td>
<td>Protects any address space or enclave managed according to the goals of that service class. Address spaces being managed as servers are managed according to the goals of the served transactions.</td>
</tr>
<tr>
<td>Assign storage protection to an ASCH, JES, OMVS, STC, or TSO address space.</td>
<td>Protects any address space which matches the classification rule, regardless of its server status. Address spaces currently running in multiperiod service classes or in service classes with a short response time goal (20 seconds or less) are excluded from protection.</td>
</tr>
<tr>
<td>Assign CPU or storage protection to a CICS or IMS transaction.</td>
<td>Protects any regions recognized as serving that CICS/IMS transaction, unless you prevent WLM from managing the regions as servers. Note that once storage protection is assigned to any transaction in a service class, then all transactions in the same service class become storage protected.</td>
</tr>
<tr>
<td>Manage a CICS or IMS region using the goals of the region.</td>
<td>Is prevented from managing the region according to the response time goals of the transactions it is running. It does not recognize the region as a server. The region is managed using the goal of the service class assigned to the region. Transaction response time data is not reported in the service classes to which the transactions are classified, but is still reported in their report classes, if assigned.</td>
</tr>
<tr>
<td>Manage a CICS or IMS region using the goals of both region and transaction.</td>
<td>Manages the region using the goal of the service class assigned to the region. This also ensures that the region tracks all transaction completions correctly so that it can still manage the CICS service classes with response time goals. The option should only be used for CICS TORs. All AORs should remain at the default (TRANSACTION). In addition, the service class for the CICS TORs should be defined with a higher importance than the service class for the CICS transactions.</td>
</tr>
<tr>
<td>Issue the RESET QUIESCE command.</td>
<td>Will no longer enforce CPU protection. All other options remain unchanged.</td>
</tr>
<tr>
<td>Issue the RESET SRVCLASS= or RESET RESUME command.</td>
<td>Will assign CPU protection if the target service class has the CPU protection attribute. All other options remain unchanged.</td>
</tr>
</tbody>
</table>
Chapter 13. Defining Application Environments

An application environment is a group of application functions requested by a client that execute in server address spaces. Workload management can dynamically manage the number of server address spaces to meet the performance goals of the work making the requests. Alternatively, the server address spaces can be started and stopped manually or by automation.

Each application environment should represent a named group of server functions that require access to the same application libraries. Grouping server functions helps simplify library security, application program change control, performance management, and system operation.

For example, an application environment could be one or more DB2 stored procedures. DB2 could have an associated application environment named PAYROLL that handles specific types of stored procedure requests.

Getting Started with Application Environments

The following conditions are required before an application environment can be used:

- The work manager subsystem must have implemented the workload management services that make use of application environments. Examples of IBM-supplied work managers that use application environments are:
  - DB2 (subsystem type DB2)
  - SOMobjects (subsystem type SOM)
  - WebSphere Application Server (subsystem CB)
  - Internet Connection Server, Domino Go Webserver, or IBM http Server for z/OS (subsystem type IWEB)
  - MQSeries Workflow (subsystem type MQ)

Refer to subsystem documentation to determine if the subsystems used in your installation make use of application environments.

- One or more application environments must be defined in the workload management service definition. The subsystem reference information should provide guidance for logically grouping applications into application environments.

- The subsystem’s work requests must be associated with the appropriate application environment. This step is unique for each subsystem and should be described in the subsystem reference information.

If you request through the service definition that server address spaces be automatically managed, workload management starts and stops server spaces as needed. For example, when a DB2 stored procedure request comes into the system, workload management determines whether there is a server address space to process the work, and if there is, makes the work available to the server. If there is no server address space available, workload management creates one.

Table 12 on page 114 shows the IBM-supplied subsystems that use application environments, the types of requests made by each subsystem, and where the subsystem stores the information that maps the work to application environments.

© Copyright IBM Corp. 1994, 2011
### Table 12. IBM-supplied Subsystems Using Application Environments

<table>
<thead>
<tr>
<th>Application Environment Subsystem Type</th>
<th>Request Type</th>
<th>Application Environment Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>CB</td>
<td>WebSphere Application Server object method requests</td>
<td>Server group name</td>
</tr>
<tr>
<td>DB2</td>
<td>Stored procedure requests</td>
<td>DB2 SYSIBM.SYSROUTINES catalog table</td>
</tr>
<tr>
<td>IWEB</td>
<td>Hyper Text Transfer Protocol (HTTP) requests</td>
<td>Web configuration file</td>
</tr>
<tr>
<td>MQ</td>
<td>MQSeries Workflow requests</td>
<td>MQ process definition for the WLM-managed queue (APPLICID) field</td>
</tr>
<tr>
<td>SOM</td>
<td>SOM client object class binding requests</td>
<td>Implementation repository</td>
</tr>
</tbody>
</table>

### Specifying Application Environments to Workload Management

**Note:** For applications exploiting the service for defining application environments (IWM4AEDF), you may not need to define the application environment manually. For further information, refer to the appropriate product documentation.

To define an application environment, specify:
- The subsystem type under which the applications are running.
- The JCL procedure to start server address spaces if you wish workload management to automatically manage the number of servers.
- If a JCL procedure is specified, any required start parameters.
- Whether requests can execute in multiple server address spaces and on multiple systems.

### Defining Application Environments

**Application Environment**
- Application environment name.

**Description (optional)**
- Description of the application environment.

**Subsystem Type**
- The subsystem type associated with the application environment, such as SOM, DB2, or IWEB.

**Procedure Name (optional)**
- The name of the JCL procedure for starting the address space. The procedure name may be omitted.

**Start Parameters (optional)**
- Parameters required by the JCL procedure to start the address space.

**Starting of address spaces**
- Specify whether workload management can start multiple or single server address spaces for a subsystem instance.
Application Environment (required)
One to 32 character name of the application environment. You must use this name when specifying to the subsystem how to map work to the application environments. You also use this name in operator commands when performing actions on the application environment. The name cannot begin with the letters SYS.

For guidance in mapping subsystem work to application environments, see the subsystem reference information.

If you check the authorization of application environment server address spaces through a SAF product, such as RACF, or if you plan to do so, choose an application environment name that does not exceed 27 characters. There is a restriction with respect to the maximum length of a server profile name passed to the SAF product (see also "Authorizing Application Environment Servers" on page 123).

Description (optional)
Up to 32 characters describing the application environment.

Subsystem Type (required)
Subsystem type is the one to four character name of the subsystem using application environments. This subsystem type is provided to workload management when the subsystem initializes. The types currently in use are listed in Table 12 on page 114. For a subsystem not listed, refer to the subsystem's documentation for the required information.

Note: If you are using DB2 stored procedures, note that the subsystem type DB2 specified here for an application environment is used only for identifying the DB2 subsystem when it begins to use the application environment. There is no connection between this value and classification. For more information on classification, see Chapter 9, "Defining Classification Rules," on page 57.

Procedure Name (optional)
Procedure name is the one to eight character name of the JCL procedure that workload management uses to start a server for the application environment work requests. Refer to the appropriate subsystem documentation for sample JCL procedures to use.

To ensure that an application environment uses the same JCL procedure across the sysplex, either (1) identical procedure proclibs must be maintained across the sysplex or (2) all the procedures must be stored in a single, shared proclib.

If you specify a procedure name, “automatic” control is in effect, and workload management manages the number of servers. If you do not specify a procedure name, manual control is in effect, and servers must be started manually or by automation. In either case, workload management processes work requests from the application environment according to the goals defined for the work once a server address space is available.

Start Parameters (optional)
Start parameters are the parameters required for the JCL procedure defined in Procedure Name. These parameters define how workload management should start the server address spaces. Specify parameters here that you would use for starting a server address space with an MVS START command.

Note that any parameters you specify here override the parameters specified in the JCL procedure for that server.
If you specify the symbol &IWMSSNM, workload management substitutes the subsystem instance name provided to workload management when the subsystem connected to it. Refer to the subsystem reference information to determine the instance name and the appropriate parameters to specify.

**Starting of server address spaces for a subsystem instance (required)**

You can limit the number of servers for a subsystem instance. Reasons for limiting the number of servers might be a need to serialize or limit application environment server activity while testing, or a restriction in the subsystem itself. There are three options:

- Managed by WLM
- Limited to a single server address space per system
- Limited to a single server address space per sysplex

The options that are valid for a subsystem depend on its scope as described in the next section. For guidance on deciding which options to use, and to find out which options are valid for subsystems not explicitly covered in the next section, see the subsystem reference information.

---

**Example of an Application Environment**

To enable workload management to dynamically start server address spaces to process work from a DB2 payroll application environment, do the following:

- Define the application environment

  **Application Environment**
  PAYROLL

  **Description**
  DB2 Payroll APPLENV

  **Subsystem Type**
  DB2

  **Procedure Name**
  PAYPROC

  **Start Parameters**
  DB2SSN=&IWMSSNM

  **Server Start**
  Managed by WLM

When payroll work arrives into the system, workload management manages system resources to meet the goals defined for the work, and dynamically starts and stops server address spaces to process the work.

---

**Selecting Server Limits for Application Environments**

The previous section describes an option how servers of an application environment can be started. You can allow WLM to manage the number of servers or you can limit them. This option is applied independently for each instance of a subsystem. For IBM-supplied subsystems, you can usually use the default value supplied by the WLM ISPF application. This section defines subsystem instance as it is used by application environments and tells how to select the server start option for IBM-supplied subsystems.
A **subsystem instance** for an application environment is defined as a unique combination of:

- Subsystem type, as specified in the service definition for an application environment, and
- Subsystem name, as defined by the work manager subsystem when it connects to workload management.

A subsystem instance using application environments has one of two different scopes, single-system or sysplex, depending on workload management services used by its subsystem type.

If the scope is single-system, all the server address spaces for the subsystem instance are created on the system where the instance connected to workload management.

If the scope is sysplex, the server spaces can be spread across the sysplex, with workload management starting at most one server on each system. The installation may choose to start additional servers on a system through the START command or automation, and these servers are equally eligible to accept application environment work as the one started automatically.

Note that this scope applies only to application environment server management. A subsystem with single-system scope for application environments, can still perform sysplex-wide functions for other purposes.

You can specify how the number of servers for a subsystem instance will be started when defining the application environment in the service definition. Reasons for limiting the number of servers might be a need to serialize or limit application environment server activity while testing, or a restriction in the subsystem itself.

There are three options how to start servers:

- Managed by WLM
- Limited to a single server address space per system
- Limited to a single server address space per sysplex

Options 1 and 2 apply when the subsystem type supports single-system scope.

Options 1 and 3 apply when the subsystem type supports sysplex scope.

“Managed by WLM” for single-system scope means any number of servers may be created for the subsystem instance on the system where it connected to workload management. “Managed by WLM” for sysplex scope means servers may be created for the subsystem instance on any number of systems in the sysplex.

The IBM-supplied subsystems using application environments, their scopes, and valid server limit options are as follows:
### Table 13. Application Environment Server Characteristics

<table>
<thead>
<tr>
<th>Subsystem Type</th>
<th>Scope</th>
<th>Valid server limit options</th>
</tr>
</thead>
</table>
| CB             | system| 1.) Managed by WLM  
Use this to address the WLM managed servers. This is the default for the WebSphere Application Server.  
2.) Limited to a single server address space per system  
Use this if you are testing and want to temporarily limit the number of servers. The WebSphere Application Server itself does not have a requirement for limiting servers. |
| DB2            | system| 1.) Managed by WLM  
Use this to address the WLM managed servers. This is the default for DB2.  
2.) Limited to a single server address space per system  
Use this if the DB2 stored procedure cannot execute concurrently in multiple address spaces. This option should be used if a stored procedure is to be run in “debug” mode and writes to a trace. |
| IWEB           | system| 1.) Managed by WLM  
Use this to address the WLM managed servers. This is the default for Internet Connection Server, Domino Go Webserver, or IBM http Server for z/OS.  
2.) Limited to a single server address space per system  
Use this if you are testing and want to temporarily limit the number of servers. The Internet Connection Server, Domino Go Webserver, or IBM http Server for z/OS does not have a requirement for limiting servers. |
| MQ             | system| 1.) Managed by WLM  
Use this to address the WLM managed servers. This is the default for MQSeries Workflow.  
2.) Limited to a single server address space per system  
Use this if you are testing and want to temporarily limit the number of servers. MQSeries Workflow itself does not have a requirement for limiting servers. |
| SOM            | sysplex| 3.) Limited to a single server address space per sysplex  
SOMobjects requires this limit on the number of servers for an application environment. This option is enforced by the WLM application. |

For guidance on deciding which option to use, or to find out what options are valid for other subsystems that use application environments, refer to the subsystem reference information.
How WLM Manages Servers for an Application Environment

You allow WLM to manage the number of servers and server address spaces for an application environment if you choose the option “Managed by WLM” for starting servers of a subsystem instance. In this case, WLM has the ability to consider the delays of work requests sent to the application environments in its algorithms to supervise the goal achievement of your service classes and to adjust the resources needed by them. Servers and server address spaces are considered resources which can be made available to the work using an application environment. This is illustrated in the following example:

1. If a DB2 stored procedure request comes into the system, DB2 classifies the work request to WLM and WLM assigns a service class to it. DB2 then queues the work request to the application environment under which the stored procedure should be executed.

2. WLM queues the work requests for each application environment by service class. This allows WLM to understand how the queuing affects the goal achievement of the service class.
   - When the first request is queued to an application environment, workload manager detects that there are no active servers for the request, and automatically starts one.

3. From then on, WLM collects statistics about the queue delays for each application environment and each service class used for the work requests. These queue delays then become part of the WLM algorithms which assess the goal achievement of the service classes and adjust the resources as needed.
   - If, for example, the service class for the DB2 stored procedure requests do not meet their goals, WLM determines which resources are needed to help the work in the service class.
   - If queue delay shows up as the dominating factor, WLM assesses how many additional servers are needed to help the work to meet their goals or at least to see a significant improvement.
   - Once the number of servers has been assessed, WLM makes sure that the system resources are available to start the necessary server address spaces for the required number of servers. During this step, WLM makes sure that more important work is not affected and that the system does not run into a shortage situation because of the new server address spaces. Among the resources considered are:
     - Processor
     - Real storage
     - Auxiliary storage constraints
     - Common storage (SQA) constraints

4. When all tests have completed successfully, WLM starts the required number of server address spaces.

In cases where the system has low utilization, WLM is also able to start one additional server address space for the application environment if this may immediately help the work and it is granted that sufficient resources remain available for other work in the system.

Server address spaces are stopped when the utilization of the servers drops and many servers become idle. WLM then returns the resources used by the server address spaces and thus allows other work to utilize the system.
Using “Managed by WLM” is the optimal way to manage the number of server address spaces. It provides the best performance for the work executed under the application environment and only uses the resources which are optimal based on the goal for the work and the overall utilization of the system.

### Using Application Environments

Application environments can be manually controlled by the installation or automatically controlled by workload management. Note, however, that Dynamic Application Environments can only be automatically controlled by workload management. All of the applications in an application environment are supported by a single JCL startup procedure. Defining the name of this startup JCL procedure to workload management indicates that workload management should control the server address spaces. This is called “automatic” control. If you omit the name of the JCL procedure in the application environment definition, then “manual” control is in effect.

Under “manual” control, the installation must create and delete, as needed, the server address spaces for each application environment. Note that the VARY WLM,APPLENV command can be used to terminate manually started server address spaces (through the quiesce or refresh options), but it will not restart them. For more information on the VARY WLM,APPLENV command, see “Using Operator Commands for Application Environments” on page 121.

Under “automatic” control, workload management creates server address spaces as started tasks using the JCL procedure specified in the application environment definitions. The startup parameters may be contained in either the JCL procedure defined for each application environment or in the application environment definition. When the server address spaces are no longer needed, workload management deletes them.

Under “automatic” control, the quantity of server address spaces is totally controlled by workload management. If an operator or automation starts or cancels the server address spaces under “automatic” control, workload management will:

- Use servers not started by workload management as if they were started by workload management
- Terminate servers not started by workload management if they are not needed
- Replace a server address space that was unexpectedly cancelled

**Note:** You should use the VARY WLM,APPLENV, or VARY WLM,DYNAPPL command to manage application environment servers rather than the CANCEL command. If there are more than five server cancellations in 10 minutes, workload management stops creating new servers for the application environment. For more information on “stop” conditions, see “Handling Error Conditions in Application Environments” on page 122.

### Managing Application Environments

Once an application environment is defined, and there are server address spaces in use by the subsystem, you can use operator commands to manage the application environment. There are options on the VARY WLM,APPLENV, or VARY WLM,DYNAPPL command that allow you to quiesce, resume, or refresh application environments. These functions allow you, for example, to make changes to the JCL procedure, start parameters, or application libraries, and ensure that new work requests run with the modified information. The resume function
also allows you to recover from error conditions that have caused workload management to stop an application environment.

An action taken for an application environment is saved in the WLM couple data set and is not discarded across an IPL. For example, if a quiesce action is in effect and the system is IPLed, the quiesce action remains in effect. You can query the current state of an application environment using the DISPLAY WLM,APPLENV, or DISPLAY WLM,DYNAPPLENV command. The scope of both the VARY and DISPLAY commands for application environments is sysplex-wide, that is, they affect the application environment on all systems in the sysplex, regardless of the scope of the subsystem using the application environment. The sysplex scope of the command ensures that an application environment remains consistent across the sysplex, especially where there are shared resources.

This section first introduces the commands that can be used to perform actions on an application environment. Then it describes activities that make use of the commands and describes other conditions that affect the state of an application environment.

**Using Operator Commands for Application Environments**

An application environment initially enters the AVAILABLE state when the service policy that contains its definition is activated. AVAILABLE means the application environment is available for use, and servers are allowed to be started for it. There are three options on the VARY command that you can use to change the state of an application environment after it has been made available:

- **VARY WLM,APPLENV=xxxx,QUIESCE** or
  **VARY WLM,DYNAPPL=xxxx,QUIESCE**

  The quiesce option causes workload management to request the termination of server address spaces for the application environment upon completion of any active requests. Additional work requests are not handled by the servers, although work requests can continue to be queued, waiting for a server. If you do not want work queued, use subsystem functions to stop the queueing.

  You can issue a quiesce action for an application environment that is in the AVAILABLE state. When a quiesce action is issued for an application environment, it first enters the QUIESCING state until all servers have been requested to terminate. It then enters the QUIESCED state.

- **VARY WLM,APPLENV=xxxx,RESUME** or
  **VARY WLM,DYNAPPL=xxxx,RESUME**

  The resume option restarts an application environment that was previously quiesced and is in the QUIESCED state. It indicates to workload management that server address spaces can once again be started for this application environment. The new servers process any queued requests and all new requests.

  When a resume action is issued for an application environment, it first enters the RESUMING state until all systems in the sysplex have accepted the action. It then enters the AVAILABLE state.

- **VARY WLM,APPLENV=xxxx,REFRESH** or
  **VARY WLM,DYNAPPL=xxxx,REFRESH**

  The refresh option requests the termination of existing server address spaces and starts new ones in their place. Existing servers finish their current work requests and end. The new servers process any queued requests and all new requests.

  You can issue a refresh action for an application environment that is in the AVAILABLE state. When a refresh action is issued for an application environment...
environment, it first enters the REFRESHING state until all servers have been requested to terminate. It then enters the AVAILABLE state.

Making Changes to the Application Environment Servers
The command options described are intended to allow changes to application environments without having to shut down the application itself. Use the quiesce function when you want to do one of the following:

- Perform maintenance on application program libraries statically allocated to server address spaces.
- Update the JCL procedure for an application environment.

When you are ready to put the changes into effect, quiesce the application environment, make the changes to the libraries or service definition as needed, then use the resume function to start new servers with the changed information.

You can also use the quiesce function to suspend execution after repeated application failures or errors. After the errors are corrected, you can resume the application environment.

You may have an application environment where the servers keep application program executable modules in a private cache. If you update the application program, you need to ensure that all copies of the changed modules are replaced wherever they are cached. Use the refresh function to do this.

Changing the Definition of an Application Environment
Workload management initiates a refresh when one of the following changes are made to the application environment definition and activated:

- The JCL procedure name is changed.
- The application environment is switched to “automatic” control, that is, the JCL procedure name was previously left blank, but now one is provided.
- The server start parameters are changed.
- The limit on server address spaces is changed, for example, from “Managed by WLM” to “Limited to a single address space per system”.

If an application environment is deleted from the service definition, it enters the DELETING state. After workload management requests the termination of all associated servers, the application environment is no longer displayed at all by the DISPLAY WLM,APPLENV, or DISPLAY WLM,DYNAPPLENV command.

Handling Error Conditions in Application Environments
Workload management stops the creation of new server address spaces when one of the following conditions exist:

- JCL errors in the procedure associated with the application environment.
- Coding errors in the server code which cause five unexpected terminations of server address spaces within ten minutes.
- Failure of the server address space to connect to workload management due to an invalid invocation environment or invalid parameters.

The application environment first enters the STOPPING state, then the STOPPED state after all systems in the sysplex have accepted the action. In STOPPED state, no new servers are created. Any existing server address spaces continue to process work, and workload management is able to accept new work. If there are no existing servers, then workload management rejects any new work requests.
In STOPPED state, you can make changes to libraries, change the procedure, or make any other changes needed to repair the condition that caused workload management to stop the application environment. When the problem is resolved, use the resume function to allow workload management to start new servers. The application environment enters the RESUMING state, then the AVAILABLE state after all systems in the sysplex have accepted the action.

Note: If you want to ensure all servers are restarted after a STOPPED state, especially after the JCL procedure or libraries have been modified, you should issue a quiesce function prior to the resume. This ensures there are no servers remaining active that are using back-level information.

Authorizing Application Environment Servers

Because the server address spaces started on behalf of an application environment can run in problem program state, workload management enables you to check the validity of a server through an SAF product such as RACF. When the server is being created, workload management makes an SAF call using a new SERVER class to check whether the server is valid for the application environment. If you do not have the SERVER class defined to your SAF product, workload management allows the server address space to be started.

You can use the SERVER and STARTED classes with a SAF product to restrict access to application environment servers. For example, if you are using DB2, SOM or IWEB servers with application environments, you first associate a userid with the MVS procedure name being used to start the server. This is done using the STARTED resource class or by changing ICHRIN03 (started procedures table). Then you use the SERVER resource class to authorize this userid, and possibly others, to become a server for DB2 stored procedures, SOM method requests, or Internet Connection Server web requests.

Example for Restricting Access to Application Environment Servers

In this example, the installation has the following situation:

- MVS JCL procedures for DB2 stored procedure servers: PAY1, PAY2, PER1, PER2
  These are the JCL procedures that workload management uses to start the DB2 servers that handle stored procedure calls.
- DB2 subsystem names: DB2A and DB2B
  These are the subsystem names used when the DB2 subsystem connects to workload management.

1. Activate STARTED and SERVER classes (if not already done):
   ```
   SETR CLASSACT(STARTED) RACLIST(STARTED) GENERIC(STARTED)
   SETR CLASSACT(SERVER) RACLIST(SERVER) GENERIC(SERVER)
   ```

2. Establish an arbitrary USERID to use in a subsequent RDEFINE command to tie an MVS procedure name to a server.
   ```
   ADDUSER DB2SERV NOPASSWORD
   ```
   The NOPASSWORD keyword here is important, it makes DB2SERV a protected user.

3. Associate the userid with the started task name.
   ```
   RDEFINE STARTED PAY*.* STDATA(USER(DB2SERV) GROUP(SYS1))
   RDEFINE STARTED PER*.* STDATA(USER(DB2SERV) GROUP(SYS1))
   ```

4. Define server profiles in the form:
subsys_type.subsys_name.applenv[.subsys_node]

where,

**subsys_type**

is the subsystem type, as specified in the service definition

**subsys_name**

is the instance name of the subsystem associated with this server. Refer to subsystem reference information for how to determine the subsystem name. The subsystem uses this name when establishing itself as the work manager for application environment server requests.

**subsys_node**

is the node name of the server when Work_Manager=Yes is specified. This is an optional parameter.

**applenv**

is the application environment name, as specified in the service definition

RDEFINE SERVER DB2.DB2A.* UACC(NONE)
RDEFINE SERVER DB2.DB2B.* UACC(NONE)

**Note:** The maximum length of a server profile name passed to a SAF product is restricted to 41 characters. WLM cannot start server address spaces for application environments that do not follow this restriction. If you ensure that the applenv is at maximum 27 characters long, it is guaranteed that the maximum server profile name length does not exceed 41 characters.

5. Permit the userid to the servers. This completes the association between the MVS procedure names and the servers:

PERMIT DB2.DB2A.* CLASS(SERVER) ID(DB2SERV) ACCESS(READ)
PERMIT DB2.DB2B.* CLASS(SERVER) ID(DB2SERV) ACCESS(READ)

6. Refresh the classes to refresh the RACF data base and make these changes go into effect:

SETR RACLST(STARTED) REFRESH
SETR RACLST(SERVER) REFRESH
A scheduling environment is a list of resource names along with their required states. It allows you to manage the scheduling of work in an asymmetric sysplex where the systems differ in installed applications, or installed hardware facilities. If an MVS image satisfies all of the requirements in the scheduling environment associated with a given unit of work, then that unit of work can be assigned to that MVS image. If any of the resource requirements are not satisfied, then that unit of work cannot be assigned to that MVS image.

Scheduling environments and resource names reside in the service definition and apply across the entire sysplex. They are sysplex-oriented. Resource states have a different setting in each system in the sysplex and are, therefore, system-oriented.

Each element in a scheduling environment consists of the name of a resource and a required state of either ON or OFF, as follows:

- If the required state is ON, then the resource state must be set to ON on an MVS image for the requirement to be satisfied.
- If the required state is OFF, then the resource state must be set to OFF on an MVS image for the requirement to be satisfied.

In theory, each resource name represents the potential availability of a resource on an MVS system. That resource can be an actual physical entity such as a data base or a peripheral device, or it can be an intangible quality such as a certain time of day or a certain day of the week. The resource names are abstract, and have no inherent meaning.

For instance, you could define a resource name to be XXXX with a required state of ON. If on system SYS1 the corresponding XXXX resource state is set to ON, then the requirement is satisfied. WLM does not care what “XXXX” means, or whether the ON setting really does signify the existence of some real resource. (You could use XXXX as nothing more than an arbitrary toggle switch, setting it ON for whatever reason you wish.) As long as the settings match, the requirement is satisfied.

This information shows how to define the scheduling environments, the resource names, and their required states. It also shows how to set the resource states on each individual MVS system, and how to associate a scheduling environment name with incoming work.

**Getting Started with Scheduling Environments**

The following steps are required to use scheduling environments:

- You must define one or more scheduling environments, and all of the resource names and required states that are listed in those scheduling environments, in the workload management service definition. See “Specifying Scheduling Environments to Workload Management” on page 126.

- For every system in the sysplex on which you want the resource settings to satisfy either ON or OFF requirements, you must set the individual resource states to either ON or OFF, as appropriate. There is also a third setting, RESET, that satisfies neither an ON nor OFF requirement. See “Managing Resource States” on page 127 for more information on the RESET state.
For each unit of work with resource state requirements that is submitted for execution, you must specify the name of the scheduling environment that should be used to determine which systems can execute that work. See "Associating Scheduling Environments with Incoming Work" on page 131.

**Specifying Scheduling Environments to Workload Management**

To define a scheduling environment, you need to specify the following information:

**Scheduling Environment Name (required)**
One to 16 character name of the scheduling environment.

- You can have up to 999 unique scheduling environments defined in a service definition.
- Alphanumerics and the special characters @, $, # and _ are allowed.
- Underscores (_) must be imbedded (for example, PLEX_D01 is valid, but PLEX_ is not).
- Names beginning with SYS_ are reserved for system use.

**Description (optional)**
Up to 32 characters describing the scheduling environment.

Once you have defined a scheduling environment, you can start selecting its resource names and required states, as follows:

**Resource Name (required)**
One to 16 character name of the resource. There can be more than one resource name listed in a scheduling environment.

- You can have up to 999 unique resource names defined in a service definition.
- Alphanumerics and the special characters @, $, # and _ are allowed.
- Underscores (_) must be imbedded (for example, PLEX_D01 is valid, but PLEX_ is not).
- Names beginning with SYS_ are reserved for system use.

**Resource Description (optional)**
Up to 32 characters describing each resource.

When you select a resource name to become part of the scheduling environment, you also need to specify a required state:

**Required State (required)**
For each resource name in a scheduling environment, you must specify a required state of either ON or OFF:

- **ON** specifies that the resource name must be set to ON on a given system for the work associated with this scheduling environment to be assigned to that system.
- **OFF** specifies that the resource name must be set to OFF on a given system for the work associated with this scheduling environment to be assigned to that system.
Scheduling Environment Example

To define a scheduling environment called DB2LATE that contains the following requirements:

- The “DB2A” resource must be set to ON. (In this example, we’ll say that DB2A has been defined to represent the existence of the DB2 subsystem.)
- The “PRIMETIME” resource must be set to OFF. (In this example, we’ll say that PRIMETIME has been defined to be ON during the normal weekday business hours, and OFF for all other times.)

You would define the following scheduling environment:

Scheduling Environment

    DB2LATE

Description

    Offshift DB2 Processing

<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Required State</th>
<th>Resource Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2A</td>
<td>ON</td>
<td>DB2 Subsystem</td>
</tr>
<tr>
<td>PRIMETIME</td>
<td>OFF</td>
<td>Peak Business Hours</td>
</tr>
</tbody>
</table>

Null scheduling environments: If you no longer need to restrict where work executes in a sysplex, you can remove all the resource state requirements from a scheduling environment. A null or empty scheduling environment always allows work to be scheduled; that is, any system in the sysplex is satisfactory for work associated with a null scheduling environment. This is a migration aid when you initially have resources that exist on only some of the systems in a sysplex, but later make the resources available to every system. It saves the effort of having to remove the scheduling environment specification from all the incoming work.

Refer to "Working with Scheduling Environments" on page 208 to see how to use the WLM ISPF application to create and modify scheduling environments.

Managing Resource States

For every resource name that is referenced by a scheduling environment, a corresponding resource state must be set on each system in the sysplex. The resource state can be:

- **ON**, which will satisfy a resource state requirement of ON.
- **OFF**, which will satisfy a resource state requirement of OFF.
- **RESET**, which will not satisfy any resource state requirement. Resources are put into the RESET state when:
  - A system is IPLed
  - A policy is activated that defines a resource name that did not exist in the previously active policy

These resource states can be manipulated in three ways:

- The operator command:

  ```
  F WLM,RESOURCE=resource_name,setting
  ```
where setting can be **ON**, **OFF**, or **RESET**.

For example, to set DB2A to ON on system SYS1, here is the command you would enter **on system SYS1**, along with the response you would receive:

```
F WLM,RESOURCE=DB2A,ON
IWM039I RESOURCE DB2A IS NOW IN THE ON STATE
```

- The equivalent WLM application programming interface IWMSESET. See [z/OS MVS Programming: Workload Management Services](#) for more information on using this interface.
- Using SDSF, you can change resource state settings directly on the panel displaying the current states.

**Note:** Do not attempt to issue the F WLM,RESOURCE= command from the COMMNDxx parmlib member, as this member is processed too early during system initialization. If you want resource states to be set on every system IPL, this needs to be done through an automation product such as System Automation for z/OS as soon as that automation product comes up during system initialization.

It is expected that, in most cases, the mechanics of managing resources states will be handled by installation-provided automation, as opposed to having a human operator issue a modify command every time a resource state is changed. Two examples of how automation could manage resource settings:

- By listening for messages from a subsystem that indicate that subsystem has completed its initialization and is ready to accept work. The automation script could issue the appropriate F WLM,RESOURCE=subsystem,ON command on that system. When messages are issued indicating that the subsystem is about to terminate, the script could issue the appropriate F WLM,RESOURCE=subsystem,OFF command on that system.
- For time-related resource settings, a simple script can turn settings ON and OFF at certain times of the day (like at the beginning and end of peak business hours).

See [z/OS MVS Programming: Workload Management Services](#) for more information on automation of resource states, WLM services, and coordination with other job scheduling programs.

When you modify the resource state settings on a given system, you do so on **that system only**. If you modify the DB2A resource state on system SYS1, it has no affect on the DB2A setting on SYS2. If you wish to modify the settings on both systems, you would have to explicitly direct the commands to each individual system.

When **all** of the resource state settings on a particular system match the resource names and required states defined in a particular scheduling environment, only then is that system eligible to receive work associated with the scheduling environment.
Resource States Example

Using the DB2LATE scheduling environment defined in the first example, here’s how the resource states might be set, and how that would affect the eligibility of work scheduled with DB2LATE to run on each system in the sysplex.

1. The resource names DB2A and PRIMETIME have just been defined (and not set on the individual systems yet) or the systems have just IPLed:

<table>
<thead>
<tr>
<th>Resource State</th>
<th>SYS1 Settings</th>
<th>SYS2 Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2A</td>
<td>RESET</td>
<td>RESET</td>
</tr>
<tr>
<td>PRIMETIME</td>
<td>RESET</td>
<td>RESET</td>
</tr>
</tbody>
</table>

2. Because of the existence of DB2 on SYS1 only, the DB2A resource state is modified to ON on that system and OFF on SYS2. Also, automation has been set up to modify the PRIMETIME setting according to the time of day. At the moment, it is 10:00 a.m. on a Monday morning, so PRIMETIME is set to ON on both systems:

<table>
<thead>
<tr>
<th>Resource State</th>
<th>SYS1 Settings</th>
<th>SYS2 Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2A</td>
<td>ON</td>
<td>OFF</td>
</tr>
<tr>
<td>PRIMETIME</td>
<td>ON</td>
<td>ON</td>
</tr>
</tbody>
</table>

At the moment, the DB2LATE scheduling environment is not satisfied by the resource state settings of either system. Therefore, any work submitted that is associated with the DB2LATE scheduling environment cannot yet be executed on either system.

3. At 5:00 p.m., automation modifies PRIMETIME to OFF on both systems:

<table>
<thead>
<tr>
<th>Resource State</th>
<th>SYS1 Settings</th>
<th>SYS2 Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2A</td>
<td>ON</td>
<td>OFF</td>
</tr>
<tr>
<td>PRIMETIME</td>
<td>OFF</td>
<td>OFF</td>
</tr>
</tbody>
</table>

SYS1 now finally has all of the correct resource state settings to satisfy the DB2LATE scheduling environment. The work submitted associated with the DB2LATE scheduling environment can now be executed on SYS1.

In the previous example, work associated with the DB2LATE scheduling environment was assigned to SYS1, because that system satisfied all of the DB2LATE requirements. In the case of a single-system sysplex, this ability to choose the “right” system is no longer applicable. But scheduling environments may still be useful. Consider if, in the previous example, there was only a SYS1 in the sysplex. Only when the DB2A resource setting was set to ON (signalling that DB2 was up and running) and the PRIMETIME resource setting was set to OFF (signalling that the peak business hours were over) would DB2LATE work be processed on SYS1. The DB2LATE scheduling environment would act as a “ready” flag, holding the work until all of the requirements were met.

Figure 26 on page 130 summarizes the relationship between scheduling environments and the resource state settings on several sample systems.
Note that in Figure 26:

- Work associated with the DB2PRIME scheduling environment can be scheduled only on SYS1, because only that system satisfies both of the requirements.
- Similarly, work associated with the DB2LATE scheduling environment can be scheduled only on SYS2, because only that system satisfies both of the requirements.
- Work associated with the ANYPRIME scheduling environment can be scheduled on either SYS1 or SYS4, because both of those systems satisfy the sole requirement (PRIMETIME must be ON). This scheduling environment does not care about the DB2A setting. Therefore the RESET state for DB2A on SYS4 is irrelevant.
Work associated with the WHEREVER scheduling environment can be scheduled on any system in the sysplex, because it is empty (it has no requirements at all).

**Associating Scheduling Environments with Incoming Work**

Having defined scheduling environments to WLM, and having set the resource states on the individual systems, all you need now is a way to associate the scheduling environment name with actual work.

The SCHENV= parameter on the JES2 or JES3 JCL JOB statement associates the job with the scheduling environment as shown in the following JCL example:

```
//SHAMIL JOB (C003,6363),'Steve Hamilton',
  MSGLEVEL=(1,1),
  REGION=4096K,
  CLASS=A,
  SCHENV=DB2LATE,
  MSGCLASS=O
...
```

This specification associates this batch job with the DB2LATE scheduling environment. It can be coded by the end user, or automatically supplied by installation-provided exits.

If the scheduling environment name specified is not defined in the active WLM policy, the job will fail with a JCL error during conversion, accompanied by an appropriate error message.

Existing JES2 or JES3 exits can be used to change the scheduling environment name associated with batch jobs. This can be done during JCL conversion. These exits can also be used to dynamically generate scheduling environment associations as work is submitted. This could be useful in migrating from another scheduling mechanism to scheduling environments. See [z/OS JES2 Installation Exits](#) or [z/OS JES3 Customization](#) for more information.

**Displaying Information about Scheduling Environments and Resource States**

Once you have defined scheduling environments, you can issue several different operator commands, both from MVS and from JES2 or JES3 to display information about the scheduling environments and about the resource states.

**MVS Operator Commands**

To display sysplex-level information about a scheduling environment, you can issue the following command from an MVS console:

```
D WLM,SCHENV=scheduling_environment
```

For example, to display information about the DB2LATE scheduling environment, here is the command you would issue and the response you would receive:
The AVAILABLE ON SYSTEMS field shows that at the time this command was issued, only systems SYS1 and SYS3 satisfied the requirements of the DB2LATE scheduling environment.

To display information about all scheduling environments in a sysplex, issue the command with an asterisk ('*') in the scheduling_environment field, as in this example:

```
D WLM, SCHENV=*
```

In this example, NOT AVAILABLE ON ANY SYSTEM is shown for the IMSPRIME scheduling environment, meaning that no systems in the sysplex currently satisfy the IMSPRIME requirements.

To display system-level information about a scheduling environment, use the SYSTEM=system_name parameter. You will see all of the resource names included in that scheduling environment, along with their required and current states. Requirements that are not satisfied are marked with an asterisk. So assuming that the DB2LATE scheduling environment is satisfied on SYS1 but not on SYS2, here is what the commands and responses might look like if you wanted the information for both systems:

```
D WLM, SCHENV=DB2LATE, SYSTEM=SYS1
D WLM, SCHENV=DB2LATE, SYSTEM=SYS2
```
To display information about a specific resource state on a specific system, use the command:

```
D WLM,RESOURCE=resource_name,SYSTEM=system
```

To display information about the DB2A resource state on system SYS1, for example, here is the command and response:

```
D WLM,RESOURCE=DB2A,SYSTEM=SYS1

IWM038I 12.21.05 WLM DISPLAY 181
RESOURCE: DB2A
DESCRIPTION: DB2 Subsystem
SYSTEM STATE
SYS1 ON
```

To display information about all resource settings, use the asterisk in the `resource_name` field. Also, to display information on all systems in the sysplex, use the `SYSTEMS` keyword in place of the `SYSTEM=system_name` parameter. So to display information about all the resource states on all systems in the sysplex, here is a typical command and response:

```
D WLM,RESOURCE=*,SYSTEMS

IWM038I 12.21.05 WLM DISPLAY 181
RESOURCE: DB2A
DESCRIPTION: DB2 Subsystem
SYSTEM STATE
SYS1 ON SYS2 OFF SYS3 RESET

RESOURCE: PRIMETIME
DESCRIPTION: Peak Business Hours
SYSTEM STATE
SYS1 OFF SYS2 OFF SYS3 OFF
```

**JES2/JES3 Operator Commands**

The JES2 display command `$D JOBQ` can be used to list those queued jobs associated with a scheduling environment, as in this example:

```
$Djobq,schenv=DB2LATE
JOB00007 $HASP608 JOB(SHAMIL)
$HASP608 JOB(SHAMIL) STATUS=(AWAITING EXECUTION),CLASS=S,
$HASP608 PRIORITY=1,SYSAFF=(ANY),HOLD=(NONE),
$HASP608 CMDAUTH=(LOCAL),OFFS=(),SECLABEL=,
$HASP608 TGS=2,PERCENT=0.3809),ARM_ELEMENT=NO,
$HASP608 SRVCLASS=HOTBATCH,SCHENV=DB2LATE
```

For JES3, use the `*INQUIRY,Q` command, as in this example:

```
*I,Q,SCHENV=IMSPROD
IATB674 JOB JOB123 (JOB32787) P=02 CL=Z MAIN(ALLOCATE)
IATB674 JOB JOBABC (JOB32790) P=02 CL=Z MAIN(ALLOCATE)
IATB674 JOB JOBDEF (JOB32791) P=02 CL=Z MAIN(ALLOCATE)
IATB674 JOB JOBGHI (JOB32800) P=02 CL=Z MAIN(ALLOCATE)
IATB674 JOB JOBJKL (JOB32807) P=02 CL=Z MAIN(ALLOCATE)
IATB674 JOB JOBMNO (JOB33101) P=02 CL=Z MAIN(ALLOCATE)
```

See [z/OS JES2 Commands](#) or [z/OS JES3 Commands](#) for more information on using JES2 or JES3 operator commands.
**SDSF Commands**

SDSF can display scheduling environment information, resource information, and allows modification of resource states.
Chapter 15. Managing Performance with the Guest Platform Management Provider (GPMP)

This information briefly describes the guest platform management provider (GPMP).

Overview of the Guest Platform Management Provider (GPMP)

Starting with zEnterprise System, the guest platform management provider (GPMP) is the interface between the z/OS Workload Manager (WLM) and the IBM zEnterprise Unified Resource Manager. The GPMP supports platform management functions and provides policy information to WLM about the platform-wide performance goals of workloads in which the z/OS system is participating.

Within the service definition, you can indicate on which system in the sysplex you want WLM to activate the guest platform management provider. If activation is selected, WLM automatically starts each guest platform management provider and restarts it as needed. Otherwise, you must manually start and stop the guest platform management provider. By default, activation is not enabled.

The guest platform management provider (GPMP) on a z/OS system sends all data over the intranode management network (INMN) using an Open Systems Adapter-Express3 1000BASE-T ethernet adapter (CHPID type OSM). Applications sending or receiving data over the INMN require z/OS Communications Server configuration to access this network. See “TCP/IP in an ensemble” in z/OS Communications Server IP Configuration Guide for the steps required to allow TCP/IP to participate in the ensemble, and to allow the guest platform management provider (GPMP) access to the INMN.

For a description of the guest platform management provider commands, refer to z/OS MVS System Commands.

For information about the guest platform management provider messages, refer to z/OS MVS System Messages, Vol 9 (IGF-IWM).

For information about the hardware requirements of the guest platform management provider and for a detailed description of how to manage performance of zEnterprise systems, refer to the following books:

- zEnterprise System: Ensemble Planning and Configuring Guide, GC27-2608

Access to these books requires an IBM Resource Link® user ID and password. To obtain your IBM Resource Link user ID and password log on to http://www.ibm.com/servers/resourcelink
Configuring the Guest Platform Management Provider on z/OS

Use this procedure to configure and run the guest platform management provider on your z/OS system. Depending on which z/OS version and release you have installed, the configuration steps you need to perform vary slightly.

Before You Begin

- Determine whether you want to run a guest platform management provider on z/OS and, if so, on which systems. Guest platform management providers are supported on logical partitions (LPARs) and z/VM guests running z/OS V1R10, or higher. See z/Enterprise System: Ensemble Planning and Configuring Guide, GC27-2608 for guidance.
- Make sure your systems have the appropriate level of service to support a guest platform management provider. For all supported releases of z/OS, specific program temporary fixes (PTFs) must be applied before you can configure and start a guest platform management provider. If you are running z/OS as a z/VM guest, specific PTFs are required for z/VM V6.1 as well.
  To find instructions about installing the necessary service when migrating to an IBM zEnterprise 196 (z196) server, see the appropriate book:
  - For z/OS V1R11 or V1R10 systems, see z/OS Migration
  - For z/VM V6R1 systems, see z/VM V6R1 Migration Guide, GC24-6201.
- Verify that the ensemble network configuration tasks for the intranode management network (INMN) have been completed. To enable the guest platform management provider to send performance data over the INMN, an ensemble administrator or network administrator must configure z/OS Communications Server to access this network. See z/OS Communications Server IP Configuration Guide, SC31-8775, for instructions about configuring TCP/IP in an ensemble.
- Make sure that you have the appropriate authority to configure the execution environment for the guest platform management provider on your system. To configure the environment, you must run modified JCL from an authorized user ID with UID=0. IBM provides a sample job, the HVEENV member in SYS1.SAMPLIB, for you to modify and run.
- Make sure that you have the appropriate authority to configure the security environment for the guest platform management provider on your system, or work with your installation's security administrator to accomplish the security-related tasks. This step documents the z/OS Security Server RACF commands that you need to use to control access to specific system resources. To complete this step, you might need to use the z/OS Security Server RACF Security Administrator's Guide, SA22-7683.
- Have additional books ready, if you need them. Because you perform all of the tasks in this procedure on your z/OS system, you might need to use these books:
  - z/OS MVS System Commands for DISPLAY WLM and MODIFY WLM command syntax and descriptions.
  - z/OS MVS System Messages, Vol 5 (EDG-GFS) for FEW messages that guest platform management providers issue.
  - z/OS MVS System Messages, Vol 9 (IGF-IWM) for IWM messages that WLM issues.
About This Task

One advantage to using a guest platform management provider is the ability to collect and view application data from IBM middleware products that are instrumented to the Open Group Application Response Measurement (ARM) 4.0 standard. To collect data from ARM-instrumented applications, z/OS WLM ARM services must be enabled. Steps in this procedure explain how to check the status of and to enable z/OS WLM ARM services.

In a sysplex environment, you can use the WLM ISPF application to enable WLM to start and manage guest platform management providers on all systems that are running supported z/OS releases. You also can name any particular systems that you want to exclude; for those named systems, WLM does not activate the guest platform management provider. Steps in this procedure explain how to enable WLM to manage guest platform management providers automatically, or how to manually start and manage them.

Both z/OS WLM and MODIFY command processing use the started task HVEMCA, in SYS1.PROCLIB, to start the guest platform management provider, which runs in the z/OS UNIX shell environment. After the guest platform management provider starts, it connects to WLM and to the IBM zEnterprise Unified Resource Manager through the INMN.

Procedure

1. Copy the sample JCL from the HVEENV member to your local proclib.
2. In your copy of the JCL from the HVEENV member, modify the variables listed in Table 14 to configure the execution environment for the guest platform management provider. If a dash is shown in the “Suggested value” column, you can use any value.

Table 14. z/OS Execution Environment Variables For The Guest Platform Management Provider

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
<th>Suggested Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA-ROOT</td>
<td>Directory on UNIX where the guest platform management provider stores the configuration and diagnostics data.</td>
<td>–</td>
</tr>
<tr>
<td>ENV_PROFILE</td>
<td>The UNIX profile for the guest platform management provider to use. This profile contains environmental information, such as TimeZone and Locale.</td>
<td>–</td>
</tr>
<tr>
<td>EXTRA_LIBPATH</td>
<td>Optional directory of environment library path.</td>
<td>–</td>
</tr>
<tr>
<td>JREBIN_ROOT</td>
<td>Directory on UNIX where the 32-bit Java executable is located. Use the latest service level of IBM Runtime Environment for Java 2 (JRE) 1.5 or 1.6.</td>
<td>–</td>
</tr>
<tr>
<td>LOGFILE_ROOT</td>
<td>Directory where the guest platform management provider startup-related diagnostics messages are stored.</td>
<td>–</td>
</tr>
<tr>
<td>MCA_GROUP</td>
<td>The group created in STEP01 of this JCL. Use HVEMCA as the group name.</td>
<td></td>
</tr>
<tr>
<td>MCA_USER</td>
<td>The user created in STEP01 of this JCL. Use HVEMCA1 as the user name.</td>
<td></td>
</tr>
</tbody>
</table>
3. In your copy of the JCL from the HVEENV member, add or modify JCL statement parameters to configure the security environment for the guest platform management provider. The HVEENV member in SYS1.SAMPLIB contains sample z/OS Security Server RACF commands that you or your installation’s security administrator can modify to control access to specific system resources.

   a. Required: Add a group profile and user profile for the HVEMCA started task.

      ADDGROUP HVEMCA SUPGROUP(SYS1) OWNER(SYS1) OMVS(GID(88888))
      ADDUSER HVEMCA1 DFLTGRP(HVEMCA) NOPASSWORD -
         OMVS(UID(0) PROGRAM(/bin/sh) -
            HOME(/usr/lpp/bcp/mca))
      RDEFINE STARTED HVEMCA.** -
         STDATA(USER(HVEMCA1) GROUP(HVEMCA))
      SETROPTS RACLIST(STARTED) REFRESH

   b. Required: Authorize the HVEMCA started task to use the INMN.

      SETROPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH) GENERIC(SERVAUTH)
      RDEFINE SERVAUTH EZB.OSM.sysname.tcpname
      SETROPTS RACLIST(SERVAUTH) REFRESH

   c. If you are using security labels (SECLABELs) in your environment, assign the user of the HVEMCA started task to the security label of SYSMULTI.

      PERMIT SYSMULTI CLASS(SECLABEL) ID(HVEMCA1) ACCESS(ALTER)
      ALTUSER HVEMCA1 SECLABEL(SYSMULTI)
      RDEFINE STARTED HVEMCA.** -
         STDATA(USER(HVEMCA1) GROUP(HVEMCA))
         SECLABEL(SYSMULTI)
      SETROPTS RACLIST(STARTED) REFRESH
      SETROPTS RACLIST(SECLABEL) REFRESH

4. Run your copy of the modified JCL from an authorized user with UID=0. The z/OS system issues a message indicating that the job was submitted, and issues another message when the job ends. Check the job log for message FEW2119I, which indicates successful processing of the environment variables.

   The job log also contains messages to indicate whether the security commands were successfully processed. For any error messages in the job log, use z/OS MVS System Messages to find message descriptions and suggested responses, correct the errors, and rerun the job.

5. From the MVS console, issue the DISPLAY WLM command to determine the status of the Application Response Measurement (ARM) services.

   D WLM,AM,ALL

   The system responds with message IWM075I. If message IWM075I indicates that ARM services are enabled, proceed to step 6. If message IWM075I indicates that ARM services are disabled, complete the following steps:

   a. Check message IWM075I for started application instances. To collect ARM data for these applications, you must restart them in step 8 after you have successfully configured and started the guest platform management provider.

      ARM-instrumented applications must be authorized to use ARM services.

   b. Issue the MODIFY WLM,AM command to enable ARM services.

      WLM,AM=ENABLE

      The system responds with message IWM072I.

6. Edit the z/OS WLM service definition to enable WLM to automatically activate and manage a guest platform management provider on all systems in the sysplex. Using the WLM ISPF application to modify the service definition is the
recommended method of managing guest platform management providers. If you do not use this method, you can manually start guest platform management providers by using the MODIFY WLM,GPMP command.

1. On the Choose Service Definition pop-up, select option 1 or 2 to edit an existing service definition.
2. On the Definition Menu, select option 11 to edit the guest platform management provider settings.
3. On the GPMP-Settings panel, enter 2 to enable WLM to automatically manage guest platform management providers. On this panel, you also can enter the names of the systems on which you do not want WLM to start a guest platform management provider.
4. Return to the Definition Menu.
5. Select File→Save or File→Save As to save your changes to this service definition.
6. Select Utilities→Install definition to make your changes available for policy activation.
7. Select Utilities→Activate service policy to activate the policy.

The system responds with messages IWM001I and IWM070I to indicate that the policy has been activated. When the system issues message FEW0600I to indicate successful connection to the INMN, the guest platform management provider is ready to collect and report performance data to the Unified Resource Manager.

7. If you enabled ARM services in step 5 and any ARM-instrumented applications were active when you issued that MODIFY command, you must restart these applications so ARM calls can be processed.

Results
The guest platform management provider is correctly configured and ready to collect performance data on your z/OS systems. To verify the status of the guest platform management provider, z/OS WLM, and any ARM-instrumented applications, you can issue the DISPLAY WLM,AM command and view the results in the message IWM075I display.

What To Do Next
If you encounter a problem running the guest platform management provider, IBM Support personnel might request you to run specific commands to collect data required to diagnose the problem.

To collect diagnostic logs and Java dumps, run the collectLogs.sh command. This command packages diagnostic data in a Java archive (JAR) file that you can send as directed by IBM Support. The command syntax is:

```
/usr/lpp/bcp/mca/bin/collectLogs.sh -dir path
```

Replace the variable path with the directory in which you want to place the JAR file. For example, suppose you issue this command, specifying /tmp as the location for the JAR file. When the collectLogs.sh command completes processing, you can see messages similar to the following example:
/usr/lpp/bcp/mca/bin/collectLogs.sh –dir /tmp
Debug logs gathered into file /tmp/evemcaDebug_2010_08_13_09_04_35.jar
PROCESSING COMPLETE
Chapter 16. Workload Management Migration

This information covers two different workload management migration scenarios, depending on whether you already have an existing service definition, or need to create one for the first time:

- **Creating a service definition for the first time.** In this scenario, you do not presently have a service definition, but would like to start exploiting workload management functions.
  
  For the required steps, see "Creating a Service Definition for the First Time."

- **Migrating to a new z/OS Release with an existing service definition.** In this scenario, you already have a service definition created on a previous release of z/OS and need to accommodate any changes introduced by a higher release.
  
  For the required steps, see "Migrating to a New z/OS Release With an Existing Service Definition" on page 143.

This information contains a checklist for each of the two scenarios. The major migration activities are described in detail following the checklist sections, as follows:

- "Restricting Access to the WLM Service Definition" on page 144
- "Start the Application and Enter/Edit the Service Definition" on page 145
- "Calculate the Size of the WLM Couple Data Set" on page 148
- "Allocate a WLM Couple Data Set" on page 148
- "Make a WLM Couple Data Set Available to the Sysplex for the First Time" on page 151
- "Make a Newly Formatted Couple Data Set Available to the Sysplex" on page 152
- "Migration Considerations for Velocity" on page 153
- "Migration Considerations for Discretionary Goal Management" on page 154
- "Migration Considerations for Protection of Critical Work" on page 155

Creating a Service Definition for the First Time

Use the following checklist to create a service definition for the first time, allocate the WLM couple data set, and then activate a service policy.

1. **Set up performance objectives.** If your installation already has performance objectives, Chapter 4, “Setting up a Service Definition,” on page 33 is still helpful in setting up a service policy with realistic performance goals.

2. **Set up a service definition** from the performance objectives.
   
   Refer to Chapter 4, “Setting up a Service Definition,” on page 33 through Chapter 14, “Defining Scheduling Environments,” on page 125 for further information.

3. **Restrict access to the ISPF administrative application.**
   
   Refer to "Restricting Access to the WLM Service Definition” on page 144 for further information.

4. **Start the application** and enter the service definition.
   
   Refer to "Start the Application and Enter/Edit the Service Definition” on page 145 for further information.

5. **Upgrade the sysplex couple data set**
Make sure that you have installed the z/OS release and allocated your sysplex couple data set with the IXCL1DSU utility.

For information about how to format the sysplex couple data set, refer to *z/OS MVS Setting Up a Sysplex*.

6. **Allocate a WLM couple data set.**
   
   For more information, refer to “Allocate a WLM Couple Data Set” on page 148.

7. **Make the WLM couple data set available** for use in the sysplex for the first time by either:
   - Issuing the SETXCF command
   - Updating the COUPLExx parmlib member and re-ipling

   For more information, see “Make a WLM Couple Data Set Available to the Sysplex for the First Time” on page 151.

8. **Install a service definition** on the WLM couple data set.
   
   Before you can activate a service policy, you need to install the service definition on the WLM couple data set. To do this, use either the WLM ISPF application, or the Install Definition Utility (new with z/OS R3).
   
   To use the ISPF application, go into it specifying the name of the data set containing your service definition. From the Definition Menu, go to UTILITIES on the action bar. Then select the pull-down option “Install Service Definition”.
   
   To use the Install Definition Utility, configure the sample JCL (member IWMINSTL, shipped in SYS1.SAMPLIB) as directed in the prolog. Once the JCL has been prepared, it can be started from the command console or submitted as a batch job. A simple service definition has been provided (member IWMSSDEF, also shipped in SYS1.SAMPLIB) that is available for those customers without any other definition.

9. **Adjust SMF recording**
   
   Before you run your systems in goal mode with z/OS V1R3, you should be aware of the changes in your SMF recording. There are several changes to SMF records for goal mode.
   
   In particular, you should turn off SMF type 99 records. They trace the actions SRM takes while in goal mode, and are written frequently. SMF type 99 records are for detailed audit information only. Before you switch your systems into goal mode, make sure you do not write SMF type 99 records unless you want them.
   
   If you do chargeback based on SMF record type 30 or record type 72 records, you may need to update your accounting package.

   For more information about SMF record changes for goal mode, see *z/OS MVS System Management Facilities (SMF)*.

10. **Activate a service policy**
    
    Once you have installed a service definition, you can activate a service policy. You can activate a policy either from the administrative application, with the VARY operator command, or the WLM Install Definition Utility.
    
    To activate a service policy from the application, choose the Utilities option from the action bar on the definition menu.
    
    To activate a service policy with the VARY command, specify
    
    VARY WLM, POLICY=xxxx
    
    where xxxx is the name of a policy defined in the installed service definition.
To activate a service policy with the Install Definition Utility, start or submit the sample JCL (member IWMINSTL in SYS1.SAMPLIB).

Once you issue the command, there is an active policy for the sysplex. Systems will start managing system resources to meet the goals defined in the service policy.

For more information about the VARY command, see z/OS MVS System Commands.

Migrating to a New z/OS Release With an Existing Service Definition

Use the following checklist to migrate to the current z/OS Release with an existing service definition. You may or may not need to reallocate the WLM couple data set, depending on which release you are migrating from.

1. Evaluate your service definition. At some point, either before or after you migrate to the new release, you may need to make one or more adjustments to your service definition, depending on which release you are migrating from:
   - If you are migrating from OS/390® Release 9 or earlier:
     - On OS/390 Release 10, you should be aware of new options available to help system administrators protect critical work, and how these options may affect other work. See Chapter 12, “Defining Special Protection Options for Critical Work,” on page 99 for more information.
     - Also, be aware that if you increase your service definition to LEVEL011, your service definition will be subject to a more rigorous verification. Errors may be flagged if you attempt to either save or install your service definition. The application will allow you to save the service definition, but will prevent you from installing it until the errors are corrected.
     - If you use the Release 10 application to modify a service definition below LEVEL011, these same conditions will instead appear as warnings when you save the service definition. In this case, it will not prevent you from installing the service definition.
     - For more information, see the discussion of LEVEL011 in “Service Definition Functionality Levels, CDS Format Levels, and WLM Application Levels” on page 146.
   - If you are migrating from OS/390 Release 10 or earlier:
     - If you plan on using the Intelligent Resource Director functions available on z/OS Release 1, you may need to define a specific coupling facility structure named SYSZWLM_xxxxxyyy in the CFRM policy. In addition, if you plan on using channel subsystem priority queuing, you’ll need to turn on I/O priority management, as described in “Specifying I/O Priority Management” on page 96.
     - See Chapter 18, “The Intelligent Resource Director,” on page 161 for more information on these and other steps you may need to take in order to make the Intelligent Resource Director work in your installation.

2. Ensure compatibility of downlevel releases.
   - If you are running mixed MVS releases on a sysplex, you need to install the compatibility PTFs on all downlevel systems to enable different levels of workload management to coexist until you can upgrade the entire sysplex to the new release. Refer to z/OS Migration for further details.

3. Selecting the correct z/OS release for policy updates
   - The WLM service definition is stored in ISPF tables, or in XML format. When a new release adds certain specifications to the service definition, structural changes to the ISPF tables are required. When you open a service definition
whose table structure is different (older) than the one currently used by the WLM application, the WLM application automatically updates the service definition structure. If this occurs, the service definition cannot be read by older levels of the WLM application.

Therefore, it is recommended to start updating a WLM policy with a higher level of administrative application (higher z/OS release) only when you are sure that you do not have to update that policy data set with a lower level of administrative application anymore.

The following releases changed the ISPF table structure:

- MVS V5R1 and V5R2
- OS/390 V1R3, V1R4, V1R6, and V1R10
- z/OS V1R1, V1R8, and V1R10

If you ship the service definition PDS between systems (for example, via a TSO transmit, FTP, shared DASD), you need to verify that the sending and receiving systems are at compatible levels.

4. **Start the WLM application**

   For more information, see “Start the Application and Enter/Edit the Service Definition” on page 145. Please note the discussion of service definition functionality levels in “Service Definition Functionality Levels, CDS Format Levels, and WLM Application Levels” on page 146. Once you choose to use a new functionality level, from that point on you must always use a level of the WLM application that is compatible with that functionality level. It is recommended that you do not use the new functions (which automatically updates the functionality level) until all those systems are upgraded to the new release.

---

### Migration Activities

The following sections provide more detail for certain migration activities referenced out of the migration checklists. To determine if you need to perform an activity, refer to the preceding checklists.

#### Restricting Access to the WLM Service Definition

Before you create a WLM service definition, you should determine who needs access to it, and the kind of functions each person needs to perform. The installation’s capacity planners, systems programmers that analyze workloads and the system’s performance, system operators, service administrator, and help desk support may all need access to the service definition information.

There are two levels of access you need to consider:

- Access to an MVS data set
- Access to the WLM couple data set

### Restricting Access to an MVS Data Set

Any user with access to the administrative application can create their own “practice” service definitions in their own MVS data set. The MVS data set containing the actual service definition that will be installed into the WLM couple data set should be protected. Use a data set profile, just as you would for any other data set. Give READ access only to those people who should be able to view the service definition in the MVS data set, and UPDATE access to those people who should be able to create or modify the service definition in the MVS data set.
Restricting Access to the WLM Couple Data Set

Once you have determined who needs access to the WLM couple data set itself, define the kind of access authority required, as follows:

**READ** With READ access, the user can extract a service definition from the WLM couple data set.

**UPDATE**
- With UPDATE access, the user can:
  - Do all the functions available for READ access
  - Install a service definition to a WLM couple data set
  - Activate a service policy.

To control access to the WLM couple data set, use RDEFINE to add a profile to the RACF database. Then use PERMIT to permit or deny access to the RACF profile. Do not forget to issue the SETROPTS REFRESH command after the PERMIT command to refresh the RACF database and activate the changes you have made.

---

**Example of RDEFINE for the WLM couple data set**

```bash
RDEFINE FACILITY MVSADMIN.WLM.POLICY UACC(NONE) NOTIFY(user)
```

Where:

- **user** Indicates the user that should be notified of unauthorized access attempts to the database.

**Example of PERMIT for the WLM couple data set**

```bash
PERMIT MVSADMIN.WLM.POLICY CLASS(FACILITY)
   ID(user) ACCESS(READ)

PERMIT MVSADMIN.WLM.POLICY CLASS(FACILITY)
   ID(user) ACCESS(UPDATE)
```

Where:

- **user** Indicates the user or user group that needs access to the WLM couple data set.

**ACCESS** Indicates the type of access, either READ or UPDATE.

---

**Start the Application and Enter/Edit the Service Definition**

Start the new release's WLM administrative application, and enter your service definition.

If you want to keep the previous release's WLM application, each version of the application must have unique library names. So, when you install the new release, make sure you keep the previous release application on the system under a unique name. You can rename your libraries by using an exit as described in Appendix A, “Customizing the WLM ISPF Application,” on page 233.
When you enter the service definition, you can keep the service definition in an MVS partitioned data set until you are ready to install the service definition into the WLM couple data set. If you are migrating to use a new version of the WLM application, always save the partitioned data set created by the previous version of the application for backup purposes.

**Before Starting the WLM Application**

To start the WLM application, you use the TSO/E REXX exec IWMARIN0. The exec concatenates (via LIBDEF and ALTLIB), the following libraries necessary to run the application:

<table>
<thead>
<tr>
<th>Library</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYS1.SBLSCLI0</td>
<td>Application REXX code</td>
</tr>
<tr>
<td>SYS1.SBLSKEL0</td>
<td>Application skeletons</td>
</tr>
<tr>
<td>SYS1.SBLSPNL0</td>
<td>Application panels</td>
</tr>
<tr>
<td>SYS1.SBLSTBL0</td>
<td>Application keylists and commands</td>
</tr>
<tr>
<td>SYS1.SBLSMSG0</td>
<td>Application messages</td>
</tr>
</tbody>
</table>

The exec also allocates some MVS partitioned data sets for the service definition using TSO ALLOCATE, and then invokes the WLM panels. If you have different data set conventions for your IPCS/WLM libraries, or if you use storage managed data sets, you should use the WLM application exits IWMAREX1 and IWMAREX2. For more information about how to code the exits, see [Appendix A, “Customizing the WLM ISPF Application,” on page 233](#). For more information about IWMARIN0, and for examples on how to start the application specifying the WLM exits, see [Appendix A, “Customizing the WLM ISPF Application,” on page 233](#).

**Start the WLM Application**

To start the application, specify:

```bash
ex 'SYS1.SBLSCLI0(IWMARIN0)'
```

For more information about IWMARIN0, and for examples on how to start the application specifying the WLM exits, see [Appendix A, “Customizing the WLM ISPF Application,” on page 233](#).

**Enter/Edit the Service Definition**

Type in the service definition (or else edit the existing service definition), and then save it as a partitioned data set (PDS), or a sequential data set (PS). For help on using the application, see [Chapter 21, “Using the WLM ISPF Application,” on page 183](#).

Note that if you change the service definition to use certain functions of the new release, you may not be able to use the service definition on the previous release.

**Service Definition Functionality Levels, CDS Format Levels, and WLM Application Levels**

A service definition has a functionality level for each release as shown in [Table 15](#).

**Table 15. Functionality Levels for Service Definition**

<table>
<thead>
<tr>
<th>Release</th>
<th>Functionality Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS/390 R4/R5</td>
<td>LEVEL004</td>
</tr>
<tr>
<td>OS/390 R6</td>
<td>LEVEL006 or LEVEL007 (LEVEL007 available only with APAR OW33509 installed)</td>
</tr>
<tr>
<td>OS/390 R7</td>
<td>LEVEL007 or LEVEL008 (LEVEL008 available only with APAR OW39854 installed)</td>
</tr>
</tbody>
</table>
Table 15. Functionality Levels for Service Definition (continued)

<table>
<thead>
<tr>
<th>Release</th>
<th>Functionality Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS/390 R8</td>
<td>LEVEL007 or LEVEL008 (LEVEL008 available only with APAR OW39854 installed)</td>
</tr>
<tr>
<td>OS/390 R9</td>
<td>LEVEL008</td>
</tr>
<tr>
<td>OS/390 V1.10 and z/OS V1.1</td>
<td>LEVEL011</td>
</tr>
<tr>
<td>z/OS V1.2, V1.3, V1.4, and V1.5</td>
<td>LEVEL013</td>
</tr>
<tr>
<td>z/OS V1.6 and V1.7</td>
<td>LEVEL013, LEVEL017 (LEVEL017 available only with APAR OA12784 installed)</td>
</tr>
<tr>
<td>z/OS V1.8 and V1.9</td>
<td>LEVEL019</td>
</tr>
<tr>
<td>z/OS V1.10</td>
<td>LEVEL021</td>
</tr>
<tr>
<td>z/OS V1.11</td>
<td>LEVEL023</td>
</tr>
<tr>
<td>z/OS V1.12</td>
<td>LEVEL025</td>
</tr>
<tr>
<td>z/OS V1.13</td>
<td>LEVEL025</td>
</tr>
</tbody>
</table>

Note: LEVEL005, LEVEL009, LEVEL010, LEVEL012, LEVEL014-016, LEVEL018, LEVEL020, LEVEL022, and LEVEL024 are reserved.

If you do not use any of the new functions for a new release, then the functionality level does not change, even if you are using the service definition on a new release. When you install the service definition, the system checks whether you have used any of the new functions, and sets the functionality level. For example, if you created your service definition on z/OS V1.8, then its functionality level is LEVEL017. If you installed this service definition from a z/OS V1.7 system but did not use any of the new functions, then its functionality level remains LEVEL013.

You should use the new functions when you are comfortable running the new release on your sysplex. Once you use the new functions and increase the functionality level, then you may not be able to use the service definition on a lower level system. For example, you cannot extract a LEVEL017 service definition from a z/OS V1.5 system. You also cannot activate a policy in a LEVEL017 service definition from a z/OS V1.5 system.

The following function, available on z/OS V1.11, increases the service definition level to LEVEL023:

- The number of report classes in the service definition exceeds 999

The following function, available on z/OS V1.12, increases the service definition level to LEVEL025:

- The service definition contains guest platform management provider (GPMP) configuration settings

Table 16. The Current WLM Couple Data Set Format Level

<table>
<thead>
<tr>
<th>Current CDS Format Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Format updated in OS/390 Release 4, with addition of scheduling environments. This is the format level for OS/390 Release 4 or higher.</td>
</tr>
</tbody>
</table>
Calculate the Size of the WLM Couple Data Set

When you reallocate a WLM couple data set, it must be at least as large as the current one or you will not be able to make it available to the sysplex. It is recommended that you increase the size of your couple data set to allow you to exploit new functions, even if you do not plan to exploit new functions immediately.

Use the WLM ISPF Application Utilities option (see “Utilities” on page 192) with the “Allocate couple data set using CDS values” option to determine your couple data set size. If you already have a record of the values, you can skip this step; just ensure you allocate the WLM couple data set with the same or larger values as the current one.

Allocate a WLM Couple Data Set

You need to define a WLM couple data set for storing the service definition information. If you are running a sysplex with mixed release levels, you should format the WLM couple data set from the highest level system. This allows you to use the current level of the WLM application.

To allocate the WLM couple data set you can either use the facility provided in the WLM application, or you can run an XCF utility. For each case, you need to estimate how many workload management objects you are storing on the WLM couple data set. You must provide an approximate number of the following:

- Policies in your service definition
- Workloads in your service definition
- Service classes in your service definition

The values you define are converted to space requirements for the WLM couple data set being allocated. The total space is not strictly partitioned according to these values. For most of these values, you can consider the total space to be one large pool.

If you specify 50 service classes, for instance, you are simply requesting that the space required to accommodate 50 service classes be added to the couple data set. You have NOT limited yourself to 50 service classes in the service definition. Although note that if you DO define more than 50 service classes, you will use up space that was allocated for something else.

You should define an alternate WLM couple data set (similar to the sysplex alternate couple data set) for recovery purposes. You can define an alternate WLM couple data set using the same method (either the ISPF application or the XCF utility), but specifying a different data set.

To allocate a WLM couple data set using the ISPF application, choose Utilities from the Definition Menu.

To allocate a WLM couple data set using the XCF utility, you can follow some JCL provided in the IWMFTCDS member of SYS1.SAMPLIB.

Sample JCL to Allocate a WLM Couple Data Set

To allocate a WLM couple data set, use the sample JCL and fill in the following information:

```
//FMTCDS JOB MSGLEVEL=(1,1)
//STEP1 EXEC PGM=IXCL1DSU
//STEPLIB DD DSN=SYS1.MIGLIB,DISP=SHR
```
Where:

**SYSPLEX**

The name of your sysplex as it appears in your COUPLExx parmlib member.

**DSN**

The name you are calling your WLM couple data set

**VOLSER**

A volume that you have access to. If you are using DFSMS, you do not need to specify a VOLSER.

**TYPE**

The type of function for which this data set is allocated. For a service definition, the type is WLM.

**ITEM NAME(POLICY) NUMBER( )**

Specifies that an increment of space large enough to accommodate the specified number of policies be allocated in the WLM couple data set (Default=5, Minimum=1, Maximum=99).

**ITEM NAME(WORKLOAD) NUMBER( )**

Specifies that an increment of space large enough to accommodate the specified number of workloads be allocated in the WLM couple data set (Default=32, Minimum=1, Maximum=999).

**ITEM NAME(SRVCLASS) NUMBER( )**

Specifies that an increment of space large enough to accommodate the specified number of service classes be allocated in the WLM couple data set (Default=128, Minimum=1, Maximum=999).

**Note:** WLM allows no more than 100 service classes to be defined in a service definition. The default, however, is 128. This will set aside as much space as you will ever need for service classes, as well as a little extra for other WLM objects.

**ITEM NAME(SVDEFEXT) NUMBER( )**

Specifies that an exact amount of space (in K bytes) for extension areas to the WLM Service Definition (IWMSVDEF) be allocated in the WLM couple data set (Default=0, Minimum=0, Maximum=8092).

**ITEM NAME(SVDCREXT) NUMBER( )**

Specifies that an exact amount of space (in K bytes) for extension areas to the WLM Service Definition Classification Rules (IWMSVDCR) be allocated in the WLM couple data set (Default=0, Minimum=0, Maximum=8092).
ITEM NAME(APPLENV) NUMBER( )
Specifies that an increment of space large enough to accommodate the
specified number of application environments be allocated in the WLM couple
data set (Default=100, Minimum=1, Maximum=999).

ITEM NAME(SVAEAEVT) NUMBER( )
Specifies that an exact amount of space (in K bytes) for extension areas to the
WLM Service Definition Application Environment Area (IWMSVAEA) be
allocated in the WLM couple data set (Default=0, Minimum=0,
Maximum=8092).

ITEM NAME(SCHENV) NUMBER( )
Specifies that an increment of space large enough to accommodate the
specified number of scheduling environments be allocated in the WLM couple
data set (Default=100, Minimum=1, Maximum=999).

ITEM NAME(SVSEAEVT) NUMBER( )
Specifies that an exact amount of space (in K bytes) for extension areas to the
WLM Service Definition Scheduling Environment Area (IWMSVSEA) be
allocated in the WLM couple data set (Default=0, Minimum=0,
Maximum=8092).

If you encounter a problem during processing, make sure you take a dump by
adding the following to your JCL and re-submit.

//SYSABEND DD SYSOUT=*  

Note: The intended users of SVDEFEXT, SVDCREXT, SVAEAEVT, and SVSEAEVT
are system management product vendors who wish to include some of their
own unique information about customer workload definitions along with
the WLM definitions. The WLM interfaces allow these extensions to
accompany the service class definitions, report class definitions, or even
classification rules. The amount of extra information is specific to each
product that exploits these interfaces. That product's documentation should
tell the customer how to set SVDEFEXT, SVDCREXT, SVAEAEVT, and
SVSEAEVT to ensure that there is sufficient space available in the WLM
couple data set to hold the extra information. For more information, see the
"Adding Program-Specific Extensions to a Service Definition" topic in the
"Using the Administrative Application Services" Chapter in z/OS MVS
Programming: Workload Management Services.

Increasing the Size of the WLM Couple Data Set
You must use a series of SETXCF commands to add a new, larger couple data set
as the primary WLM couple data set. During this processing you may encounter a
message from XCF (IXC250I) indicating your new couple data set is too small. The
message indicates which subrecords had insufficient space. If this occurs you must
realloc the new couple data set with a larger size.

From the Definition Menu in the ISPF application, choose Utilities. In the Utilities
pull-down, choose "4. Allocate couple data set." Record the values that you see in
that panel. Then choose "5 Allocate couple data set using CDS values." Record the
values that you see in that panel. Now compare the two sets of values and choose
the highest from each category. For example, if the two panels showed the
following values:

<table>
<thead>
<tr>
<th>Allocate couple data set</th>
<th>Allocate couple data set using CDS values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service policies ... 5</td>
<td>Service policies ... 10</td>
</tr>
<tr>
<td>Workloads ....... 40</td>
<td>Workloads ....... 35</td>
</tr>
<tr>
<td>Service classes .... 35</td>
<td>Service classes .... 30</td>
</tr>
</tbody>
</table>
Then you should use the highest values in each category, in this case 10, 40, 35, 100, and 80, to allocate your new couple data set.

### Make a WLM Couple Data Set Available to the Sysplex for the First Time

This section applies when creating a service definition for the first time. If you already have a service definition and want to make a re-allocated WLM couple data set available, see “Make a Newly Formatted Couple Data Set Available to the Sysplex” on page 152.

To make your WLM couple data set available to the sysplex, you must either:

- Update your COUPLExx parmlib member to include the data set name and volume of your WLM couple data set, and re-IPL. Use this option if you have not yet IPLed in a sysplex.
- Issue the SETXCF command, if you are already IPLed in a sysplex. You must still update your COUPLExx member for subsequent IPLs.

### Using the SETXCF Command

To make the WLM couple data set available to the sysplex, you can use the SETXCF command. Remember that you still need to update your COUPLExx member as shown in “Updating the COUPLExx Member” so that any subsequent IPLs will automatically pick up the WLM couple data sets.

For more information about using the SETXCF command, see z/OS MVS System Commands.

### Examples of SETXCF Command

- To make a primary WLM couple data set called SYS1.WLMCDS01 residing on volume TEMP01 available to the sysplex, enter the following command:

  ```
  SETXCF COUPLE,TYPE=WLM,PCOUPLE=(SYS1.WLMCDS01,TEMP01)
  ```

- To make an alternate WLM couple data set called SYS1.WLMCDS02 residing on volume TEMP02 available to the sysplex, enter the following command:

  ```
  SETXCF COUPLE,TYPE=WLM,ACOUPLE=(SYS1.WLMCDS02,TEMP02)
  ```

### Updating the COUPLExx Member

To make the WLM couple data set available for use in the sysplex, you need to update the DATA keyword in the COUPLExx parmlib member, and IPL so that the member is in use. For more information about updating the COUPLExx member, see z/OS MVS Setting Up a Sysplex.
Example of Updating COUPLExx Member

<table>
<thead>
<tr>
<th>DATA</th>
<th>TYPE(WLM)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PCOUPLE(SYS1.WLMCDS01,TEMP01)</td>
</tr>
<tr>
<td></td>
<td>ACOUPLE(SYS1.WLMCDS02,TEMP02)</td>
</tr>
</tbody>
</table>

Where:

**TYPE**
The function type, WLM.

**PCOUPLE(dataset.name, volume)**
The WLM couple data set name, and the volume it resides on.

**ACOUPLE(dataset.name, volume)**
The alternate WLM couple data set name, and the volume it resides on. If you do not have an alternate WLM couple data set, then delete this keyword.

Specify the modified COUPLExx member on your next IPL.

---

**Make a Newly Formatted Couple Data Set Available to the Sysplex**

This section applies when you already have WLM couple data sets, have just allocated new WLM couple data sets, and want to make them available to the sysplex. You do this, for example, if you want to increase the size of the existing WLM couple data sets.

You must use a series of SETXCF command to switch from the currently active primary and alternate couple data sets to the new couple data sets. All systems in the sysplex then operate with the newly allocated data set.

If you are making newly formatted WLM couple data sets available to the sysplex, you can continue to use an older WLM application to modify, install and activate your service definition (as long as new functions are not exploited), or you can switch to using the z/OS Release 2 WLM application.

For more information on compatibility of release levels, WLM application levels, couple data set formats, and functionality levels, see “Service Definition Functionality Levels, CDS Format Levels, and WLM Application Levels” on page 146.
Example of Making Re-allocated Couple Data Sets Available

1. Allocate two new couple data sets as described in "Allocate a WLM Couple Data Set" on page 148. For this example, it is assumed you want a primary and an alternate couple data set, and that the names of the new data sets are SYS1.WLMP residing on volume SYS001 and SYS1.WLMA residing on volume SYS002.

2. Make SYS1.WLMP the alternate using the command:
   
   SETXCF COUPLE,TYPE=WLM,ACOUPLE=(SYS1.WLMP,SYS001)

   As part of this processing, SETXCF copies the contents of the current primary WLM couple data set to SYS1.WLMP which now is the new alternate.

3. Switch SYS1.WLMP to primary using the command:
   
   SETXCF COUPLE,TYPE=WLM,PSWITCH

4. Now make SYS1.WLMA the new alternate using the command:
   
   SETXCF COUPLE,TYPE=WLM,ACOUPLE=(SYS1.WLMA,SYS002)

   As in Step 1, this causes the contents of the new primary WLM couple data set SYS1.WLMP to be copied to the new alternate SYS1.WLMA.

Migration Considerations for Velocity

Initiation delays cause the velocity value to decrease. Recalculate and adjust your velocity goals accordingly. See "Velocity Formula" on page 50 for information on calculating velocity.

Before migrating to WLM batch management, you can estimate the new velocity goal for a service class as follows:

Note: All jobs with the same service class should be migrated together to WLM-managed job classes.

- Run the jobs under normal circumstances
- Examine the initiation delay data:
  - In the IWMWRCAA data area if you are using the workload reporting services:
    - RCAETOTDQ
      Total delay samples, including initiation delay
    - RCAETOTU
      Total using samples
  - In the SMF Type 72 record, Subtype 3, if you are using RMF:
    - R723CTDQ
      Total delay samples, including initiation delay
    - R723CTOU
      Total using samples

Include the initiation delay in the velocity formula for an estimate of the new, lower velocity. Plugging this delay data into the velocity formula gives you:
Migration Considerations for Discretionary Goal Management

Certain types of work, when overachieving their goals, potentially will have their resources “capped” in order to give discretionary work a better chance to run. Specifically, work that is not part of a resource group and has one of the following two types of goals will be eligible for this resource donation:

- A velocity goal of 30 or less
- A response time goal of over one minute

Work that is eligible for resource donation may be affected in OS/390 Release 6 and higher if this work has been significantly overachieving its goals. If you have eligible work that must overachieve its goals to provide the required level of service, adjust the goals to more accurately reflect the work’s true requirements.

Migration Considerations for Dynamic Alias Management

With dynamic alias management, WLM can automatically perform alias address reassignments to help work meet its goals and to minimize IOS queueing.

It is recommended not to use dynamic alias management for a device unless all systems sharing that device have dynamic alias management enabled. Otherwise, WLM will be attempting to manage alias assignments without taking into account the activity from the non-participating systems.

See “Specifying Dynamic Alias Management” on page 96 for more information.

Migration Considerations for Multisystem Enclaves

Before using multisystem enclaves, an installation needs to define a specific coupling facility structure named SYSZWLW, WORKUNIT in the CFRM policy. See Chapter 17, “Defining a Coupling Facility Structure for Multisystem Enclave Support,” on page 157 for more information.
Programs that use data from the SMF 30 record may need to be updated in conjunction with multisystem enclave support. The enclave owner's SMF 30 record has new fields containing the CPU time accumulated by all of its split transactions, for all systems on which they executed.

For more detailed information on multisystem enclaves, see the “Creating and Using Enclaves” in "z/OS MVS Programming: Workload Management Services".

**Migration Considerations for Protection of Critical Work**

You should be aware of several options available to help system administrators protect critical work, and how these options may affect other work.

These options include:

- Long-term storage protection
- Long-term CPU protection
- Exemption from management as a transaction server

See Chapter 12, “Defining Special Protection Options for Critical Work,” on page 99 for more information on these options.

**Migration Considerations for Managing Non-Enclave Work in Enclave Servers**

Starting with z/OS Release V1R12 the non-enclave work of enclave servers is managed towards the first service class period of the address space performance goal. Based on this expanded performance management it is recommended to verify the performance goals for the service class of the address spaces which process enclave work.

For a detailed description about how performance is managed in address spaces with enclaves, refer to "z/OS MVS Programming: Workload Management Services".

**WLM Managed Batch Initiator Balancing**

Starting with z/OS Release V1R4 and JES2 V1R4, and z/OS V1R5 with JES V1R5 WLM is enhanced to improve the balancing of WLM managed batch initiators between systems of a sysplex. While in earlier releases a balancing of initiators between high and low loaded systems was only done when new initiators were started, this is now done when initiators are already available. On highly utilized systems, the number of initiators is reduced while new ones are started on low utilized systems. This enhancement can improve sysplex performance with better use of the processing capability of each system. WLM attempts to distribute the initiators across all members in the sysplex to reduce batch work on highly used systems while taking care that jobs with affinities to specific systems are not hurt by WLM decisions. Initiators are stopped on systems that are utilized over 95% when another system in the sysplex offers the required capacity for such an initiator. WLM also increases the number of initiators more aggressively when a system is low utilized and jobs are waiting for execution.

Batch Initiator Balancing improves the performance and throughput of batch workload over the sysplex. Its intention is not to reach an equally balanced distribution of batch jobs over the LPARs of a sysplex. That is why initiator balancing only comes into effect, when at least one of the systems of the sysplex has a CPU utilization of more than 95%, while other systems have more idle capacity. When the most loaded system has still enough idle capacity to run batch
jobs without CPU constraints, it would not improve the total batch throughput if initiators were moved away from that system to other systems, even if the other systems had more idle CPU capacity.

Consider Resource Group Maximum in WLM Batch Initiator Management

Starting with z/OS V1R12 WLM considers the resource group maximum and whether the projected increase in service demand will not exceed the resource group maximum. When the service is already capped due to the resource group maximum, then no additional initiators are started.

Note: This resource maximum check is not done on a LPAR for a service class if WLM has not started an initiator for that particular service class on that LPAR, or if WLM has only started initiators in that service class on other LPARs in the sysplex. In this case, WLM can start an initiator for that service class on that LPAR even if resource group capping is already active for that service class. Otherwise, WLM cannot determine how much capacity is being used on average by a batch job for that service class.
Chapter 17. Defining a Coupling Facility Structure for Multisystem Enclave Support

Some work managers split large transactions across multiple systems in a parallel sysplex, improving the transaction's overall response time. These work managers can use multisystem enclaves to provide consistent management and reporting for these types of transactions.

Among the benefits of using multisystem enclaves:

- All parts of a split transaction are managed to the same service class. If the service class has multiple periods, the CPU usage of the entire transaction is used to switch periods.
- The enclave owner’s SMF 30 record includes CPU time accumulated by all of its split transactions, for all systems on which they executed.

Before using multisystem enclaves, an installation needs to define a specific coupling facility structure named SYSZWLM_WORKUNIT in the CFRM policy. Once the CFRM policy with this structure definition is activated, WLM will automatically connect to the structure, enabling the use of multisystem enclaves.

This information shows how to define the SYSZWLM_WORKUNIT structure, a prerequisite to the use of multisystem enclaves. For more information on defining coupling facilities, see z/OS MVS Setting Up a Sysplex.

Programs that use data from the SMF 30 record may need to be updated in conjunction with multisystem enclave support. The enclave owner’s SMF 30 record has new fields containing the CPU time accumulated by all of its split transactions, for all systems on which they executed.

For more detailed information on multisystem enclaves, see the “Creating and Using Enclaves” in z/OS MVS Programming: Workload Management Services.

Defining the Coupling Facility

It may be difficult to size the SYSZWLM_WORKUNIT structure at first, as there is no sure way to know exactly how many parallel units-of-work may exist at any given time. The best option is take a best guess at the initial and maximum sizes and then alter the structure size based on performance and/or change in demand.

If the structure’s maximum size is defined too low, work managers will experience failures when they try to export enclaves. It is the work manager’s responsibility to respond to such a failure. The work requests may instead be run locally (increasing the response time), or the work requests may fail.

The best way to estimate the storage size needed is to use the CFSIZER tool, which can be accessed through the following web page:

http://www.ibm.com/systems/z/cfsizer

Alternately, there are formulas in PR/SM Planning Guide to help estimate the storage size needed. As shown in Table 17 on page 158, the TDEC value is the estimated number of concurrently executing parallel units-of-work. Use the TDEC estimate along with the other values explicitly given in the table, as follows:
Table 17. Values to Use in Storage Estimation Formulas

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
<th>Specify:</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDEC</td>
<td>Total directory entry count — the maximum number of concurrently executing parallel units-of-work</td>
<td>Best Estimate</td>
</tr>
<tr>
<td>TDAEC</td>
<td>Total data area element count</td>
<td>TDEC X 2</td>
</tr>
<tr>
<td>MSC</td>
<td>Maximum number of storage classes</td>
<td>1</td>
</tr>
<tr>
<td>MCC</td>
<td>Maximum number of castout classes</td>
<td>1</td>
</tr>
<tr>
<td>MDAS</td>
<td>Maximum number of data area elements associated with a directory entry</td>
<td>32</td>
</tr>
<tr>
<td>DAEX</td>
<td>Data area element characteristic</td>
<td>3</td>
</tr>
<tr>
<td>AAI</td>
<td>Adjunct assignment indicator</td>
<td>0</td>
</tr>
<tr>
<td>R_de/R_data</td>
<td>Directory to data ratio</td>
<td>1/2</td>
</tr>
</tbody>
</table>

Once you have estimated the initial and maximum sizes for the SYSZWLM_WORKUNIT structure, define the structure as described in z/OS MVS Setting Up a Sysplex. Keep in mind the following:
- WLM requests a coupling facility with “default” connectivity.
- Non-volatility is not required.
- The coupling facility control code must be at CFLEVEL 9 or higher.
The following sample JCL shows the definition of a SYSZWLM_WORKUNIT structure:

```
//POLICYX JOB
//STEP1 EXEC PGM=IXCMIAPU
//SYSPRINT DD SYSOUT=A
//SYSIN DD *

DATA TYPE(CFRM) REPORT(YES)
DEFINE POLICY NAME(POLICY1) REPLACE(YES)

CF NAME(FACIL01)
    TYPE(123456)
    MFG(IBM)
    PLANT(02)
    SEQUENCE(123456789012)
    PARTITION(1)
    CPCID(00)
    SIDE(0)
    DUMPSPACE(2000)

CF NAME(FACIL02)
    TYPE(123456)
    MFG(IBM)
    PLANT(02)
    SEQUENCE(123456789012)
    PARTITION(2)
    CPCID(00)
    SIDE(1)
    DUMPSPACE(2000)

STRUCTURE NAME(SYSZWLM_WORKUNIT)
    SIZE(4000)
    INITSIZE(3328)
    PREFLIST(FACIL02,FACIL01)
```
Shutting Down the Coupling Facility

If it becomes necessary to shut down a coupling facility containing the SYSZWLM_WORKUNIT structure (either to apply maintenance or to reconfigure), there are two options:

- If another coupling facility has enough storage available, use the XES system-managed rebuild function to rebuild the SYSZWLM_WORKUNIT structure into another coupling facility. See z/OS MVS System Commands for more information.
- If there is no other coupling facility into which the SYSZWLM_WORKUNIT structure can be rebuilt, the structure will be deleted when its coupling facility is shut down and therefore multisystem enclave support will be disabled (as described in "Coupling Facility Failures").

An installation should take the appropriate steps to quiesce any active work which may be using multisystem enclaves before shutting down the coupling facility containing the SYSZWLM_WORKUNIT structure.

Coupling Facility Failures

If the coupling facility containing the SYSZWLM_WORKUNIT structure fails, or if the structure itself fails, then all existing multisystem enclaves will be lost. It is the work manager’s responsibility to respond to such a failure. The work manager may fail the work requests, or it may process them without using multisystem enclaves.

If another coupling facility is available, WLM will automatically create a new (empty) SYSZWLM_WORKUNIT structure in it. New multisystem enclaves can now be created for new work requests.

If the original coupling facility is still intact, but the link fails, then the use of multisystem enclaves is temporarily disabled. Again, it is the work manager’s responsibility to respond to this situation, either failing the work requests, or processing them without using multisystem enclaves. When the link is restored, then the use of multisystem enclaves can continue.
The Intelligent Resource Director (IRD) extends the concept of goal-oriented resource management by allowing you to group system images that are resident on the same physical server running in LPAR mode, and in the same parallel sysplex, into an “LPAR cluster.” This gives Workload Management the ability to manage processor and channel subsystem resources, not just in one single image but across the entire cluster of system images. The following figure shows one LPAR cluster in one central processor complex (CPC):

![Figure 27. One LPAR Cluster on One CPC](image)

A CPC can have multiple LPAR clusters supporting different parallel sysplexes, and a parallel sysplex can in turn comprise multiple LPAR clusters in different CPCs. This is illustrated in the following figure, in which two sysplexes across two CPCs are grouped into four LPAR clusters:
WLM manages a parallel sysplex by directing work to the available resources. With the Intelligent Resource Director, WLM can additionally move resources within an LPAR cluster to the work. Processor resources are automatically moved to the partitions with the greatest need, based on the business goals of the workloads they are running. Channels are automatically moved to the I/O control units with the greatest need, based on the business goals of the workloads using them.

The LINUX partition is not part of the parallel sysplex, but WLM manages it as if it were part of it.

The three functions that make up the Intelligent Resource Director are as follows:

- LPAR CPU Management
- Dynamic Channel Path Management
- Channel Subsystem Priority Queuing.

**Note:** Dynamic channel path management and channel subsystem priority queueing also are functional in single-system environments, such as a CPC operating in basic mode, or a z/OS system operating as a monoplex. If a z/OS system is operating as a monoplex, each z/OS system must be assigned a different sysplex name.
LPAR CPU Management

LPAR CPU Management allows dynamic adjustment of processor resources across logical partitions in the same LPAR cluster. WLM achieves this with two different mechanisms:

- **LPAR weight management**
  When you divide your central processing complex into separate logical partitions, each partition is assigned its own “LPAR weight,” which corresponds to the percentage of overall processing power that is guaranteed to the work in that partition. Previously, if the workload shifted to the extent that more processing power was needed in a particular partition, the weights had to be changed manually. With LPAR weight management, you give each logical partition an initial LPAR weight, along with an optional minimum and maximum weight if desired. WLM will then dynamically balance these weights to best meet the goals of the work in the partitions, with no human intervention. Note that the total weight of the cluster as a whole will remain constant, so LPARs outside the cluster are unaffected.

  LPAR weight management takes effect when two or more LPARs in the cluster are running CPU-constrained, and the CPC’s shared physical CPs are fully utilized. When just one LPAR is using all or most of the CPC because other LPARs are idle, then LPAR weight management will have no effect on resource distribution.

  If there are non z/OS partitions in the LPAR cluster, the Intelligent Resource Director manages the weight of these partitions by exchanging weight between the non z/OS and the z/OS partitions of the cluster. LPAR weight of any non z/OS partition will not be reset to its initial weight when the z/OS partition is reset or deactivated. Non z/OS partitions weight can be reset through the service element panel in this situation.

  **Note:** LPAR weight management can be done for standard processors only. LPAR weight management is not supported for zIIPs and zAAPs.

- **VARY CPU Management**
  VARY CPU Management works for z/OS partitions only. It works hand-in-hand with LPAR weight management. As the LPAR weights change, the number of online logical CPUs is also changed to maintain the best match between logical CPU speed and physical CPU speed. Optimizing the number of logical CPUs benefits workloads that have large amounts of work done under single tasks, and minimizes LPAR overhead for all workloads.

  LPAR CPU Management requires zSeries servers in z/Architecture® mode. z/OS images require a CF level 9 coupling facility structure.
  LINUX for S/390® or z/Series requires kernel 2.4 or higher. General purpose CPUs are supported, but Integrated Facility for LINUX (IFL) CPUs are not supported.

Dynamic Channel Path Management

Prior to Dynamic Channel Path Management, all channel paths to I/O control units had to be statically defined. In the event of a significant shift in workload, the channel path definitions would have to be reevaluated, manually updated via HCD, and activated or POR'ed into the configuration. Dynamic Channel Path Management lets Workload Management dynamically move channel paths through the ESCON® Director from one I/O control unit to another, in response to changes in the workload requirements. By defining a number of channel paths as “managed,” they become eligible for this dynamic assignment. By moving more
bandwidth to the important work that needs it, your DASD I/O resources are used more efficiently. This may decrease the number of channel paths you need in the first place, and could improve availability — in the event of a hardware failure, another channel could be dynamically moved over to handle the work requests.

Dynamic Channel Path Management operates in two modes: balance mode and goal mode. In balance mode, Dynamic Channel Path Management will attempt to equalize performance across all of the managed control units. In goal mode, which is available only when WLM is operating in goal mode on all systems in an LPAR cluster, WLM will still attempt to equalize performance, as in balance mode. In addition, when work is failing to meet its performance goals due to I/O delays, WLM will take additional steps to manage the channel bandwidth accordingly, so that important work meets its goals.

Dynamic Channel Path Management requires z/OS and a zSeries server in z/Architecture mode, running in either Basic or LPAR mode. If a system image running Dynamic Channel Path Management in LPAR mode is defined as being part of a multisystem sysplex, it also requires a CF level 9 coupling facility structure, even if it is the only image currently running on the CPC.

See the WLM website at:


for the latest list of devices that support Dynamic Channel Path Management:

---

Channel Subsystem Priority Queuing

Channel Subsystem Priority Queuing is an extension of the existing concept of I/O priority queuing. Previously, I/O requests were handled by the channel subsystem on a first-in, first-out basis. This could at times cause high priority work to be delayed behind low priority work. With Channel Subsystem Priority Queuing, if important work is missing its goals due to I/O contention on channels shared with other work, it will be given a higher channel subsystem I/O priority than the less important work. This function goes hand in hand with the Dynamic Channel Path Management described — as additional channel paths are moved to control units to help an important workload meet goals, Channel Subsystem Priority Queuing ensures that the important workload receives the additional bandwidth before less important workloads that happen to be using the same channel.

WLM sets the priorities using the following basic scheme:

- System related work is given the highest priority
- High importance work missing goals is given a higher priority than other work
- Work meeting goals is managed so that light I/O users will have a higher priority than heavy I/O users
- Discretionary work is given the lowest priority in the system.

Channel Subsystem Priority Queuing requires z/OS and a zSeries server in z/Architecture mode. It works in both basic mode and LPAR mode, and does not require a coupling facility structure.
Example

To illustrate how the Intelligent Resource Director works in a mixed workload environment, here’s an example:

- You have three workloads running on one server:
  - **Online Transactions**, your most important workload. This runs only during the day shift.
  - **Data Mining**, which has a medium importance. This is always running, and will consume as much resource as you give it.
  - **Batch**, which is your lowest importance work. Like data mining, it is always running, and will consume as much resource as you give it.

- In this example, the server is divided into two logical partitions (and both partitions are in the same sysplex):
  - **Partition 1** runs both the online transactions and the batch work, as they happen to share the same database.
  - **Partition 2** runs the data mining work.

Day Shift

This first figure shows a day shift configuration. As the online transaction workload is the most important, Partition 1 is given a high enough weight to ensure that the online transaction work does not miss its goals due to CPU delay. Within the partition, the existing Workload Management function is making sure that the online transaction work is meeting its goals before giving any CPU resource to the batch work.

The DASD used by the online transaction work is given enough channel bandwidth to ensure that channel path delays do not cause the work to miss its goals. The channel subsystem I/O priority ensures that online transaction I/O
requests are handled first. Even though the batch work is running in Partition 1 (with the increased partition weight and channel bandwidth), the data mining I/O requests will still take precedence over the batch I/O requests if the data mining work is not meeting its goals.

Night Shift

Partition 1:
Weight = 25

Partition 2:
Weight = 75

Batch
(Imp. 3)

Data Mining
(Imp. 2)

Channel Subsystem

I/O Priority
1. Data Mining
2. Batch

Figure 30. Intelligent Resource Director Example – Night Shift

This figure shows the night shift, when there are no more online transactions. If the partition weights had remained the same, then the batch work would be consuming most of the CPU resource, and using most of the I/O bandwidth, even though the more important data mining work may still be missing its goals. LPAR CPU Management automatically adjusts to this change in workload, adjusting the partition weights accordingly. Now the data mining work will receive the CPU resource it needs to meet its goals. Similarly, dynamic channel path management will move most of the I/O bandwidth back to the data mining work.

Making the Intelligent Resource Director Work

There are several tasks you need to perform to make the Intelligent Resource Director work in your installation:

- Defining the SYZSWLM_xxxxxyy CoP Coupling Facility Structure
- Enabling LPAR CPU Management
- Enabling Dynamic Channel Path Management
- Enabling Channel Subsystem Priority Queuing.
Defining the SYSZWLM_xxxxxyyyy Coupling Facility Structure

Before using either LPAR CPU Management or Dynamic Channel Path Management in a multisystem sysplex, you'll need to define a specific coupling facility structure named SYSZWLM_xxxxxyyyy in the CFRM policy for each LPAR cluster.

In each SYSZWLM_xxxxxyyyy structure, the 8-character xxxxyyyy field represents a portion of the CPU ID for the CPC. To obtain this 8-character field, issue the DM=CPU command, as follows:

```
SYS1 D m=cpu
SYS1 IEE174I 13.51.42 DISPLAY M 884
PROCESSOR STATUS
ID CPU SERIAL
1 + 1315122064
```

Use the last eight characters in the returned CPU ID. So in this example, the coupling facility structure would be named SYSZWLM_15122064.

Define the structure as in the following sample JCL code:

```jcl
//POLICYX JOB
//STEP1 EXEC PGM=IXCMIAPU
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
DATA TYPE(CFRM) REPORT(YES)
DEFINE POLICY NAME(POLICY1) REPLACE(YES)
...
STRUCTURE NAME(SYSZWLM_15122064)
  MINSIZE(10000) /* 1K UNITS */
  INITSIZE(10000)
  SIZE(10000)
  PREFLIST(FACIL01,FACIL02)
```

Activate the new CFRM policy using the SETXCF START command, as follows:

```
SETXCF START,POL,POLNAME=POLICY1,TYPE=CFRM
```

Once the CFRM policy is activated, WLM will automatically connect to the structure, enabling the use of the Intelligent Resource Director.

Keep in mind the following:
- WLM requests a coupling facility with “default” connectivity.
- Non-volatility is not required.
- The coupling facility control code must be at CFLEVEL 9 or higher.

If it becomes necessary to shut down a coupling facility containing the SYSZWLM_xxxxxyyyy structure (either to apply maintenance or to reconfigure), there are two options:
- If another coupling facility has enough storage available, use the XES system-managed rebuild function to rebuild the SYSZWLM_xxxxxyyyy structure into another coupling facility.
- If there is no other coupling facility into which the SYSZWLM_xxxxxyyyy structure can be rebuilt, the structure will be deleted when its coupling facility is shut down and therefore LPAR CPU Management or Dynamic Channel Path Management in a multisystem sysplex will be disabled.

If the coupling facility containing the SYSZWLM_xxxxxyyyy structure fails, or if the structure itself fails, then all LPAR CPU Management or Dynamic Channel Path Management in a multisystem sysplex is stopped. If another coupling facility is
available, WLM will automatically create a new (empty) SYSZWLM_xyyyyyy structure in it. These functions can now resume.

If the original coupling facility is still intact, but the link fails, then these functions will be temporarily disabled. When the link is restored, they will resume.

**Enabling LPAR CPU Management**

Once you have defined the SYSZWLM_xyyyyyy coupling facility structure, the remaining actions you need to take to enable LPAR CPU Management will all occur on the hardware management console (HMC).

- In the Primary Support Element Workplace, make sure that “Not dedicated central processors” is selected in the activation profile of each logical partition. This ensures that the logical partition will use shared CPs instead of dedicated CPs. On the same panel, specify the initial and reserved number of logical CPs you wish to have available to each logical partition. Production partitions which potentially need access to the full power of the CPC should be defined with the maximum number of logical CPs (equal to the number of shared physical CPs). Other partitions can be defined with a fewer number of logical CPs if they need less power, or if you wish to specifically restrict them to less power. At a minimum, the number of logical CPs should be sufficient to achieve the partition’s maximum weight.

  For some logical partitions, you may not need to make any of those changes, as the settings are already correct. If you do need to make any of those changes, note that you’ll need to deactivate and then reactivate the logical partition for the changes to take effect. For the remaining actions, you will *not* need to do this after making changes.

- For each logical partition that will participate in LPAR weight management, do the following:
  - Make sure that “Initial Capping” is turned off. WLM cannot manage the weight of a logical partition that is capped.
  - Enter the initial processing weight. This becomes the logical partition’s weight when it is first IPLed.
  - Enter the minimum and maximum weights. These set the lower and upper limits for the weights that WLM will assign to the logical partition.
  - Check the “WLM Managed” box. This is the final step in activating LPAR weight management.

Once you have made those changes for z/OS images, you do not have to do anything else to activate VARY CPU Management. You can disable VARY CPU Management, if you wish, by adding the keyword VARYCPU=NO to the IEAOPTxx parmlib member and then issuing the SET OPT=xx command to activate the change. The scope of this command is by single system. If there are multiple systems in the LPAR cluster, then the other systems will continue to use VARY CPU Management. You can set VARYCPU=YES to return the system to VARY CPU Management. Note that if any CPs were taken offline by an operator, those CPs will need to be configured back online before they can again be managed by VARY CPU Management.

**Enabling Non z/OS CPU Management**

In addition to the actions you have already taken to enable LPAR CPU Management, you need to do the following to enable non z/OS CPU Management:
Before activating the LINUX partition, specify the CP management cluster name under the Options tab of the activation profile. This causes LPAR to group the LINUX partition with the appropriate sysplex.

For the LINUX partition the system name must be set in order to activate LPAR CPU management. WLM needs the LINUX system name in order to handle the LINUX partition. This also applies if you only use the PX qualifier. The sysplex name specified in the service element panel must be the same as the z/OS sysplex name. The system name must be unique in relation to the other system names in the same sysplex. Set the system name by running the modprobe cpi_module system_name=sysname command, where cpi_module is either hwc_cpi or sclp_cpi, depending on the LINUX distribution level. If you change the LINUX system name you must re-IPL to make it known to LPAR CPU Management. VSE/ESA and z/VM set the system name during IPL.

Goals for non z/OS partitions are specified in the WLM service definition. You can define velocity goals, but no discretionary goals. Multiple periods are not supported.

Enabling Dynamic Channel Path Management
When using Dynamic Channel Path Management, your CPC can be running in either basic or LPAR mode. In basic mode, a coupling facility structure is not needed, as there will be no logical partitions grouped together in an LPAR cluster. In LPAR mode, a coupling facility structure is required if you wish to use dynamic channel path management in any logical partition containing a system that is a member of a multisystem complex (even if the system image is the only member of that sysplex on this CPC). You do not need a coupling facility structure if all the logical partitions are running in XCFLOCAL or MONOPLEX mode.

The IBM Redbook z/OS Intelligent Resource Director (SG24-5952) provides extensive guidance on choosing the appropriate channels and control units for Dynamic Channel Path Management. Once you have selected them, there are two specific HCD definitions that will need to be changed:

- **Channel Definitions**
  When defining (or modifying) a channel, you must specify YES in the “Managed” field. You must also specify a dynamic switch (in this case, the ESCON Director) to which the channel is attached. It is recommended that you also specify the entry switch ID and entry port so that HCD can do consistency checking.

  If you are running in LPAR mode, you must also define the name of the sysplex to which a logical partition must belong in order to have this channel in its configuration. Specify this name in the “I/O Cluster” field. Finally, you must also define this channel as being shared. Note that unlike traditional shared channels which potentially can be shared by all logical partitions on a CPC, managed channels can only be shared by logical partitions in the same LPAR cluster.

  **Note:** Ensure that each LPAR cluster name (the sysplex name that is associated with the LPAR cluster) is uniquely named across the entire CPC. As managed channels have an affinity to a specific LPAR cluster, non-unique names would create problems with the scope of control.

- **Control Unit Definitions**
  Whereas non-managed channel paths (otherwise called “static” channel paths) are defined in the traditional way, via the CHPID number, a managed path is defined by specifying a double asterisk (“**”). The number of double asterisks...
you specify will limit the number of managed channel paths per LPAR cluster. The total number of non-managed and managed channel paths, per CPC, cannot exceed 8.

**Important**
You must define at least one non-managed channel path (which must be defined as shared) per control unit.

The control unit must be attached to a switch (again, in this case an ESCON Director) which in turn must be attached to managed channels.

After changing the HCD definitions, the remaining actions to enable Dynamic Channel Path Management occur in the Primary Support Element Workplace, as follows:

- For basic or LPAR mode, ensure that the CPC’s reset profile is enabled for dynamic I/O.
- For an LPAR cluster environment, ensure that all partitions in the cluster are authorized to control the I/O configuration.
- In addition, ensure that the automatic input/output (I/O) interface reset option is enabled in the CPC’s reset profile. This will allow Dynamic Channel Path Management to continue functioning in the event that one participating system image fails.

If at any time you wish to disable Dynamic Channel Path Management, issue the SETIOS DCM=OFF command. The SETIOS DCM=ON command will turn it back on. Issue SETIOS DCM=REFRESH to refresh the control unit model table (for instance, to include a new IOSTNNn load module provided by the control unit’s manufacturer).

**Note:** After issuing the SETIOS DCM=OFF command, your I/O configuration may now be unable to handle your workload needs, as it will now be in whatever state Dynamic Channel Path Management left it before being disabled. You may need to activate a new I/O configuration that will meet your workload needs across the entire LPAR cluster.

### Enabling Channel Subsystem Priority Queuing

To enable Channel Subsystem Priority Queuing, you’ll need to do the following:

- In the WLM ISPF application, make sure that “I/O Priority Queuing” is set to YES on the “Service Coefficients/Service Definition” panel.
- If your CPC is partitioned, click on the “Change LPAR I/O Priority Queuing” icon in the Primary Support Element Workplace. You’ll see a list of all of the logical partitions. Define the range of I/O priorities that will be used by each image, specifying the minimum and maximum I/O priority numbers. It is recommended that you use a range of eight values (for example 8 to 15), as this will correspond to the number of values in the range that WLM uses. While it is not enforced, it is assumed that you’ll set the same range of I/O priorities for all images in the same LPAR cluster. You can, however, prioritize multiple LPAR clusters on the same CPC by setting different ranges for each LPAR cluster.

If you have a partition running a system other than z/OS (for instance z/OS, VM, TPF, VSE) specify an appropriate default priority by setting the minimum and maximum to the same number. For example, if you have a VM partition running work that is equal in importance to the discretionary work in a z/OS partition, you could set the VM partition’s range to 8–8, and the z/OS partition’s
range to 8–15. If you have critical OLTP applications running in a partition, on the other hand, you could set the range for that partition to 16–16, and the z/OS partition’s range to 8–15. In this way, the OLTP work would always have a higher priority than the z/OS work.

- In the Primary Support Element Workplace, click on the “Enable I/O Priority Queuing” icon. You’ll see a simple panel that reads “Global input/output (I/O) priority queuing” — click on the “Enable” box. Channel Subsystem Priority Queuing is now enabled for the entire CPC.

**For More Information**

- For more detailed information on the Intelligent Resource Director, see the IBM Redbook, *z/OS Intelligent Resource Director*.
- For more information on defining coupling facilities, see [*z/OS MVS Setting Up a Sysplex*]
- For more information on PR/SM, see the *PR/SM Planning Guide*.
- For more information on the Primary Support Element Workplace, see the *Support Element Operations Guide*.
- For more information on HCD, see the *z/OS HCD User’s Guide*.
- For more information on RMF, see the *z/OS RMF User’s Guide*.
- For more information on the IEAOPTxx parmlib member, see the *z/OS MVS Initialization and Tuning Reference*.
- For more information on any of the system commands mentioned here, see *z/OS MVS System Commands*. 
Chapter 19. Using zSeries Application Assist Processor (zAAP)

Starting with z/OS V1R6 on z890 and z990 servers, you can run Java applications on a new type of processor called the IBM eServer™ zSeries Application Assist Processor (zAAP). You may see this processor also be referred to as IFA (Integrated Facility for Applications) in information related to zAAPs. zAAPs operate asynchronously with the general purpose processors to execute Java programming under control of the IBM Java Virtual Machine (JVM). This helps reduce the demands and capacity requirements on general purpose processors which may then be available for reallocation to other zSeries workloads. The IBM JVM processing cycles can be executed on the configured zAAPs. zAAPs allow you to integrate and run e-business Java workloads on the same server as your database, helping to simplify and reduce the infrastructure required for Web applications.

Benefits of having zAAPs on a z890: When running standard CPs on the z890 server, there are a wide variety of speeds or MSU ratings available. The zAAPs on a z890 server, however, are always run at full speed. If you have a workload that is heavy in Java execution, you could run that workload on z890 on lower speed standard CPs, along with zAAPs, which would provide significant capacity at a lower cost.

There are no anticipated modifications to Java applications required to use zAAPs.

The steps for starting to use zAAPs and operations considerations are provided in the following. The following tasks relate to using zAAPs:

- Performing capacity planning activities to project how many zAAPs will be needed
- Meeting software and hardware requirements associated with the zAAPs
- Acquiring the zAAPs
- Defining zAAPs to the desired LPARs
- Reviewing parameter settings associated with zAAP usage
- Considering automation changes related to zAAP usage
- Monitoring zAAP utilization and configuring changes appropriately

Performing capacity planning activities to project how many zAAPs will be needed

zAAP Projection Tool

Before you have z/OS V1R6, Java SDK 1.4, or a z890 or z990 server, you can do some capacity planning to determine how many zAAPs you need. There is a projection tool (zAAP Projection Tool) available from http://www.ibm.com/servers/eserver/zseries/zaap which is a modified Java SDK V1R3, that has some of the same functionality that has been incorporated into Java SDK 1.4 and higher. This tool gathers usage information about how much CPU time is spent executing Java code which could potentially execute on zAAPs. By running a Java workload that is representative of the production system operations, it reports, via the Java log, how much of that workload could be eligible for execution on zAAPs. This
information is also useful in predicting the number of zAAPs that might be necessary in order to provide an optimum zAAP configuration.

If you have several systems on the same server on which you are interested in using zAAPs, you can collect the Java log from all the applications running on all the LPARs to have a comprehensive prediction of the total number of zAAPs for a z890 or z990 server. You may choose to run selected Java workloads, and then extrapolate how much total capacity for zAAPs will be required for all the LPARs where you plan to run Java applications.

Meeting software and hardware requirements associated with the zAAPs

To use zAAPs, there are certain software and hardware requirements. The minimum software requirements are:

- z/OS V1R6 (5694-A01)
- IBM SDK for z/OS, Java 2 Technology Edition V1.4 (Product: 5655-I56, Subscription and Service: 5655-I48) with the PTF for APAR PQ86689, or higher.

With regard to the IBM SDK 1.4 requirement, you should be aware that several products include or require the SDK. The level of SDK that has been included or required, may or may not meet the requirements to use zAAPs. For instance if you are a WebSphere Application Server user, WebSphere Application Server V5.0.2 has included and requires the SDK at the V1R3 level, which does not meet this requirement. However, WebSphere Application Server V5.1 has included and requires the SDK at the 1.4 level which does meet this requirement. It is important to understand which SDK level you are using with your products, and to ensure that you meet the requirements when using zAAPs. Failure to run with the required IBM SDK level means that Java workload will not execute on zAAPs.

Hint for planning migration: Because WebSphere Application Server for z/OS, V4.0.1 is not supported on z/OS V1.6 and its service was discontinued on April 30, 2005 (as announced in Software Announcement 904-021, dated February 3, 2004), customers currently running WebSphere V4.0.1 are strongly encouraged to migrate to WebSphere V5.1 now in order to avoid the need to simultaneously upgrade the WebSphere and z/OS levels when z/OS V1.6 becomes available.

The hardware requirements for zAAPs are:
- z890 or z990 server. If you are running on a z990 server, the driver level must be D55
- The hardware management console (HMC) for defining and configuring the zAAPs must be at driver level D55

For information on migrating to z/OS V1R6 on a z890 or z990 server, refer to z/OS Migration.

Acquiring the zAAPs

Contact an IBM representative for purchasing zAAPs for your z890 or z990 server. zAAPs can only be purchased as an additional Processor Unit for your server. That is, you may not convert a standard central processor (CP) you already have acquired to a zAAP.
You may order zAAPs up to the number of permanently purchased CPs, (including z990 unassigned CPs or z890 Downgrade - Record Only CPs), on a given machine model. The number of zAAPs ordered may not exceed the limit of available engines in the machine model.

For the IBM eServer zSeries 990:
- Using the 1 to 1 ratio for planning zAAP on machine - one CP must be installed with or prior to any zAAPs being installed. That means the total number of zAAPs installed may not exceed the total number of CPs plus unassigned CPs installed on the z990.

For the IBM eServer zSeries 890:
- One zAAP may be purchased for one-way z890 capacity settings (110 to 170).
- Two zAAPs may be purchased for two-way z890 capacity settings (210 to 270).
- One zAAP may be purchased for three-way z890 capacity settings.
- There are no available Processor Units (PUs) for purchase as zAAPs on four-way z890 capacity settings (410 to 470).

It is possible to concurrently install temporary capacity by ordering On/Off Capacity on Demand Active zAAPs. The number of On/Off Capacity on Demand zAAPs that you may rent is limited by the number of permanently purchased zAAPs on a given server. On/Off Capacity on Demand zAAPs may not exceed the number of permanently purchased zAAPs on a server.

### Defining zAAPs to the desired LPARs

zAAPs are configured via the normal PR/SM logical partition image profile. There are some requirements regarding zAAPs and standard CPs. The following requirements are associated with zAAPs:
- The number of zAAPs must not exceed the number of standard CPs for a server.
- You must have at least one standard CP defined for a partition. The z/OS system needs at least one standard CP online at all times.
- You have the ability to set the number of zAAPs for an LPAR, although you cannot specify a weight or if the zAAPs are dedicated or shared. The zAAPs inherit the dedicated or shared attribute from the standard CPs in that LPAR.
- There is hard capping for zAAPs, but there is no support for soft capping (which is the WLM support for 4-hour rolling average). The existing single set of PR/SM logical partition processor weights (INITIAL, MIN, MAX) are applied independently to the shared standard CPs - capping only applies to shared standard CPs configured to the logical partition. If you use WLM weight management for the LPAR with SCRT, then z/OS WLM will manage shared standard CPs as today, but not the zAAPs.
- zAAPs will not participate in Intelligent Resource Director. The zAAPs will not participate in dynamic share management or in the number of logical zAAPs online.

Note that zAAPs are brought online and offline like standard CPs.

To define the zAAPs to your logical partitions, refer to [PR/SM Planning Guide](#).

### Reviewing parameter settings associated with zAAP usage

This topic describes the parameter settings associated with zAAP usage.
Review z/OS parameter settings

There are several SRM/WLM options that allow you to control how work is assigned between zAAPs and standard CPs. Java-eligible work that could execute on zAAPs may also execute on standard CPs in order to achieve workload goals.

In parmlib member IEAOPTxx, there are two statements which control the workflow to zAAPs. The default values will allow both zAAPs and standard CPs to be used for receiving zAAP-eligible work, as well as dispatching both zAAP-eligible and non-zAAP-eligible work in priority order.

IFAHONORPRIORITY: If you specify IFAHONORPRIORITY=YES (the default) in IEAOPTxx, you indicate that standard CPs may execute both Java and non-Java work in priority order, if zAAP processors are unable to execute all zAAP-eligible work.

If you specify IFAHONORPRIORITY=NO, then zAAP-eligible work can execute on standard CPs, but at a lower priority than non-Java work.

Refer to "z/OS MVS Initialization and Tuning Reference" for more information on the parmlib member IEAOPTxx.

Review Java parameter settings

When running the JVM at SDK 1.4 or higher, there are options provided to allow you to control Java code execution on zAAPs. The default value will cause Java code to be dispatched to zAAPs. The option is specified with other Java startup options. You can use these JVM options to compare execution results of Java code on standard CPs and zAAPs.

-Xifa:on : This is the default. The -Xifa:on option indicates to the JVM to call the switch service for zAAPs. If there are no zAAPs available, the JVM will silently disable switching. Once the JVM calls the switch service successfully, it will continue calling the switch service, even if the last zAAP goes offline.

-Xifa:force : The -Xifa:force option indicates to the JVM to always call the switching service for zAAPs, even when there are no zAAPs online. As mentioned previously, this option is useful when gathering information for capacity planning.

-Xifa:off : The -Xifa:off option indicates to the JVM to bypass all switching to zAAPs. This causes all Java work to be executed on standard CPs.

Refer to Java Diagnostics Guide for more information on setting the -Xifa startup option in the JVM.

Considering automation changes related to zAAP usage

Based on your IEAOPTxx zAAP settings, there are some automation changes you could make to ensure that Java work will continue to execute in priority order.

Monitoring zAAP utilization and configuring changes appropriately

z/OS V1.6 SMF and RMF have been enhanced to monitor zAAP usage. The same skills you have today for monitoring standard CPs can be used for monitoring zAAPs. There are no zAAP-specific skills required for capacity monitoring.
SMF type 30 and type 72 records have been enhanced to provide zAAP usage information:

- For the SMF 30 records, both the amount of time zAAP-eligible Java work spends executing on zAAP processors and on standard CPs is reported. Job step time provided by SMF 30 reports the amount of standard CP time consumed by the job step and the amount of zAAP-eligible time consumed by the job step executing Java on standard CPs, if any.

  Note that installation exit IEFACTRT is enhanced for R7. If you update the IEFACTRT exit, it allows installations to update their system messages with additional accounting information from the SMF type 30 records. The system messages will report information about the new zAAP fields IFA CPU, enclave on IFA, and dep_enclave on IFA.

  Refer to MVS: Installation Exits for further information about installation exit IEFACTRT.

- For SMF 72 records, the amount of time spent executing on zAAP processors is reported as well as Using and Delay sample counts for zAAP-eligible work.

  When running the same Java workload with zAAPs as you were running before without zAAPs, you should expect to see less capacity shown in your SCRT reports, as well as less capacity used for standard CPs in your RMF reports. If new Java workload has been added, this increases CP usage.

  Refer to z/OS MVS System Management Facilities (SMF) for more information on SMF type 30 and type 72 records.

  Refer to z/OS RMF User’s Guide for more information on RMF monitoring.

  Note that the diagnosis tools and service aids that you use today, for example, SLIP traps and traces, can be used unchanged with respect to zAAPs.
Chapter 20. Using zSeries Integrated Information Processor (zIIP)

Starting with V1R8, z/OS on IBM System z9® Enterprise Class (z9 EC) or IBM System z9 Business Class (z9 BC) servers support integrated information processor (zIIP) - a processor type for a dedicated workload.

Conceptually similar to zSeries Application Assist Processors (zAAP), using zIIPs allows to offload certain workloads, for example, selected DB2 tasks, from CPUs to zIIPs. This can help free up capacity on general purpose processors which may then be available for reallocation to other zSeries workloads. With zIIPs available, DB2 V8, for example, can send eligible work to z/OS and z/OS.e to be offloaded to zIIPs. Thus, using zIIPs helps to optimize resource usage, contributes to cost-effective zSeries exploitation and enhances the role of the mainframe as the data hub of the enterprise.

Meeting Software and Hardware Requirements for Using zIIPs

To use zIIPs, the minimum software requirements are the following:

- z/OS V1R6 (5694-A01), or higher
- DB2 V8 (5675-DB2) with zIIP enabling APARs installed

Starting with z/OS V1R8 the zIIP support is integrated. For systems running z/OS V1R6 and z/OS V1R7, the zIIP web deliverable with the appropriate service APARs need to be installed. For information about which APARs to install, refer to the Program Directory for IBM System z9 Integrated Information Processor Support for z/OS V1R6/R7 Web Deliverable, and to the appropriate zIIP PSP buckets.

To use zIIPs, the hardware requirements are one of the following:

- IBM System z9 Enterprise Class (z9 EC) with IBM System z9 Integrated Processor Feature Code 7815
- IBM System z9 Business Class (z9 BC) with IBM System z9 Integrated Processor Feature Code 7868

Planning for zIIPs

The SYS1.PARMLIB member IEAOPTxx provides statement PROJECTCPU. Specifying the PROJECTCPU parameter allows you to project zIIP (and zAAP) consumption when a zIIP (or zAAP) processor is not yet defined to the configuration. RMF and SMF will show the potential calculated zIIP time, so that an accurate zIIP projection can be made. The PROJECTCPU option can be used while running the target workload, once all software is installed that enables hardware sizing data to be produced.

You can use the DISPLAY M=CPU command to show if a zIIP processor is defined in the configuration. (In the D M=CPU command output zIIPs are represented by the letter "I"). A zIIP processor is considered to be defined in the offline or reserved state, as well as in the online state.

Refer to z/OS MVS Initialization and Tuning Reference for more information on the parmlib member IEAOPTxx.
The SMF type 30 record (IFASMFR3) includes zIIP consumption fields. Refer to z/OS MVS System Management Facilities (SMF) for further information.

The TIMEUSED macro allows zIIP execution time to be requested in addition to the standard CP consumption.

### Acquiring zIIPs

Contact an IBM representative for purchasing zIIPs for your IBM System z9 Enterprise Class (z9 EC) or IBM System z9 Business Class (z9 BC) server. zIIPs can only be purchased as an additional Processor Unit for your server. That is, you may not convert a standard central processor (CP) you already have to zIIP.

You may order zIIPs up to the number of permanently purchased CPs, on a given machine model. The number of zIIPs ordered may not exceed the limit of available engines in the machine model.

It is possible to concurrently install temporary capacity by ordering On/Off Capacity on Demand Active zIIPs. The number of On/Off Capacity on Demand zIIPs that you may rent is limited by the number of permanently purchased zIIPs on a given server. On/Off Capacity on Demand zIIPs may not exceed the number of permanently purchased zIIPs on a server.

### Defining zIIPs

zIIPs are configured via the normal PR/SM logical partition image profile. A z9 EC or z9 BC system can have one zIIP for each CP on the system.

You can define a logical partition to use one or more zIIPs and/or zAAPs with z/OS 1.6 or higher with either of the following combinations:

- One or more dedicated general purpose CPs and one or more dedicated zIIPs/zAAPs
- One or more shared general purpose CPs and one or more shared zIIPs/zAAPs

The mode specified for the logical partition must be set to ESA/390 to allow for the definition of zIIPs or zAAPs to the logical partition.

Furthermore, there are the following requirements regarding zIIPs and standard CPs:

- The number of zIIPs must not exceed the number of standard CPs on a server.
- You must have at least one standard CP defined for a partition. The z/OS system needs at least one standard CP online at all times.
- You can set the number of zIIPs for an LPAR, and each processor pool (for example, CPs and zIIPs) can be assigned a unique weight when the processors are being shared.
- There is hard capping for zIIPs, but there is no support of defined capacity (which is the WLM support for 4-hour rolling average). The existing single set of PR/SM logical partition processor weights (INITIAL, MIN, MAX) are applied independently to the shared standard CPs - capping only applies to shared standard CPs configured to the logical partition. If you use WLM weight management for the LPAR with SCRT, then z/OS WLM manages shared standard CPs as today, but not the zIIPs.
- zIIPs do not participate in Intelligent Resource Director. zIIPs do not participate in dynamic share management or in the number of logical zIIPs online.
Note that zIIPs are brought online and offline like standard CPs.

Combining zIIP-enabled sysplex members with non-zIIP enabled sysplex members is supported.

**Reviewing z/OS parameter settings**

There are several SRM/WLM options that allow you to control how work is assigned between zIIPs and standard CPs. zIIP-eligible work may also execute on standard CPs in order to achieve workload goals.

Parmlib member IEAOPTxx contains the IIPHONORPRIORITY statement which controls the workflow to zIIPs. If you specify IIPHONORPRIORITY=YES this indicates that standard CPs may execute zIIP-eligible and non-zIIP-eligible work in priority order, if zIIP processors are unable to execute all zIIP-eligible work. This is the default.

If you specify IIPHONORPRIORITY=NO, then zIIP-eligible work can execute on standard CPs, but at a lower priority than non-zIIP-eligible work.

Note that the IFACROSSOVER=YES option in IEAOPTxx SYS1.PARMLIB is not applicable for either zIIPs or zAAPs. If specified, it is ignored.

Refer to [z/OS MVS Initialization and Tuning Reference](#) for more information on the parmlib member IEAOPTxx.

**Using zIIPs - Miscellaneous Services**

The WLM IWMEQTME and IWM4EDEL services are enhanced to support zIIP usage. For further information about the new parameters of IWMEQTME and IWM4EDEL, refer to [z/OS MVS Programming: Workload Management Services](#).

Check RMF measurement for zIIP usage statistics. Refer to [z/OS RMF Report Analysis](#) for further information on RMF’s reporting of zIIP.

**Activating zIIPs**

z/OS zIIP support is operational when the SMP/E installation is complete and the target image has been restarted. The zIIP engines can be configured online any time after restarting the logical partition. No further customization is required.
Chapter 21. Using the WLM ISPF Application

This information explains how to use the ISPF application. It explains the functions that are available, and how you can navigate through the panels.

Before You Start

You should prepare at least one service policy and your classification rules to be ready to start using the ISPF application. Your service policies and classification rules make up a service definition. You can store a service definition in the following kinds of data sets:

**WLM couple data set**

In order for all systems in a sysplex to process with an active service policy, they must all be able to access a service policy. They all access the policy from a WLM couple data set. To use workload management, you must allocate a WLM couple data set, define it to the sysplex, and install your service definition onto it. You can allocate the WLM couple data set from the application. Only one service definition can be installed on the WLM couple data set.

**MVS partitioned data set (PDS)**

You do not need to preallocate the data sets. You specify a data set name, and the application allocates it for you. You can save one service definition per MVS PDS.

**Notes:**

1. If you use customized data sets in your installation, or if you use DFSMS, you can use WLM application exits IWMAREX1 and IWMAREX2 to specify those changes. See Appendix A, “Customizing the WLM ISPF Application,” on page 233 for how to code the exits.

2. The data set `userid.WLM.SAVExx` (where `userid` is the TSO ID running the application and `xx` is some numeric value such as `SAVE01`) is allocated by the WLM application for recovery and is deleted by WLM upon exiting the application. This naming convention should therefore **not** be used for a new service definition.

**MVS sequential data set (PS)**

You can store a service definition in XML format in a sequential data set. You need not preallocate the sequential data sets. Specify a data set name, then the application allocates it for you.

**Notes:**

1. If you use customized data sets in your installation, or if you use DFSMS, you can use WLM application exits IWMAREX1 and IWMAREX2 to specify those changes. See Appendix A, “Customizing the WLM ISPF Application,” on page 233 for how to code the exits.

2. The data set `userid.VDEF.TEMP.dddddd.Tttttt` (where `userid` is the TSO ID running the application, `ddddd` is the current date and `tttttt` is the current time) is allocated by the WLM application to temporarily save the XML service definition as ISPF tables during editing. The data set is deleted by WLM upon exiting the application. The naming convention should therefore **not** be used for a new service definition.
Panel Areas and How to Use Them

Most panels have a menu bar, action field, status line, scrollable area, function key area, and command line. You tell the application what actions to perform by making choices or typing information on a panel.

In this topic, examples of panels and pop-ups are shown to help familiarize you with the product. The examples closely match what you see on your terminal, but in some cases the spacing or function key settings may not exactly match what you see on your terminal.

Using the Menu Bar

A menu bar at the top of every panel shows the actions you can take on that panel. Press F10, or the Home key on some terminals, to move the cursor to the beginning of the menu bar from any position on a panel.

Figure 31 shows an example of the menu bar on the Definition menu.

![Figure 31. Menu Bar on the Definition Menu](image)

To select an action, use the Tab or cursor movement keys to position the cursor on your choice, then press ENTER.

When you select an option on the menu bar, WLM displays a pull-down with choices related to the option you selected. While a pull-down is displayed, only the actions in the pull-down or on the menu bar are available. For example, if you select a pull-down and the option you want is not listed in it, you can select another pull-down on the menu bar.

To select an option in a pull-down, type the number of your choice in the action field and press ENTER. You can also use the cursor movement keys to position the cursor on your choice, then press ENTER. Figure 32 shows an example of the pull-down choices on the File option on the Definition Menu.

![Figure 32. Definition Menu File Choices](image)

Using the Menu Bar on Selection Lists

The menu bar is a bit different on selection lists. You type a slash next to the name of the object you want to work with in the action field. Then, move to the menu bar, select an option, and press ENTER. You then choose the desired action from the pull-down.
For example, from the service class selection list, choose the STC_1 service class, move to the menu bar on the Service-Class option, and press ENTER. Then, type 3 in the menu pull-down, and press ENTER.

<table>
<thead>
<tr>
<th>Service-Class</th>
<th>View</th>
<th>Notes</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Create</td>
<td>Service Class Selection List</td>
<td>ROW 14 TO 26 OF 29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Copy</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Modify</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Browse</td>
<td>Create, 2=Copy, 3=Modify, 4=Browse, 5=Print, 6=Delete,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Print</td>
<td>Menu Bar</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Delete</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Exit</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For example, a service class selection list status line such as **Row 1 to 8 of 16** states that displayed in the panel is a list of eight of the 16 service classes contained in
the service definition.

**Using the Status Line**

Some, but not all panels have a status line. A status line is displayed on the right side of a panel or pop-up beneath the title. The status line indicates the number of items or lines currently displayed in a list or topic, and the total size of and your current location in that list or topic.

For example, a service class selection list status line such as **Row 1 to 8 of 16** states that displayed in the panel is a list of eight of the 16 service classes contained in
the service definition.

**Using the Scrollable Area**

On selection list type panels, there is a scrollable area that contains a list or text. In a service class selection list you see a list of service classes. The status line shows how many service classes there are in the list. You scroll backward or forward to move through the list. Selection lists can also be pop-ups, such as selecting a workload from a list to associate it with a service class.

**Using the Menu Bar on a Selection List**

On selection lists, you can select the object you want to work with before you select actions you want to perform from the menu bar.

To mark items for selection in a scrollable area, type a slash (/) over the underscore in front of the listed choice. When you select an menu bar item, the application performs the action for the marked items only.

**Figure 34 on page 186** shows an example of the scrollable area on a service class selection list. The first two service classes are marked with a / for actions on the menu bar.

Chapter 21. Using the WLM ISPF Application 185
Using the Action Field

The action field is where you specify the action to take. On the definition menu, the action field is where you specify the workload management object that you want to work with. The action codes are standard in the application – except on a few selection lists, such as the subsystem type selection list in classification rules, and the service policy selection list. Figure 35 shows the action codes and action field on the subsystem type selection list.

Note: The menu bar pull-down choices for the file match the action codes on the selection lists. So you can choose the method according to your preference.

Using the Command Line

The command line is displayed according to the currently active user profile. You can issue the following commands from the command line:

- `=value` Repeats the previous command.
- `BACKWARD` Scrolls backward.
- `DOWN` Scrolls forward.
- `EXIT` Exits the panel.
- `FORWARD` Scrolls forward.
- `FKA ON|OFF` Determines whether to display the function key area.
- `HELP` Displays the help panel for the displayed panel.
- `KEYLIST` Displays the keylist utility where you can adjust PF key settings.
- `PFSHOW` Shows the PF key settings.
Using the Function Keys

The function key area at the bottom of each panel or pop-up displays actions that you can complete by pressing a function key. When a pop-up is displayed, you can press only the function keys listed in that pop-up, not the keys listed at the bottom of the panel.

Standard actions are assigned to function keys 1 through 12 and are repeated for function keys 13 through 24. The function key assignments can vary slightly, depending on options selected during installation. If you want to customize the key settings for your installation, you can use the KEYLIST utility. See "Customizing the Keylists" on page 240 for more information about how to customize the key settings.

The WLM application displays function keys 1 to 12. If you want to display function keys 13 to 24, or see the standard function key settings, see ISPF Dialog Management Guide and Reference.

Figure 36 shows a sample of the function keys on the Service Class Selection list.

Starting the WLM Application

The application is shipped in the IPCS library. When you start the application, the system needs to concatenate the WLM/IPCS data sets, allocate some data sets, and then invoke the WLM panels.

To start the application, specify:

```
ex 'SYS1.SBLSCLI0(IWMARIN0)' 
```

If you have different data set conventions for your IPCS data sets, or if you use storage managed data sets, you should use the WLM application exits IWMAREX1 and IWMAREX2. For more information about IWMARIN0 and how to customize the application with the WLM exits, see Appendix A, “Customizing the WLM ISPF Application,” on page 233.

Now You’re Started

Upon entry to the interface, the WLM logo panel is displayed. Press ENTER to continue, and the application displays the Specify Definition pop-up as shown in Figure 37 on page 188. You should then specify which service definition you want to work with. You can work with one service definition at a time in the application.
If this is your first time in the application, choose option 3, Create a new service definition. This option brings you to the definition menu, where you can define your service definition. Once you created a new service definition, upon exiting the application, you can either:

- Save the service definition in a PDS as ISPF tables, or in a PS as XML
  The application prompts you for a data set name, allocates the data set, and saves the service definition.
- Install the service definition on the WLM couple data set
  The install option puts the service definition currently displayed in the application out on the WLM couple data set. When installed, any changes you made to the service definition are available when you activate a policy. “Installing and Extracting a Service Definition” on page 222 explains how to install a service definition.
- Discard the service definition.

If you want to edit a service definition previously defined and stored in an MVS data set, choose option 1 Read saved definition.

If you want to work with the service definition on the WLM couple data set, choose option 2 Extract definition from WLM couple data set. You must have previously installed a service definition on the WLM couple data set. “Installing and Extracting a Service Definition” on page 222 explains how to extract a service definition.

### Using the Definition Menu

The definition menu is the central place for entering your service definition. When you set up a service definition, you must enter a service definition name and optionally, a description on the Definition Menu.

Figure 37 on page 189 shows a sample Definition Menu with the service definition name and a description filled in.
When you define your service definition for the first time, you should define it in the following order:

1. **Policies**
   A policy consists of a name, a description, and policy overrides. The first time you set up a service definition, define a policy name and description. If you do not have a business need to change your goals, you can run with one service policy, without any policy overrides.

   You use a policy override only if you have a business need to change a goal for a certain time, such as for the weekend, or for nighttime. You can define your policy overrides once you have defined you service classes.

2. **Workloads**
   A workload logically consists of a group of one or more service classes. You associate a workload with a service class in the Service Class panel. Enter your workloads before creating your service classes.

3. **Resource groups (optional)**
   A resource group is a minimum or maximum amount of processing capacity. You associate a resource group with a service class in the Service Class panel. Enter resource groups before creating your service classes.

4. **Service classes**
   A service class is a group of work with similar performance goals, resource requirements, or business importance. You make the association with a workload and a resource group in the service class panel. You associate a service class with incoming work in the classification rules panel. Enter service classes before creating classification rules.

   **Policy overrides**
   Once you have created a service class, you can create a policy override. You specify the policy override by selecting Service Policies from the Definition Menu, and then specifying the action code for Override service class or Override resource group.

5. **Classification groups (optional)**
   You use groups to simplify classification. You associate a classification group with a service class in the classification rules panel. If you intend to use them,
create groups before creating classification rules. See Chapter 9, “Defining Classification Rules,” on page 57 for descriptions of group qualifiers.

6. **Classification rules**
Classification rules assign incoming work to service classes. Before you create your classification rules, you must understand which subsystem's work is represented in each of your service classes.

When you choose the option Classification Rules, you go to the Subsystem Type Selection List for Rules. This selection list is primed with all of the IBM-Supplied subsystem types. They are reserved names.

7. **Report classes (optional)**
A report class is a group of work for which you want reporting data. You do not have to define report classes before assigning them to work in classification rules. You can create them from within the classification rules menu.

8. **Service coefficients/options**
Service coefficients define the weight to be applied to one type of service over another in the calculation of service rates. You can enter new values for the CPU, IOC, MSO, and SRB service coefficients.

See “Service Definition Coefficients” on page 94 for more information.

There are two additional options on this panel:

- **I/O Priority Management**: The default is no, meaning that I/O priorities will be the same as dispatching priorities. Specifying yes means I/O priorities should be managed separately from dispatching priorities, according to the goals of the work. See “Specifying I/O Priority Management” on page 96 for more information.

- **Dynamic Alias Management**: The default is no, meaning that dynamic alias management is disabled for the entire sysplex. Specifying yes will cause workload management to dynamically reassign parallel access volume aliases to help work meet its goals and to minimize IOS queueing. See “Specifying Dynamic Alias Management” on page 96 for more information.

9. **Application Environments**
An application environment is a group of application functions invoked by request and executed in server address spaces. You can have workload management start and stop these server address spaces automatically, or do this manually or through automation. You define the application environment, an optional procedure name for starting the server address spaces, and any start parameters needed for the start procedure.

10. **Scheduling Environments**
A scheduling environment is a list of resource names along with their required states. By associating incoming work with a scheduling environment, you ensure that work is assigned to a system only if that system satisfies all of the requirements. You define the scheduling environment, listing all of the resource names and required states that are contained within. You also define the resource names themselves.

11. **Guest Platform Management Provider (GPMP)**
Starts the guest platform management provider (GPMP) to allow for performance management of zEnterprise systems. For further information about the guest platform management provider, refer to Chapter 15, “Managing Performance with the Guest Platform Management Provider (GPMP)” on page 135.
Using the Menu Bar on the Definition Menu

The menu bar on the Definition Menu has some functions not accessible from any other panel in the application. From the menu bar, you:

- Verify a service definition
- Allocate a WLM couple data set
- Install a service definition on the WLM couple data set
- Activate a service policy

Table 18 shows the options available from the menu bar on the Definition Menu. Each of the options are explained.

<table>
<thead>
<tr>
<th>File</th>
<th>Utilities</th>
<th>Notes</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Install definition</td>
<td>Edit notepad</td>
<td>Process ISPF list data set</td>
<td>General Help</td>
</tr>
<tr>
<td>Open</td>
<td>Extract definition</td>
<td></td>
<td></td>
<td>Keys Help</td>
</tr>
<tr>
<td>Save</td>
<td>Activate service policy</td>
<td></td>
<td></td>
<td>Using Help</td>
</tr>
<tr>
<td>Save as</td>
<td>Allocate couple data set</td>
<td></td>
<td></td>
<td>Tutorial</td>
</tr>
<tr>
<td>Print</td>
<td>Allocate couple data set</td>
<td></td>
<td></td>
<td>About...</td>
</tr>
<tr>
<td>Print as GML</td>
<td>using CDS values</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cancel</td>
<td>Validate definition</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exit</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

File

New Use new to define a new service definition.

Open Use open to read a previously defined service definition. The Read saved definition panel is displayed, where you can specify the data set name.

Save Use save to save the currently displayed service definition.

Save as

Use Save as to save the currently displayed service definition in a PDS as ISPF tables or in a PS as XML. The Save to... panel is displayed where you can specify the data set name and the save format. You do not need to preallocate the data set. If the data set does not exist, the application displays the Create Data Set? panel where you can continue with the data set create.

Print Use Print to print the complete service definition to the ISPF list data set. Use the Options menu bar option to process the ISPF list data set. This option requires no formatting step.

Print as GML

Use Print as GML for a more readable, tabular display of service definition objects and values. This option creates a source data set with GML starter set tags imbedded. The data set must be allocated as variable block with logical record length 255. This data set can then be formatted with the SCRIPT/VS processor.

Cancel Use cancel to cancel any actions performed. Cancel is the same as using the cancel PF key.

Exit Use exit to exit from the definition menu and the application. Exit is the same as using the exit PF key.
Utilities

Install definition
Use this option to install the service definition onto the WLM couple data set. Installing the service definition makes any changes available for policy activation.

Extract definition
Use this option to extract the service definition previously installed on the WLM couple data set.

Activate service policy
Use this option to activate a policy. When you select this option, the application displays a list of the service policies defined in the service definition currently installed on the WLM couple data set. You activate the service policy by selecting it from the list.

Note: If you have just made changes for a service definition, make sure you install it to have the changes take effect.

Allocate couple data set
Use this option to allocate both your primary and alternate WLM couple data sets. This option is for users who are allocating a WLM couple data set for the first time.

All other users should use the option "Allocate couple data set using CDS values". To make the WLM couple data set available for use in the sysplex, you must update your COUPLExx parmlib member and issue the SETXCF command.

Allocate couple data set using CDS values
Use this option to allocate both your primary and alternate WLM couple data sets based on your existing WLM couple data set size. The application displays the current size values on the panel.

To make the WLM couple data set available for use in the sysplex, you must update your COUPLExx parmlib member and issue the SETXCF command.

Validate definition
Use this option to verify that your service definition is free from certain errors that would otherwise be flagged when you attempt to save or install the service definition.

Notes
Edit notepad
Use this option to create and edit a notepad. You can use the notepad to keep track of changes made to all parts of a service definition.

Options
Process ISPF list dataset
Use this to process the list data set if you have previously done a Print.

Help
General Help
Use general help for information about the panel currently displayed.

Keys Help
Use keys help for information about the using PF keys.
Using Help
Use this option for information about how to get help while using the WLM application.

Tutorial
Use the tutorial option for information about how to use the panels. This option provides context-specific examples and scenarios.

About...
This option provides information about the copyright and license.

Working with Service Policies
When you choose the Policy option for the first time, the application displays the Create a Service Policy Menu. Figure 39 shows a sample panel.

<table>
<thead>
<tr>
<th>Service-Policy Notes Options Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>----------------------------------</td>
</tr>
<tr>
<td>Create a Service Policy</td>
</tr>
<tr>
<td>----------------------------------</td>
</tr>
<tr>
<td>Enter or change the following information:</td>
</tr>
<tr>
<td>Service Policy Name . . . . . . . . (Required)</td>
</tr>
<tr>
<td>Description . . . . . . . . . .   (Required)</td>
</tr>
</tbody>
</table>

Command ===> ___________________________________________________________

Figure 39. Create a Service Policy

Once you have created a service policy, any other time you choose the policy option from the definition menu, the application displays a policy selection list. From here, you can modify your policy description, print and browse your service policies, and define your service policy overrides. Figure 40 on page 194 shows a Service Policy Selection List.
Working with Workloads

When you choose the Workload option for the first time, the application displays the Create a Workload panel. Figure 41 shows a Create a Workload panel.

You associate the workload with a service class in the service class panel.

Once you have created a workload, any other time you choose the workload option from the definition menu, the application displays a workload selection list. The Workload Selection List is similar to the Policy Selection List. From here, you can modify your workload description, print, and browse your workloads. Figure 42 on page 195 shows a Workload Selection List.
Working with Resource Groups

To define a resource group, choose option 3 on the Definition menu. Define a name, a description (optional), a resource group type (1, 2, or 3), a minimum and/or maximum capacity. Associate the resource group with a service class in the service class panel. Figure 43 shows a Create a Resource Group panel.

As with a workload, once you have created a resource group, any other time you choose the resource group option from the definition menu, the application displays a selection list. From here, you can modify your resource group description, as well as print, and browse it.
Working with Service Classes

Once you have defined your workloads and resource groups, you can define your service classes. Choose the Service Class option on the definition menu. Figure 44 shows a Create a Service Class panel. You must assign a workload to the service class in the Workload field.

Use the “Cpu Critical” field to specify CPU protection for critical regions.

Important
The use of these options limits WLM's ability to manage the system. This may affect system performance and/or reduce the system's overall throughput.

Defining Goals
To enter the goal information, enter an i in the Action field, as shown in Figure 44. The goal selection pop-up is displayed, as shown in Figure 45. From this panel, select the type of goal you want to assign to the service class.

Choose a goal type for period 1
1. Average response time
2. Response time with percentile
3. Execution velocity
4. Discretionary

When you choose option 1, average response time, the application displays the average response time goal pop-up. There is a different pop-up for each goal type where you can fill in the information for the goal. If you are defining a single period goal, then you should not fill in a duration. If you are defining multiple
periods then you must fill in a duration. Figure 46 shows an average response time goal pop-up.

<table>
<thead>
<tr>
<th>Average response time goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter a response time of up to 24 hours for period 1</td>
</tr>
<tr>
<td>Hours ....... _ (0-24)</td>
</tr>
<tr>
<td>Minutes ....... _ (0-99)</td>
</tr>
<tr>
<td>Seconds ....... <em>5</em>___ (0-9999)</td>
</tr>
<tr>
<td>Importance . . 2 (1=highest, 5=lowest)</td>
</tr>
<tr>
<td>Duration . . . ________ (1-999,999,999, or none for last period)</td>
</tr>
</tbody>
</table>

**Figure 46. Average Response Time Goal Pop-Up**

When you press EXIT, you return to the create a service class panel with the goal information filled in, as shown in Figure 47.

**Service-Class Notes Options Help**

<table>
<thead>
<tr>
<th>Create a Service Class ROW 1 TO 2 OF 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Class Name...... QUACK (Required)</td>
</tr>
<tr>
<td>Description ........... ________________________________</td>
</tr>
<tr>
<td>Workload Name ......... APPC (name or ?)</td>
</tr>
<tr>
<td>Base Resource Group ..... ________ (name or ?)</td>
</tr>
<tr>
<td>Cpu Critical ......... NO (YES or NO)</td>
</tr>
</tbody>
</table>

Specify BASE GOAL information. Action Codes: I=Insert new period, E=Edit period, D=Delete period.

---Period--- ---------------------Goal---------------------
Action # Duration Imp. Description
_ __ 1 2 Average response time of 00:00:05.000

**Figure 47. Create a Service Class Panel**

**Using Action Codes on Service Class Panels**

You use action codes on this panel to define and edit goals. Figure 48 shows the edit codes available for service class.

**Figure 48. Action Codes for Goal**

**I=Insert new period**

Use I to define a new period. The application adds a line below. If you have multiple periods, then a duration is required on the previous period. Use
action code E to edit the previous period. You'll go through the windows with the goals information filled in, and you can add a duration.

**E=Edit period**
Use E to edit a period.

**D=Delete period**
Use D to delete a period. If you have defined multiple periods for a service class, remember that you do not define a duration for the last period.

---

### Defining Service Policy Overrides

You define all service policy overrides from the service policy selection list. You can define three kinds of service policy overrides:
- Override service class goals
- Override resource group assignment
- Override resource group attributes.

To override a service class goal, choose either the action code or the menu bar option to Override Service Classes. Figure 49 shows a service policy selection list where you have chose to override the service classes for the weekend policy.

![Figure 49. Service Policy Selection List](image)

The application displays the Override Service Class Selection List. This is the list of all of your defined service classes, similar to the Service Class Selection list, except it has an extra field called Overridden Goal which indicates whether or not the service class goal or resource group assignment was overridden for that policy. Figure 50 on page 199 shows an Override Service Class Selection List. Select a service class whose goal or resource group assignment you want to change and specify the action code or menu bar option to override the service class.
When you choose the override service class option, the application displays the Override Attributes for a Service Class panel. Figure 51 shows an Override Attributes for a Service Class panel. To override the goal, use the same codes to edit the goal as you do on the Create or Modify a Service Class panel. You can also change the “Cpu Critical” setting, just as you do on the Create or Modify a Service Class panel. To change the resource group assignment of the service class, either enter in a resource group name, or put a ? in the resource group field to select a resource group from the selection list.

If you want to remove a service from a resource group, blank out the name.

Once you have edited your goal or changed the resource group assignment, press EXIT, and you return to the Override Service Class Selection List. The Overridden Goal field for that service class now says YES.
Working with Classification Rules

Classification of work depends on having the rules defined for the correct subsystem type. When you choose the Classification Rules option from the Definition Menu, you go to the Subsystem Type Selection List for Rules panel. This panel initially contains the reserved names of the IBM-supplied subsystem types.

Although you may want to change the description of the subsystem types, you should not delete any of the entries provided by IBM unless your installation does not plan to ever use them. If your installation later does need to use them, they can be manually added at that time.

Figure 52 shows the subsystem type selection panel:

To create your rules, use the Modify option (3) to create rules for the IBM-supplied subsystem types. For example, use Modify on subsystem type IMS as show in Figure 52 to create the rules for your IMS work. Figure 53 on page 201 shows the Modify Rules for the Subsystem Type panel.
Note that the “Fold qualifier names” option, set to the default Y, means that the qualifier names will be folded to upper case as soon as you type them in and then press Enter. If you set this option to N, then the qualifier names will remain in the case they are typed in. Leave this option set to Y unless you know that you need mixed case qualifier names in your classification rules.

While creating rules, you can scroll right (PF11) to fill in the description fields. Figure 54 shows the Modify Rules for the Subsystem Type panel, scrolled right to the description fields.

For ASCH, CICS, IMS, JES, OMVS, STC, and TSO work, you can scroll right (PF11) yet again to fill in the “Storage Critical” field. For JES and STC work only, you can also fill in the “Manage Regions Using Goals Of:” field, with a value of either TRANSACTION, in which case the region will be managed to the transaction.
response time goals, or **REGION**, in which case the region is managed to the goal of the service class assigned to the region. Note that for ASCH, CICS, IMS, and OMVS work, the “Manage Regions Using Goals Of:” field will read **N/A**. See [Chapter 12, “Defining Special Protection Options for Critical Work,”](#) on page 99 for more information on using these fields.

**Important**
The use of these options limits WLM’s ability to manage the system. This may affect system performance and/or reduce the system’s overall throughput.

---

**Figure 55** shows the **Modify Rules for the Subsystem Type** panel, scrolled right to the “Storage Critical” and “Manage Regions Using Goals Of:” fields.

<p>| Subsystem Type Xref Notes Options Help | Row 1 to 2 of Command ===| SCROLL ===| PAG |
|----------------------------------------|--------------------------|---------------------------|
| Command ===| Modify Rules for the Subsystem Type | Row 1 to 2 of | SCROLL ===| PAG |
| Subsystem Type .: STC | Fold qualifier names? Y (Y or N) | Description ..: IBM-defined subsystem type |
| Action codes: A=After C=Copy M=Move I=Insert rule | B=Before D=Delete row R=Repeat IS=Insert Sub-rule | More |</p>
<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Qualifier</th>
<th>Name</th>
<th>Start</th>
<th>Critical</th>
<th>Storage</th>
<th>Manage Region</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TN</td>
<td>COMBL*</td>
<td>___</td>
<td>____</td>
<td>NO</td>
<td>TRANSACTION</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>UI</td>
<td>COMBLD</td>
<td>___</td>
<td>____</td>
<td>NO</td>
<td>TRANSACTION</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>UI</td>
<td>COMFTP</td>
<td>___</td>
<td>____</td>
<td>YES</td>
<td>TRANSACTION</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>UI</td>
<td>SRVLIB</td>
<td>___</td>
<td>____</td>
<td>YES</td>
<td>TRANSACTION</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>PF</td>
<td>99</td>
<td>___</td>
<td>____</td>
<td>YES</td>
<td>REGION</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>TC</td>
<td>B</td>
<td>___</td>
<td>____</td>
<td>NO</td>
<td>REGION</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>TC</td>
<td>C</td>
<td>___</td>
<td>____</td>
<td>NO</td>
<td>REGION</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 55. Modify Rules for the Subsystem Type, Scrolled Right to Storage Critical and Region Management Fields**

Keep your classification rules as simple as possible. Use subsystem defaults to cover most cases, and list the exceptions. Remember that the rules are order sensitive, and the first matched rule applies. The Start field applies only to the accounting information and subsystem parameter qualifiers.

If you have a non-IBM supplied subsystem type, you can use Create to create your subsystem type. You can also modify the qualifier selection list to include only the qualifier types that apply to your subsystem type.

For more information about defining the order of rules, and how to use the Start field, see [Chapter 9, “Defining Classification Rules,”](#) on page 57.

**Using Action Codes on the Modify Rules Panel**
You use action codes on this panel to create and change your classification rules.
In the Qualifier Type field, enter the type of qualifier that you are defining. A qualifier type can be one of the following:

- **AI**: Accounting information
- **CI**: Correlation information
- **CN**: Collection name
- **CT**: Connection type
- **ESC**: zEnterprise service class name from a Unified Resource Manager performance policy
- **CTG**: Connection type group
- **LU**: LU name
- **LUG**: LU name group
- **NET**: Net ID
- **NETG**: Net ID group
- **PC**: Process name
- **PF**: Perform
- **PFG**: Perform group
- **PK**: Package name
- **PKG**: Package name group
- **PN**: Plan name
- **PNG**: Plan name group
- **PR**: Procedure name
- **PRI**: Priority
- **PX**: Sysplex name
- **SE**: Scheduling environment name
- **SI**: Subsystem instance
- **SIG**: Subsystem instance group
- **SPM**: Subsystem parameter
- **SSC**: Subsystem collection name
- **SY**: System name
- **SYG**: System name group
- **TC**: Transaction class
- **TCG**: Transaction class group
- **TN**: Transaction name
- **TNG**: Transaction name group
- **UI**: Userid
- **UIG**: Userid group

You can either enter the qualifier type, or select it from a selection list. To select it from a list, enter a ? in the Qualifier Type field. "Using Selection Lists for Classification Rules" on page 204 explains more about using selection lists.

Then type in the name of the qualifier in the Name field, and the service class in the Service Class field.
You can use the action codes on this panel to enter in additional classification rules. The action codes are:

**I=Insert rule**
To create a rule at the same level as the rule you type the action code next to.

**IS=Insert Sub-rule**
To create a next level rule or a nest. IS=Insert Sub-rule specifies a rule under the previous rule; that is, a rule at the next indented level.

**R=Repeat**
To copy a rule and its sub-rules, if any. The copied rule is placed directly below the one you are copying. If a rule has sub-rules; that is, it is a family, the entire family is copied.

**D=Delete row**
To delete a rule.

**A=After**
Use with C=Copy and M=Move to specify that the copied/moved rule is to go after this rule.

**B=Before**
Use with C=Copy and M=Move to specify that the copied/moved rule is to go before this rule.

**C=Copy**
To copy a rule and its sub-rules, if any. Use B=Before or A=After with C=Copy to specify where to copy the rule. The rule is copied at the same level as the rule you are placing it Before or After. For example, a level 3 rule becomes a level 2 rule if you copy it before or after a level 2 rule.

When a rule changes levels as a result of a Copy, the levels of any sub-rules that were copied with it are bumped up or down accordingly. The rule family is copied at the same level as the Before or After rule.

**M=Move**
To move a rule and its sub-rules, if any. Use B=Before or A=After with M=Move to specify where to move the rule. The rule is moved according to the way you do a copy, by placing it Before or After.

### Using Selection Lists for Classification Rules

For a list of applicable qualifiers for the subsystem type, put a ? in the qualifier type field. You can select a qualifier from the list by putting a / next to the one you want to use.

For a list of the defined service classes put a ? in the service class field. You can select a service class from the list by typing a / next to the one you want to use.

You can define a report class by typing the report class name in the report class field. A pop-up panel lets you confirm the create. For a list of the defined report classes, put a ? in the report class field.

### Creating a Subsystem Type for Rules

Use the Create option only if your installation has its own subsystem type, or a vendor subsystem type that supports workload management. You must check that product's documentation for its reserved subsystem type name.
Deleting a Subsystem Type for Rules

Use the delete option on the Subsystem Type Selection List for Rules only to remove an IBM-supplied subsystem type that your installation does not have, or does not plan on using. The application displays a pop-up for confirmation of the delete.

Working with Classification Groups

If you have a long list of work that you want to use in a classification rule, you can create a group. You can create groups for the following qualifier types:
- Connection Type
- LU Name
- Net ID
- Package Name
- Plan Name
- Perform
- Subsystem Instance
- System Name
- Transaction Class
- Transaction Name
- Userid Groups

For example, you may want to create a transaction name group to use in classification rules for your started tasks. Figure 57 shows a Create a Group panel. You can use wild-card and masking notation in qualifier groups. Note that there is also room next to each classification group for a description.

Figure 57. Create a Group

Then, you go to modify the STC subsystem on the Subsystem Type Selection List for Rules panel, and reference the group in the rules. Choose transaction name group for the qualifier type, and enter the name of the group. You can also type a ? in the qualifier name field for a list of the defined groups.

Figure 58 shows a Modify Rules for STC Subsystem panel. On this panel, you specify TNG in the qualifier type field. Then you can enter the transaction group name STC_GR1 in the qualifier name field.
You can create a report class by either:
- Selecting the report classes option on the Definition Menu
- Typing in a name on the report class field of the Modify Rules for the Subsystem Type panel.

When you select the report classes option on the Definition menu, you go to the Create a Report Class panel. From there, you enter the report class name and description. The next time you choose the option from the Definition Menu, the application displays Report Class selection list.

To create a report class from the Modify Rules for Subsystem panel, enter a name in the report class field. Figure 58 shows a Modify Rules for the CICS Subsystem panel. If you type in a name, for example CICSHOT, in the report class field, the application displays a Create Report Class pop-up panel. Figure 60 on page 207 shows a Create a Report Class pop-up.

---

**Working with Report Classes**

---

Figure 58. Modify Rules for STC Subsystem

---

Figure 59. Creating a Report Class on the Modify Rules Panel
When you press ENTER, the report class is created, and is added to the report class selection list. You can type a ? in the report class field on the Modify Rules for a Subsystem Type panel for a selection list of report classes.

Working with Service Coefficients and Options

On the Service Coefficient/Service Definition Option panel, you can specify the service definition coefficients, whether you want workload management to manage your I/O priorities, and whether you want workload management to manage your parallel access volume alias addresses.

Chapter 11, “Defining Service Coefficients and Options,” on page 93 provides some advice on how and when to adjust your coefficients.

To find out what service definition coefficients you are currently running with, check your RMF Monitor I Workload Activity Report. This report lists the service definition coefficients your installation is currently using.

If you want workload management to manage I/O priorities for you, specify YES for I/O priority management. When you specify YES, workload management manages your I/O priorities and includes I/O usings and delays in its execution velocity calculation. When you specify NO (which is the default), workload management sets the I/O priority to be the same as the dispatching priority, and I/O usings and delays are not included in the execution velocity calculation.

If you want workload management to manage parallel access volume alias addresses for you, specify YES for Dynamic alias management. When you specify YES, workload management dynamically reassigns alias addresses from one base to another to help work meet its goals and to minimize IOS queueing. See “Specifying Dynamic Alias Management” on page 96 for more information on this global setting and its relationship to the WLMPAV= setting on each individual device. When you specify NO, which is the default, dynamic alias management is globally disabled for the entire sysplex. Systems will still use the aliases assigned to the base devices, but there will be no automatic reassignment of aliases based on goals or queueing.

Figure 61 on page 208 shows the Service Coefficient/Service Definition Options panel.
Working with Application Environments

Define an application environment so that workload management can assist in managing work that runs in server address spaces. You can specify that workload management should dynamically start and stop server address spaces to process work running in the application environment. You can define application environments for the DB2, SOM, and IWEB IBM-supplied subsystem types. Refer to subsystem reference information for guidance on how to specify application environments for the subsystem.

Figure 62 shows the Application Environments panel.

Working with Scheduling Environments

Use the scheduling environment panels to define scheduling environments and their lists of resource names and required states, and to define the individual resource names themselves.
Creating a New Scheduling Environment

If you have not yet defined any scheduling environments, Figure 63 shows the first panel you will see.

![Figure 63. Decide What to Create Panel](image)

From here you can either enter a **1** to create a scheduling environment, or a **2** to create resources. (You might want to create all of your resources, for instance, before creating your first scheduling environment. This is covered in "Creating a New Resource Directly from the Main Panel" on page 213.)

**Note:** Until you have created at least one scheduling environment (no matter how many resources you create), you will always see this initial panel.

Figure 64 shows the main Scheduling Environment Selection List panel. Assuming that you have at least one scheduling environment already created, this is the first panel you see when you work with scheduling environments.

![Figure 64. Scheduling Environment Selection List Panel](image)

By entering a **1** on any Action line, you will go to the Create a Scheduling Environment panel, as shown in Figure 65 on page 210 (You can also get there by putting the cursor under the Scheduling-Environment field at the top of the screen, and then entering a **1** in the pop-up box.)
Enter the scheduling environment name and description, and then enter an A on the Action line to go to the next panel, Resource Definition List, as seen in Figure 66.

This panel shows all of the resource definitions in the service definition. The YES in the In Use field simply means that the resource name is currently a part of some scheduling environment. (It does not mean that it is in use by the scheduling environment you are defining when you get to this screen, in this case DB2LATE.)

You can select as many resources as you like. In this case, we'll select DB2A and PRIMETIME. Both have already been defined (although only PRIMETIME is currently being used by another scheduling environment). To make the resource name part of the scheduling environment, put an S on the Action line next to it. (Alternately, you can put a / next to the resource names and then put the cursor under the Resources field at the top of the screen and hit ENTER. Then select 2 from the options in the pop-up box.)

If you have many resources, you can use the LOCATE primary command to scroll the display to a particular resource name.

Note: You can create new resources from this panel, as well. See "Creating a New Resource While Working with a Scheduling Environment" on page 215. This will take you back to the Create a Scheduling Environment panel, as shown in Figure 67 on page 211.
The DB2A and PRIMETIME names and descriptions are automatically brought back to this panel. Now you must specify either \textbf{ON} or \textbf{OFF} under the Required State field. In this example, we have chosen \textbf{ON} for DB2A and \textbf{OFF} for PRIMETIME (meaning that the DB2A resource state must be set to \textbf{ON} and the PRIMETIME resource state must be set to \textbf{OFF} for this scheduling environment to be satisfied).

**Modifying a Scheduling Environment**

Once a scheduling environment is created, you can modify it at any time, either changing the required states of resources, adding new resources to the scheduling environment, or deleting resources from the scheduling environment.

Now that DB2LATE has been created in the previous section, here's how we would modify it. From the main Scheduling Environment Selection List panel, enter a 3 in the \textbf{Action} line next to the scheduling environment you want to change, as shown in Figure 68 (Alternately, you can enter a / next to it, put your cursor under the Scheduling-Environments field at the top of the screen, and then select 3 from the pop-up box.)

If you have many scheduling environments, you can use the LOCATE primary command to scroll the display to a particular name.

This will take you to the Modify a Scheduling Environment panel. Here, you can add and delete resources, and change required states. In Figure 69 on page 212, we could delete DB2A, and change the required state for PRIMETIME from \textbf{OFF} to \textbf{ON}. (To add new resources from this panel, see "Creating a New Resource While Working with a Scheduling Environment" on page 215.) When you press enter, you'll get a confirmation message, telling you to press \textbf{EXIT} to save your changes,
or CANCEL to discard them.

Important! When you “delete” a resource name from this panel, you are simply deleting it from the scheduling environment. The resource name will still exist on the Resource Definitions List panel. To delete the resource definition itself, see “Deleting a Resource” on page 218.

Copying a Scheduling Environment

To copy a scheduling environment, from the main Scheduling Environment Selection List panel (as shown in Figure 68 on page 211) enter a 2 on the Action line next to the name. This will bring you to the Copy a Scheduling Environment panel, as shown in Figure 70.

You must give the new scheduling environment a new, unique name. From this screen, you can modify the new scheduling environment, as shown in “Modifying a Scheduling Environment” on page 211.

The copy function is useful when you are working with scheduling environments with large numbers of resource names. If you wish to create a new scheduling environment that is similar to an existing one, you can simply copy the original and then make whatever changes are necessary.

Browsing a Scheduling Environment

To browse a scheduling environment, from the main Scheduling Environment Selection List panel (as shown in Figure 68 on page 211) enter a 4 on the Action line next to the name. You'll see a browse screen as shown in Figure 71 on page 213.
Printing a Scheduling Environment
To print a scheduling environment, from the main Scheduling Environment Selection List panel (as shown in Figure 68 on page 211) enter a 5 on the Action line next to the name. The output will be written to your ISPF list data set.

Deleting a Scheduling Environment
To delete a scheduling environment, from the main Scheduling Environment Selection List panel (as shown in Figure 68 on page 211) enter a 6 on the Action line next to the name. This will bring you to a confirmation screen, as shown in Figure 72.

Creating a New Resource
In the example shown in Figure 66 on page 210, as we were creating a new scheduling environment (DB2LATE), we chose resource names (DB2A and PRIMETIME) that already existed. What if we had needed to add a new resource name called VECTOR? As you saw on that Resource Definition List panel, VECTOR did not yet exist.

There are two ways to add a new resource definition, as follows:

Creating a New Resource Directly from the Main Panel
From the main Scheduling Environment Selection List panel, put the cursor under the Resources field at the top of the screen and hit ENTER. When the pop-up box appears, enter a 1, as shown in Figure 73 on page 214. (In this pop-up, 1 is the only
option.)

This will take you directly to the Resource Definition List panel, where you can add new resources. Enter an A on any Action line, as shown in Figure 74. (Alternately, you can put a / next to the resource names and then put the cursor under the Resources field at the top of the screen and hit ENTER. Then select 1 from the options in the pop-up box.)

This would take you to the Define Resource panel, where you would enter the information for the new resource name, as shown in Figure 75.

When we return to the Resource Definition List panel, the new resource name is there, as shown in Figure 76 on page 215.
Note that the panel automatically alphabetizes the resource names. Note also that VECTOR is not yet shown as In Use. We would need to go create or modify a scheduling environment and then select VECTOR.

Creating a New Resource While Working with a Scheduling Environment

While you are in the middle of creating or modifying a scheduling environment, you can take this shortcut method to create new resources. In this case where you wanted to create the new VECTOR resource and add it to DB2LATE, this method would be the logical choice.

From either the Create a Scheduling Environment panel or the Modify a Scheduling Environment panel (they look very similar), enter an A on any Action line, as shown in [Figure 77].

This would take you to the Resource Definition List panel, where you can enter an A on any Action line, as shown in [Figure 78 on page 216]. (Alternately, you can put a / next to the resource names and then put the cursor under the Resources field at the top of the screen and hit ENTER. Then select 1 from the options in the pop-up box.)
As in the method described in "Creating a New Resource Directly from the Main Panel" on page 213, from here you'll go to the Define Resource panel (as was shown in Figure 75 on page 214) to enter the new resource name and description.

When you come back to the Resource Definition List panel, you can now select the new VECTOR resource to be part of the DB2LATE scheduling environment, as shown in Figure 79.

Having done that and exiting, you come back to the Modify a Scheduling Environment panel, as shown in Figure 80.
Now the VECTOR resource name is part of the DB2LATE scheduling environment. As always, you need to specify either ON or OFF under the Required State. In this case, we chose ON.

**Summary of the Two Methods for Creating New Resources**

You can create new resources in two ways:

- **Directly from the main panel** ("Creating a New Resource Directly from the Main Panel" on page 213). This method is useful when you want to create resources without actively working on one particular scheduling environment. Also, note that when you go to the Resource Definition List panel with this method, the Action Codes look like this:
  
  Action Codes: A=Add  D=Delete  X=XREF /=Menu Bar

  There is no S=Select, because you are not actively working on any scheduling environment (and therefore there is no place for you to select resources to).

  There is a D=Delete action code. This will be important when you are deleting resources, as in "Deleting a Resource" on page 218.

- **While creating or modifying scheduling environments** ("Creating a New Resource While Working with a Scheduling Environment" on page 215). This method is useful when you want to create resources while in the midst of creating or modifying a scheduling environment. Note that when you go to the Resource Definition List panel with this method, the Action Codes (and the line above it) look like this:

  Action Codes: A=Add  S=Select  X=XREF /=Menu Bar

  The name of scheduling environment that you are currently working with appears on this panel, and there is a S=Select among the action codes. You can select the new resource to be a part of the scheduling environment immediately after creating it.

  There is no D=Delete action code, because you cannot delete resources when you come the Resource Definition List panel this way. See "Deleting a Resource" on page 218.

**Showing All Cross-References for a Resource Definition**

You can check which specific scheduling environments use a given resource definition by entering an X in the action column next to the resource name, as shown in Figure 81. (You can use either path to get to the Resource Definition List path, as discussed in "Summary of the Two Methods for Creating New Resources.")

```
Resources  Notes  Options  XREF  Help
------------------------------------------------------------------------
<table>
<thead>
<tr>
<th>IWMAPAS</th>
<th>Resource Definition List</th>
<th>Row 1 to 4 of 4</th>
</tr>
</thead>
</table>
Command ===>

Selection For Scheduling Environment DB2LATE

Action Codes: A=Add  S=Select  X=XREF /=Menu Bar

<table>
<thead>
<tr>
<th>Action</th>
<th>Resource Name</th>
<th>In Use</th>
<th>Resource Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>CICSS001</td>
<td>YES</td>
<td>CICS Subsystem</td>
</tr>
<tr>
<td>-</td>
<td>IMS1</td>
<td>YES</td>
<td>IMS Subsystem</td>
</tr>
<tr>
<td>X</td>
<td>PRIMETIME</td>
<td>YES</td>
<td>Peak Business Hours</td>
</tr>
<tr>
<td>-</td>
<td>VECTOR</td>
<td></td>
<td>Vector Processor</td>
</tr>
</tbody>
</table>

Figure 81. Resource Definition List Panel
This will take you to the Resource Cross-Reference of Scheduling Environments panel, as shown in Figure 82.

Deleting a Resource

Unlike adding a resource definition, there is only one path you can take to delete a resource definition: from the main Scheduling Environment Selection List panel (as discussed in “Summary of the Two Methods for Creating New Resources” on page 217).

Put the cursor under the Resources field at the top of the screen and hit ENTER. When the pop-up box appears, enter a 1, as was shown in Figure 73 on page 214. (In this pop-up, 1 is the only option.)

This will take you directly to the Resource Definition List panel. Enter a D on the Action line next to the resource definition you wish to delete, as shown in Figure 83.

Note that you can only delete a resource definition that is not currently in use by any scheduling environment. If you attempted to delete the PRIMETIME resource in Figure 74 on page 214, you would receive an error message stating the resource cannot be deleted because a scheduling environment is using it. You must first check the cross-references for that resource definition (as in “Showing All Cross-References for a Resource Definition” on page 217) and then go into every scheduling environment that uses it to delete those references.
Working with the Guest Platform Management Provider (GPMP) Settings

On z/OS system images, the guest platform management provider (GPMP) is the interface between the z/OS Workload Manager (WLM) and the IBM zEnterprise Unified Resource Manager. The GPMP supports platform management functions and provides policy information to WLM about the platform-wide performance goals of workloads in which the z/OS system is participating.

Figure 84 shows the guest platform management provider (GPMP) settings panel:

![GPMP-Settings Panel](image)

Figure 84. Guest Platform Management Provider Settings - Sample Panel

On this panel you can specify whether you want WLM to manage the guest platform management provider on each system of the sysplex. If activation is selected, WLM automatically starts each guest platform management provider and restarts it as needed. Otherwise, you must manually start and stop the guest platform management provider. By default, activation is not enabled.

In addition, you can specify the systems in the sysplex on which the guest platform management provider should not be started automatically.

Coordinating Updates to a Service Definition

Since you can keep your service definition in either the WLM couple data set or in MVS data sets, it is possible to have multiple, uncoordinated updates. You should decide on a process for updating your service definition.

If you decide to work from the WLM couple data set, installing and extracting the service definition when you need to edit it, you should keep a copy in an MVS data set. The WLM application allows only one user to update an MVS data set at a time. If you do try to edit an MVS data set that is in use, you are locked out, and get a message saying the data set is in use.

If you select the Install option, and someone has done an install since you’ve extracted, the application displays a pop-up panel asking for confirmation. This way you can prevent inadvertent overwrites of the service definition. Figure 85 on page 220 shows the Overwrite warning panel.
Using the WLM Couple Data Set

The service definition installed on the WLM couple data set is the *installed* service definition. The policy that you activate must be in the installed service definition.

Of course, before you can install your service definition on the WLM couple data set, you must first allocate one, and define it to the sysplex. You can allocate a WLM couple data set through a function in the WLM application, or you can use a utility. For information on how to allocate using the utility, see "Allocate a WLM Couple Data Set" on page 148.

Allocating a WLM Couple Data Set

For availability purposes, you should allocate both a primary and an alternate WLM couple data set.

You can allocate a new primary and alternate WLM couple data set from the application using the `Allocate couple data set using CDS values` function on the Utilities menu bar option of the Definition Menu. Figure 86 on page 221 shows the allocate panel that uses CDS values.
This function primes the size parameters with values from the current couple data set. This way, you can ensure that the new couple data set is at least as large as the current one. If the new one is smaller than the current one, you will not be able to use SETXCF to make this data set available to the sysplex. (You will receive a series of messages starting with IXC255I UNABLE TO USE DATA SET.) You may need to increase one or more of these primed values if you have added new objects to your service definition since the last time you installed it.

If there is no current WLM couple data set, use the Allocate couple data set function on the Utilities menu. This function fills in the size parameters based on the service definition you are about to install. It gets the values from the ISPF PDS containing the service definition. Figure 87 shows the allocate panel that uses service definition values.

![Allocate Couple Data Set Using CDS Values](image1)

**Figure 86. Allocate Couple Data Set Using CDS Values**

![Allocate Couple Data Set Using Service Definition Values](image2)

**Figure 87. Allocate Couple Data Set Using Service Definition Values**
If you have defined your service definition, the application fills in the number of policies, workloads, service classes, application environments, and scheduling environments in that definition. These actual values are the best estimate of what this service definition requires when installed on the couple data set. If you define too large a number, it can use up a lot of space.

If you are replacing couple data sets already in use, the new data sets must be at least as large as the existing ones. You need to run a special sizing utility to determine their size. Use this method to ensure that the new couple data set is at least as large as the current one, and, in most cases, can accommodate any new objects you might have added to the service definition:

1. First use the Allocate couple data set option as a quick way of displaying the values required for the service definition you have just created or modified. Write down these values and exit the panel without allocating a couple data set.
2. Then use the Allocate couple dataset using CDS values function to obtain a panel primed with the current couple data set values.
3. Take the maximum of the values from steps 1 and 2 and enter these values on the panel. Complete the couple data set allocation from this panel.

Once you have allocated the WLM couple data sets, you need to make them available for use in the sysplex. If you are making couple data sets available for the first time, see “Make a WLM Couple Data Set Available to the Sysplex for the First Time” on page 151. If you are replacing couple data sets already in use, see “Make a Newly Formatted Couple Data Set Available to the Sysplex” on page 152.

**Installing and Extracting a Service Definition**

You can install and extract the service definition using the install and extract functions on the Utilities menu bar option of the Definition Menu.

Installing the service definition overwrites any service definition previously installed on the WLM couple data set.

When you extract the service definition, a copy remains on the WLM couple data set until the next install.

**Using MVS data sets**

If you have multiple people updating the service definition, or if you want to work with more than one service definition, you can work in multiple MVS data sets. For example, if you want to work on the service definition for next year's SLA, you can keep it in a MVS data set.

If you specify Read saved definition, you can choose from a list of data sets already created. The list is made up of all data sets your userid has worked with (the list is built based on use).

**Restricting Access to Your Service Definition**

To restrict access to the service definition stored in both the MVS data set and the WLM couple data set, you can use the same resource control facility as you do for an MVS data set. For information about how to restrict access to the WLM application functions, see “Restricting Access to the WLM Service Definition” on page 144.
Activating a Service Policy

You can activate a service policy from the ISPF application, as well as from the operator console. To activate a policy from the operator console, you issue the VARY WLM command. For more information about issuing the VARY command, see "Migration Activities" on page 144. To activate a policy from the application, choose the utilities option from the menu bar on the Definition menu.

When you choose the activate option, the application extracts the service definition information from the WLM couple data set, and displays the Policy Selection List, as shown in Figure 88. From the list, you select the service policy that you want to activate.

If you have been editing your service definition and want your changes to take effect, you have to install the changed service definition, and then activate the service policy.

<table>
<thead>
<tr>
<th>Policy Selection List ROW 1 TO 3 OF 3</th>
</tr>
</thead>
</table>
The following is the current Service Definition installed on the WLM couple data set.

Name ...: SLA1993
Installed by : USERID from system SYSTEM
Installed on : 1996/04/01 at 11:30

Select the policy to be activated with "/

--- Last Change ---
Sel Name Description User Date
--- Last Change ---
- BACKUPS During weekly backups NORTH 1996/05/11
- DAYTIME Policy from 7:00am to 5:00 pm NORTH 1996/05/11
- NIGHT Late night batch window NORTH 1996/05/11
******************************************************** Bottom of data ********************************************************

Command ==> ______________________________________________________

Figure 88. Activating a Service Policy

Printing in the Application

The print function on the selection lists in the WLM application prints the information into an ISPF list data set. You can process the ISPF list data set just as you would any other ISPF list data set. For more information about using list data sets, see the Interactive System Productivity Facility (ISPF) User’s Guide.

The "Print as GML" function under the File action bar choice of the Definition menu prints the complete service definition to a data set in GML format. The data set can be formatted using SCRIPT/VS. For more information, see the DCF SCRIPT/VS User’s Guide.

Note: To use SCRIPT/VS, you must have Document Composition Facility (DCF) Version 3, or higher, installed.
Browsing Definitions

The browse function is available on selection lists in the application, and is similar to the print function. Browse prints information into a temporary data set, and displays it.

**Figure 89** shows the browse on the BAT_T service class from the service class selection list.

```plaintext
Browse Line 00000000 Col 001 072
***************************************************************************** TOP OF DATA ****************************************************************************
* Service Class BAT_T - Batch class T
Created by user BERKEL on 1996/10/07 at 11:29:05
Base last updated by user BERKEL on 1996/10/07 at 11:29:05
Base goal:

# Description
--- --------------------------
1 5 80% complete within 00:30:00.000

***************************************************************************** BOTTOM OF DATA ****************************************************************************
```

**Figure 89. Browse Function from the Service Class Selection List**

Using XREF Function to View Service Definition Relationships

To view the relationship between any two definitions (service class, workload resource group etc), you can use the XREF function. From any Modify panel in the application, you can use the XREF function to determine an definition's relationship to another.

For example, on the Modify Service Class panel, you may want to know whether a service policy overrides the defined goal. You use the XREF function to check whether its goals are overridden in a service policy. From the Modify Service Class panel, you can also check which subsystem types reference that service class in the classification rules. If there are some that do reference it, the application displays a pop-up selection list where you can browse the Rules referencing that service class. **Figure 90** shows an example of the pop-up displayed for an Xref by Subsystem for a service class.

```plaintext
Service Class Reference ROW 1 TO 1 OF 1
These subsystem types refer to the service class.
Action Codes: 4=Browse

<table>
<thead>
<tr>
<th>Action</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>STC</td>
<td>IBM-defined subsystem type</td>
</tr>
</tbody>
</table>

***************************************************************************** BOTTOM OF DATA ****************************************************************************
```

**Figure 90. Service Class Subsystem Xref**
WLM Application Messages

IWMAM040 Unexpected error, RC=xx, RSN=xxxxxxxx.

Explanation: The application has detected an unexpected error.
In the message text:
xx The return code.
xxxxxxxx The hexadecimal reason code.
Source: Workload manager (WLM)
System action: The requested operation is not performed.
Service Administrator Response: Increase the region size and repeat the operation. If this does not help, search problem reporting data bases for a fix for the problem. If no fix exists, contact the IBM Support Center. Provide the text of this message.

IWMAM041 WLM couple data set is unavailable.

Explanation: Workload manager (WLM) was unable to find the WLM couple data set. It is possible that it does not exist. Or it may exist, but it has not been defined to the sysplex.
Source: Workload manager (WLM)
System action: The requested operation is not performed.
Service Administrator Response: Either:
- Make the couple data set available to the sysplex by updating the DATA keyword of the COUPLExx parmlib member or issuing the SETXCF command.
- Allocate the WLM couple data set.

IWMAM042 Extract failed, no service definition was found on the WLM couple data set.

Explanation: Workload manager (WLM) was unable to find a service definition on the WLM couple data set.
Source: Workload manager (WLM)
System action: The extract is not performed.
Service Administrator Response: Search problem reporting data bases for a fix for the problem. If no fix exists, contact the IBM Support Center. Provide the text of this message.

IWMAM043 Unable to obtain storage, RSN=xxxxxxxx.

Explanation: Workload manager (WLM) was unable to obtain enough storage to complete the operation.
In the message text:
xxxxxxxx The hexadecimal reason code.
Source: Workload manager (WLM)
System action: The requested operation is not performed.
Service Administrator Response: Increase the region size and repeat the operation. If this does not help, search problem reporting data bases for a fix for the problem. If no fix exists, contact the IBM Support Center. Provide the text of this message.

IWMAM044 Install failed, service definition is not valid. Validation reason code: xxxx, Validation offset: yyyy

Explanation: The service definition you are trying to install is not valid.
xxxx is the validation reason code. yyyy is the validation offset (a hex offset into the iwmszzz data structures where the problem exists).
Source: Workload manager (WLM)
System action: The install is not performed.
Service Administrator Response: For further explanation of this error, see the topic “Application Validation Reason Codes” in z/OS MVS Programming: Workload Management Services. Search problem reporting data bases for a fix for the problem. If no fix exists, contact the IBM Support Center. Provide the text of this message.

IWMAM046 Errors were found during validation of the service definition. The install has failed.

Explanation: The errors listed on the previous panel (shown when you attempted to install) have prevented the installation of the service definition.
Source: Workload manager (WLM)
System action: The install is not performed.
Service Administrator Response: Correct the errors and retry. To capture a list of the errors, go back to the error panel by either attempting to install the service definition again or by validating the service definition (use the “Validate definition” utility as described in "Using the Menu Bar on the Definition Menu” on page 191). From that panel, you can capture a list of the errors by selecting “Save listing” on the “File” menu bar option.

IWMAM050 Exceeded the maximum number of attempts to read the service definition.

Explanation: A failure in reading the service definition
from the WLM couple data set has occurred repeatedly.

**Source:** Workload manager (WLM)

**System action:** The requested operation is not performed.

**Service Administrator Response:** Search problem reporting data bases for a fix for the problem. If no fix exists, contact the IBM Support Center. Provide the text of this message.

---

**IWMAM051 Access was denied to the WLM couple data set.**

**Explanation:** The user does not have appropriate RACF authority to the WLM couple data set.

**Source:** Workload manager (WLM)

**System action:** The requested operation is not performed.

**Service Administrator Response:** Verify that the user should be authorized and have the RACF administrator give the user appropriate access. For information about how to restrict access to the couple data set, see “Restricting Access to the WLM Service Definition” on page 144.

---

**IWMAM052 The service definition functionality level (LEVELxxx) is not compatible with the WLM ISPF application level (LEVELyyy). To extract a service definition or to activate a policy, the WLM ISPF application and the MVS system must be at the same level as the service definition.**

**Explanation:** The service definition in the WLM couple data set uses functions that are not compatible with this level of the WLM ISPF application.

**Source:** Workload manager (WLM)

**System action:** The requested operation is not performed.

**Service Administrator Response:** Use the WLM ISPF application on a system that is compatible with the functionality level of the service definition.

---

**IWMAM054 Failure in ISPF: ISPF error information.**

**Explanation:** An error occurred using the application.

**Source:** Workload manager (WLM)

**System action:** The requested operation is not performed.

**Service Administrator Response:** The ISPF error information may provide a clue as to how to overcome the problem. If not, search problem reporting data bases for a fix for the problem. If no fix exists, contact the IBM Support Center. Provide the text of this message.

---

**IWMAM055 Extract failed, service definition is not valid. Validation reason code: xxxx, Validation offset: yyyy**

**Explanation:** The service definition on the WLM couple data set is not valid. The data set may be corrupted.

***xxxx*** is the validation reason code. ***yyyy*** is the validation offset.

**Source:** Workload manager (WLM)

**System action:** The extract is not performed.

**Service Administrator Response:** If the data set is corrupted, try restoring it from backups.

---

**IWMAM058 Install failed. WLM couple data set has not been reallocated for use with this OS/390 Release. Reallocate the WLM couple data set. Refer to the Migration Chapter in “MVS Planning: Workload Management.”**

**Explanation:** The WLM couple data set upon which you are attempting to install a service definition is not formatted for your service definition.

**Source:** Workload manager (WLM)

**System action:** The system does not install the service definition.

**Service Administrator Response:** Re-allocate the WLM couple data set for the service definition.

---

**IWMAM072 The service definition was not read due to a mismatch between the service definition PDS (LEVELxxx) and the WLM ISPF application (LEVELyyy). To read the service definition PDS, restart the WLM ISPF application at level LEVELxxx or higher.**

**Explanation:** The service definition in the PDS uses functions that are not compatible with this level of the WLM ISPF application.

**Source:** Workload manager (WLM)

**System action:** The requested operation is not performed.

**Service Administrator Response:** Use the WLM ISPF application on a system that is compatible with the functionality level of the service definition.
IWMAM077  Unable to use datasetname for service definition data, member membername has an unrecognized format.

Explanation:  WLM cannot use the datasetname definition data as a WLM service definition. WLM has determined that member membername in the PDS contains information that is not recognized by WLM. The incorrect member contains key and name information that does not match what WLM expects. Note that the member shown is the first member that was found to be invalid. It is possible that others are invalid, or that the entire data set is corrupted.

Source:  Workload manager (WLM)

System action:  The requested operation is not performed.

Service Administrator Response:  If you are using a downlevel version of the WLM ISPF, it may no longer be possible to open an ISPF service definition data set written by a higher level WLM Administrative Application. (See "Migrating to a New z/OS Release With an Existing Service Definition" on page 143.) Otherwise, the data set may be corrupted. If you cannot see an obvious problem in the data set, contact the IBM Support Center.

IWMAM098  The service definition was not read due to a mismatch between the service definition XML (LEVELxxx) and the WLM ISPF application (LEVELyyy). To read the service definition XML restart the WLM ISPF application at LEVELxxx or higher.

Explanation:  The service definition in the data set uses functions that are not compatible with this level of the WLM ISPF application.

Source:  Workload manager (WLM)

System action:  The requested operation is not performed.

Service Administrator Response:  Use the WLM ISPF application on a system that is compatible with the functionality level of the service definition.

IWMAM313  No more than 2047 report classes may be defined.

Explanation:  It is not possible to define more than 2047 report classes.

Source:  Workload manager (WLM)

System action:  The requested operation is not performed.

Service Administrator Response:  Do not use more than 2047 report classes.
Chapter 22. Using the z/OS Management Facility (z/OSMF) to Administer WLM

This information briefly describes the Workload Management task in z/OS Management Facility (z/OSMF).

Overview

The Workload Management task in z/OS Management Facility (z/OSMF) provides a browser-based user interface that you can use to manage z/OS Workload Manager (WLM) service definitions and provide guidelines for WLM to use when allocating resources. Specifically, you can define, modify, view, copy, import, export, and print WLM service definitions. You can also install a service definition into the WLM couple data set for the sysplex, activate a service policy, and view the status of WLM on each system in the sysplex.

Key Functions of the Workload Management Task in z/OSMF

The following information describes some of the key functions available in the Workload Management task in z/OSMF:

Display list of service definitions.

The Workload Management task provides a list of the WLM service definitions that have been defined in z/OSMF along with history information (for example, when the service definition was installed or modified), messages, and user activity. The list of service definitions is retrieved from the service definition repository, which refers to the directory in the z/OSMF data file system in which the data for the Workload Management task is stored.

Work with multiple service definitions.

In the Workload Management task, you can work with multiple service definitions simultaneously. To do so, open the service definitions with which you want to work in its own View, Modify, Copy, or Print Preview tab. You can also define multiple service definitions at the same time by opening several New tabs.

Install service definitions.

The Workload Management task provides features that you can use to install a service definition into the WLM couple data set for the z/OSMF host sysplex.

Extract the installed service definition.

The Workload Management task automatically extracts the service definition that is installed in the WLM couple data set for the z/OSMF host sysplex and stores it in the service definition repository so that you can view it, modify it, or activate one of its service policies.

Import and export service definitions.

The Workload Management task provides features that you can use to import a service definition from or export a service definition to your local workstation or a sequential data set on the z/OSMF host system. The exported service definition is formatted so that it can be opened with the z/OS WLM Administrative Application (also called the WLM ISPF application).
Provide table view and print preview of the service definition.

The Workload Management task provides two views of a service definition:

- **Table View.** The table view displays the parts of the service definition as tables. You can display the table view by opening the service definition in the New, View, Modify, or Copy tab. If you open the service definition in the New, Modify, or Copy tab, you can modify the service definition. In the View tab, you cannot modify the service definition.

- **Print Preview.** The print preview presents the service definition in HTML format and allows you to select which parts of the service definition you want to preview or print. You can display the print preview by opening the service definition in the Print Preview tab.

Activate service policies.

In the Workload Management task, you can specify which policy to activate when you install a service definition or you can activate a service policy that is defined in the service definition currently installed in the WLM couple data set for the sysplex.

Preview service policies with overrides applied.

The Workload Management task allows you to preview an HTML formatted version of the service policy with overrides applied. The HTML formatted service policy contains the information that would be included in the policy if it were activated. To preview a service policy, open the policy in the Print Preview tab.

View the sysplex status.

The Workload Management task provides an HTML formatted view (WLM Status tab) of the same data that is retrieved when you issue the D WLM,SYSTEMS command on the z/OS console. Specifically, the WLM Status tab displays the status of WLM on each system in the sysplex, and lists details about the installed service definition and the active service policy.

Define settings.

The Workload Management task provides a shared location (Settings tab) where you can specify for how long to keep the service definition history and define the code page, time zone, and backup sequential data set for the sysplex. You can also enable consistency checking between z/OSMF and the WLM couple data set, and indicate whether you want the Workload Management task to display or suppress information messages.

Actions that require the Workload Management task to interact with the sysplex are limited to the sysplex in which the z/OSMF host system is a member. Such actions include installing a service definition, activating a service policy, viewing the sysplex status, and so on. If you want to interact with another sysplex, z/OSMF must be installed on a system in that sysplex and you must log into that z/OSMF instance. You can use the service definition import and export functions to copy a service definition from one z/OSMF instance to another z/OSMF instance.

To display the Workload Management task, expand the Performance category in the z/OSMF navigation area and select **Workload Management**.

![Figure 91 on page 231](image-url) shows the main panel for the Workload Management task. The Overview tab serves as the launch point for the actions that your user ID is authorized to access within the Workload Management task. To start using the
Workload Management task, select one of the actions listed in the **Overview** tab.

For further information about the configuration of the Workload Management task in z/OSMF refer to the [IBM z/OS Management Facility Configuration Guide](#).
Appendix A. Customizing the WLM ISPF Application

This information explains how to customize your WLM application to:

- Customize the WLM application libraries.
  
  If you have renamed or changed your IPCS/WLM library names, then you can use the IWMAREX1 exit to specify your library names.

- Customize the WLM application data sets.
  
  If you would like to allocate the application data sets with your storage management policies, use IWMAREX2.

- Add the WLM application as an option on your ISPF menu.
  
  If you plan to use the WLM application frequently, you can add the WLM application as an option on your ISPF application.

- Move pop-up windows.

- Customize the keylists.

Specifying the Exits

To start the WLM application, you use a TSO/E REXX exec IWMARIN0. IWMARIN0 concatenates the IPCS/WLM data sets, allocates some data sets required for a service definition, and invokes the application panels. If EXITS is specified on the EXEC statement, IWMARIN0 uses the exits in the specified data set.

Table 19 shows the return codes from IWMARIN0.

Table 19. Return Codes from IWMARIN0

<table>
<thead>
<tr>
<th>Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Not in ISPF. Application cannot be started.</td>
</tr>
<tr>
<td>8</td>
<td>Unexpected keyword (parameter) on WLM application invocation. Unexpected keyword (parameter) is: keyword. keyword represents the keyword (parameter) from the command invocation.</td>
</tr>
</tbody>
</table>
| 12          | Unexpected error occurred when calling installation exit IWMAREX1. One of the following:  
  - Installation exit IWMAREX1 must exist in data-set name for the WLM application to run.  
    data-set name represents the data set that must contain the installation exit. Check to make sure the data set contains IWMAREX1.  
  - Installation exit IWMAREX1 must exist in the current concatenation order for the WLM application to run. |
| 16          | Unexpected keyword (parameter) from WLM exit IWMAREX1. Please check coding in WLM installation exit IWMAREX1 for incorrect keyword (parameter): parameter. parameter represents the keyword (parameter) that is incorrect. |
Table 19. Return Codes from IWMARIN0 (continued)

<table>
<thead>
<tr>
<th>Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>Unexpected RC=&quot;rc&quot; from TSO ALTLIB/ISPF LIBDEF for data-set-name. The WLM application cannot be started due to ALTLIB or LIBDEF failures for data-set-name. See specific REXX messages for the names of the data sets which failed.</td>
</tr>
<tr>
<td>24</td>
<td>ALTLIB failed during attempt to find installation exit data set. WLM application can not be started. TSO ALTLIB RC=xx. xx represents return code from TSO ALTLIB service.</td>
</tr>
</tbody>
</table>

Coding the WLM Exits

The exit stubs are shipped in SYS1.SBLSCL10. If you have a need to customize the application with the exits, you should create a data set for your exits, copy IWMAREX1 or IWMAREX2 into the data set, and modify the exits with your options. When you create the data set for your exits, you should specify the same data set characteristics as SYS1.SBLSCL10.

You specify the exit data set on the TSO/E EXEC statement when you start the WLM application. Specify the fully qualified data set name containing the exit. For example, suppose you created data set IPCS.EXITS, and coded IWMAREX1. To start the WLM application with the exit, you specify:

`EX 'SYS1.SBLSCL10(IWMARIN0)' 'EXIT(IPCS.EXITS)'`

IWMARIN1

If you have previously allocated the required data sets, either in a logon procedure, or in a CLIST, you can use IWMARIN1 to start the WLM application. To use IWMARIN1, specify:

`EX 'SYS1.SBLSCL10(IWMARIN1)'`

Customizing the WLM Application Libraries - IWMAREX1

WLM provides the IWMAREX1 exit to specify the IPCS/WLM libraries names. If you have renamed or customized the following IPCS/WLM libraries, use IWMAREX1 to specify your names.

- SYS1.SBLSCL10 – Application REXX code data set
- SYS1.SBLSEKEL0 – Application skeleton data set
- SYS1.SBLSPN10 – Application panel data set
- SYS1.SBLSTBL10 – Application keylists and commands data set
- SYS1.SBLSMG0 – Application messages data set

If you have renamed them in your installation, use IWMAREX1 to set up the allocations.

IWMAREX1 is a REXX routine for specifying installation-customized data sets required for starting the WLM ISPF application.

If you have renamed or changed the WLM/IPCS data sets in your installation, use IWMAREX1 to set up the allocations.
Processing

IWMAREX1 is called from the IWMARIN0 REXX exec.

Parameters

IWMAREX1 has the following parameters:

- **REXXDS**
  - The application REXX code data set.

- **SKELDS**
  - The application skeleton data set.

- **PANELDS**
  - The application panel data set.

- **TABLEDS**
  - The application tables (keylist and commands) data set.

- **MESSAGEDS**
  - The application messages data set.

Example

---

**Example: Using IWMAREX1**

Suppose the IPCS/WLM application resides in SYS1.IPCS.SBLSCLI0, and you have renamed your IPCS/WLM libraries to:

- SYS1.IPCS.SBLSMSG0
- SYS1.IPCS.SBLSPNL0
- SYS1.IPCS.SBLSKEL0
- SYS1.IPCS.SBLSTBL0

Suppose you have created your exit in a data set called WLM.EXITS. You code IWMAREX1 in the following way:

```
/* REXX */
queue 'REXXDS(SYS1.IPCS.SBLSCLI0)'  
queue 'SKELDS(SYS1.IPCS.SBLSKEL0)'  
queue 'PANELDS(SYS1.IPCS.SBLSPNL0)'  
queue 'TABLEDS(SYS1.IPCS.SBLSTBL0)'  
queue 'MESSAGEDS(SYS1.IPCS.SBLSMSG0)'  
Exit 0
```

To start the WLM application with the exit, you specify:

```
EX 'SYS1.IPCS.SBLSCLI0(IWMARIN0)' 'EXIT(WLM.EXITS)'
```

---

Customizing the WLM Application Data Sets - IWMAREX2

WLM provides the IWMAREX2 exit to specify:

- The application recovery data set and data set characteristics.
- The service definition data set and data set characteristics.

You should use this exit to allocate data sets according to your installation’s storage management policies. You cannot change any of the following TSO/E ALLOCATE parameters in IWMAREX2:

- Data set name (DA)
- Record format (RECFM)
• Logical record length (LRECL)
• Data set organization (DSORG)

Unless you code otherwise in this exit, the WLM application uses the following data set information as the defaults:

UNIT(SYSDA)
TRACKS SPACE(15,15)

**Processing**

This exit is called from the IWMARZAL REXX exec.

**Parameters**

IWMAREX2 has the following parameters:

**ARDSDIR**

Specifies the number of 256 byte records to be allocated for the application recovery data set.

**ARDSOPTS**

Specifies the options for the application recovery data sets, with the following sub-parameters:

**UNIT(xxxx)**

Specifies the unit type for the TSO ALLOCATE command.

**STORCLAS**

The SMS storage class.

**MGMTCLAS**

The SMS management class.

**DATACLAS**

The SMS data class.

To determine what to specify for unit, check which UNIT type is coded for the TSO ALLOCATE command in your installation.

**ARDSSPACE**

Specifies the options to allocate a data set. The options are:

**SPACE(quantity,(increment))**

**TRACKS**

**CYLINDERS**

**BLOCKS(value)**

**PDDSDIR**

Specifies the number of 256 byte records to be allocated for the print service definition data set.

**PDDSOPTS**

Specifies the options for the print service definition data set, with the following sub-parameters:

**UNIT(xxxx)**

Specifies the unit type for the TSO ALLOCATE command.

**STORCLAS**

The SMS storage class.

**MGMTCLAS**

The SMS management class.

**DATACLAS**

The SMS data class.
To determine what to specify for unit, check which UNIT type is coded for the TSO ALLOCATE command in your installation.

**PDDSSPACE**

Specifies the options to allocate a print definition data set. The options are:

- \( \text{SPACE(quantity, (increment))} \)
- \( \text{TRACKS} \)
- \( \text{CYLINDERS} \)
- \( \text{BLOCKS(value)} \)

**SDDSDIR**

Specifies the number of 256 byte records to be allocated for the service definition data set.

**SDDSOPTS**

Specifies the options for the service definition data sets, with the following sub-parameters:

- \( \text{UNIT(xxxx)} \)
  
  Specifies the unit type for the TSO ALLOCATE command.
- \( \text{STORCLAS} \)
  
  The SMS storage class.
- \( \text{MGMTCLAS} \)
  
  The SMS management class.
- \( \text{DATACLAS} \)
  
  The SMS data class.

**SDDSSPACE**

Specifies the options to allocate a data set. The options are:

- \( \text{SPACE(quantity, (increment))} \)
- \( \text{TRACKS} \)
- \( \text{CYLINDERS} \)
- \( \text{BLOCKS(value)} \)

**XMLDSOPTS**

Specifies the options for the service definition XML data sets, with the following sub-parameters:

- \( \text{UNIT(xxxx)} \)
  
  Specifies the unit type for the TSO ALLOCATE command.
- \( \text{STORCLAS} \)
  
  The SMS storage class.
- \( \text{MGMTCLAS} \)
  
  The SMS management class.
- \( \text{DATACLAS} \)
  
  The SMS data class.

**XMLDSSPACE**

Specifies the options to allocate a service definition XML data set. The options are:

- \( \text{SPACE(quantity, (increment))} \)
- \( \text{TRACKS} \)
- \( \text{CYLINDERS} \)
- \( \text{BLOCKS(value)} \)
Examples

Examples: Using IWMAREX2

- Suppose you want to specify that the UNIT type for your TSO allocate commands in your installation is type SYSDS. In exit IWMAREX2, you specify the following:
  ```
  /* REXX */
  'ARDSOPTS(UNIT(SYSDS))'
  Exit 0
  ```

- Suppose you want to specify the service definition data sets as SMS managed data sets in the standard storage class, and in the NOMIG management class. In exit IWMAREX2, you specify the following:
  ```
  /* REXX */
  queue 'ARDSOPTS(STORCLAS(STANDARD) MGMTCLAS(NOMIG))'
  queue 'AROSSPACE(SPACE(10,10) TRACKS)'
  Exit 0
  ```

Adding WLM as an ISPF Menu Option

To add the WLM application as an option on your ISPF primary panel, you should make a copy of the ISPF primary option menu - ISR@PRIM. You need to add some information to the processing section of the panel. What you add depends on whether you use IWMARIN0 or IWMARIN1 to start the application.

If you use IWMARIN0, specify:

```
WLM,'CMD(EXEC "SYS1.SBLSCLI0(IWMARIN0)"") NEWAPPL(IWMP) PASSLIB'
```

If you use the WLM exits together with IWMARIN0, specify:

```
WLM,'CMD(EXEC "SYS1.SBLSCLI0(IWMARIN0)" "EXIT(ISPF.EXITS)"") NEWAPPL(IWMP) PASSLIB'
```

If you use IWMARIN1, specify:

```
WLM,'CMD(EXEC "SYS1.SBLSCLI0(IWMARIN1)"") NEWAPPL(IWMP) PASSLIB'
```

Make sure you concatenate the library containing your customized primary panel before any others in your logon procedure or CLIST.

The following example shows the two lines added to a copy of the ISPF primary panel. In the example, IWMARIN1 is specified. The two lines are highlighted in the figure.
Moving Pop-up Windows

If you would like to customize the placement of the pop-up windows in the WLM application, you can use a manual ISPF function. You must place the cursor anywhere on the active window frame and press ENTER. ISPF acknowledges the window move request by displaying WINDOW MOVE pending message. Then place the cursor where you want the upper left corner of the pop-up placed. press Enter a second time, and the pop-up is moved to the new location.

Note: The placement lasts only for the duration of the session. If you exit the application, your changes are lost.

Example: Adding WLM as an Option on your ISPF Menu

```
Example: Adding WLM as an Option on your ISPF Menu

------------------------ ISPF/PDF PRIMARY OPTION MENU ----------------
%OPTION ===>_ZCMD
  % 0 +ISPF PARMS - Specify terminal and user parameters +TIME -
  % 1 +BROWSE - Display source data or output listings +TERMINAL -
  % 2 +EDIT - Create or change source data +PF KEYS -
  % 3 +UTILITIES - Perform utility functions
  % 4 +FOREGROUND - Invoke language processors in foreground
  % 5 +BATCH - Submit job for language processing
  % 6 +COMMAND - Enter TSO/E command or CLIST
  % 7 +DIALOG TEST - Perform dialog testing
  % 8 +LM UTILITIES - Perform library management utility functions
  % C +CHANGES - Display summary of changes for this release
  % W +WLM - WLM administrative application
  % T +TUTORIAL - Display information about ISPF/PDF
  % X +EXIT - Terminate ISPF using log and list defaults
%
+Enter%END+command to terminate ISPF.
%
)INIT
  .HELP = ISR00003
  &ZPRIM = YES /* ALWAYS A PRIMARY OPTION MENU */
  &ZHTOP = ISR00003 /* TUTORIAL TABLE OF CONTENTS */
  &ZHINDEX = ISR91000 /* TUTORIAL INDEX - 1ST PAGE */
  VPUT (ZHTOP,ZHINDEX) PROFILE
)PROC
  &ZSEL = TRANS( TRUNC (&ZCMD, '.')
    0,'PANEL(ISPOPTA)'
    1,'PGM(ISRBRO) PARM(ISRBRO01)'
    2,'PGM(ISREDIT) PARM(P,ISREDM01)'
    3,'PANEL(ISRUTIL)'
    4,'PANEL(ISRFPA)'
    5,'PGM(ISRJBI) PARM(ISRJPA) NOCHECK'
    6,'PGM(ISRPTC)'
    7,'PGM(ISRYXDR) NOCHECK'
    8,'PANEL(ISRLPRIM)'
    C,'PGM(ISPTUTOR) PARM(ISR00005)'
    W,'CMD EXEC "SYS1.SBLSCLI0(IWMARIN1)" NEWAPPL(IWMP) PASSLIB'
    T,'PGM(ISPTUTOR) PARM(ISR00000)'
    X,'EXIT'
    *,'?'
  )
  &ZTRAIL = .TRAIL
)END
```
There are some other options for moving pop-up windows. For more information about them, see ISPF Dialog Management Guide and Reference.

**Customizing the Keylists**

The WLM application uses a set of keylists which you can customize to your purposes. To edit a keylist, type:

```
KEYLIST
```

on the command line from any panel in the WLM application. The application displays the keylist utility panel, as shown in Figure 92. From this panel, select the keylist you would like to work with, and choose an action from OPTIONS on the menu bar.

```
Options Change Keylists
------------------------------------------------------------
ISPKLUP Keylist Utility for WMPI MP ROW 1 TO 9 OF 9
Command ==> ________________________________________________
Enter keylist name ____ OR
Select one keylist name from the list below:
Select Keylist T -
- KEYSBRP S
- KEYSWRK S
- KEYS001 S
- KEYS002 S
- KEYS01H S
- KEYS01P S
- KEYS01S S *** CURRENTLY ACTIVE KEYLIST ***
- KEYS02A S
- KEYS02B S
- KEYS02P S

******************************************************
Figure 92. Keylist Utility Panel
```

The keylists and the type of panels on which they are used are:

<table>
<thead>
<tr>
<th>Keylist</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>KEYS001</td>
<td>Non-scrollable, non pop-up</td>
</tr>
<tr>
<td>KEYS002</td>
<td>Scrollable, non pop-up</td>
</tr>
<tr>
<td>KEYS01P</td>
<td>Non-scrollable, pop-up</td>
</tr>
<tr>
<td>KEYS02P</td>
<td>Scrollable, pop-up</td>
</tr>
<tr>
<td>KEYS01H</td>
<td>All help panels</td>
</tr>
<tr>
<td>KEYS01S</td>
<td>Non-scrollable, non pop-up, no PF4=RETURN</td>
</tr>
<tr>
<td>KEYS02A</td>
<td>Scrollable, non pop-up, PF1=HELPD, PF10=ACTIONS.</td>
</tr>
<tr>
<td>KEYS02B</td>
<td>Scrollable, non pop-up, PF1=HELPD, PF10=LEFT, PF11=RIGHT.</td>
</tr>
<tr>
<td>KEYSBRP</td>
<td>Browse panel</td>
</tr>
<tr>
<td>KEYSWRK</td>
<td>&quot;Working...&quot; panel</td>
</tr>
</tbody>
</table>
Appendix B. CPU Capacity Table

The following table shows the unweighted CPU service units per second by CPU model. You use this information to define your minimum and maximum capacity for a resource group.

For the latest information about the processor version codes and SRM constants, see the online documentation at:


**Table 20. IBM zEnterprise 196 (z196) processor models**

<table>
<thead>
<tr>
<th>z196 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds task or SRB execution time per service unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z196, Model 401</td>
<td>12648.2213</td>
<td>0.000079</td>
</tr>
<tr>
<td>z196, Model 402</td>
<td>12075.4717</td>
<td>0.000083</td>
</tr>
<tr>
<td>z196, Model 403</td>
<td>11782.0324</td>
<td>0.000085</td>
</tr>
<tr>
<td>z196, Model 404</td>
<td>11552.3466</td>
<td>0.000087</td>
</tr>
<tr>
<td>z196, Model 405</td>
<td>11371.7129</td>
<td>0.000088</td>
</tr>
<tr>
<td>z196, Model 406</td>
<td>11220.1964</td>
<td>0.000089</td>
</tr>
<tr>
<td>z196, Model 407</td>
<td>11080.3324</td>
<td>0.000090</td>
</tr>
<tr>
<td>z196, Model 408</td>
<td>10951.4031</td>
<td>0.000091</td>
</tr>
<tr>
<td>z196, Model 409</td>
<td>10832.7691</td>
<td>0.000092</td>
</tr>
<tr>
<td>z196, Model 410</td>
<td>10716.6778</td>
<td>0.000093</td>
</tr>
<tr>
<td>z196, Model 411</td>
<td>10603.0484</td>
<td>0.000094</td>
</tr>
<tr>
<td>z196, Model 412</td>
<td>10491.8033</td>
<td>0.000095</td>
</tr>
<tr>
<td>z196, Model 413</td>
<td>10389.6104</td>
<td>0.000096</td>
</tr>
<tr>
<td>z196, Model 414</td>
<td>10289.3891</td>
<td>0.000097</td>
</tr>
<tr>
<td>z196, Model 415</td>
<td>10191.0828</td>
<td>0.000098</td>
</tr>
<tr>
<td>z196, Model 501</td>
<td>30888.0309</td>
<td>0.000032</td>
</tr>
<tr>
<td>z196, Model 502</td>
<td>29534.0212</td>
<td>0.000034</td>
</tr>
<tr>
<td>z196, Model 503</td>
<td>28776.9784</td>
<td>0.000035</td>
</tr>
<tr>
<td>z196, Model 504</td>
<td>28218.6949</td>
<td>0.000035</td>
</tr>
<tr>
<td>z196, Model 505</td>
<td>27729.6360</td>
<td>0.000036</td>
</tr>
<tr>
<td>z196, Model 506</td>
<td>27303.7543</td>
<td>0.000037</td>
</tr>
<tr>
<td>z196, Model 507</td>
<td>26890.7563</td>
<td>0.000037</td>
</tr>
<tr>
<td>z196, Model 508</td>
<td>26533.9967</td>
<td>0.000038</td>
</tr>
<tr>
<td>z196, Model 509</td>
<td>26186.5794</td>
<td>0.000038</td>
</tr>
<tr>
<td>z196, Model 510</td>
<td>25848.1422</td>
<td>0.000039</td>
</tr>
<tr>
<td>z196, Model 511</td>
<td>25518.3413</td>
<td>0.000039</td>
</tr>
<tr>
<td>z196, Model 512</td>
<td>25196.8504</td>
<td>0.000040</td>
</tr>
<tr>
<td>z196, Model 513</td>
<td>24883.3593</td>
<td>0.000040</td>
</tr>
<tr>
<td>z196, Model 514</td>
<td>24577.5730</td>
<td>0.000041</td>
</tr>
</tbody>
</table>
Table 20. IBM zEnterprise 196 (z196) processor models (continued)

<table>
<thead>
<tr>
<th>z196 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds task or SRB execution time per service unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z196, Model 515</td>
<td>24279.2109</td>
<td>0.000041</td>
</tr>
<tr>
<td>z196, Model 601</td>
<td>40404.0404</td>
<td>0.000025</td>
</tr>
<tr>
<td>z196, Model 602</td>
<td>38369.3046</td>
<td>0.000026</td>
</tr>
<tr>
<td>z196, Model 603</td>
<td>37296.0373</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 604</td>
<td>36529.6804</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 605</td>
<td>35874.4395</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 606</td>
<td>35320.0883</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 607</td>
<td>34782.6087</td>
<td>0.000029</td>
</tr>
<tr>
<td>z196, Model 608</td>
<td>34261.2420</td>
<td>0.000029</td>
</tr>
<tr>
<td>z196, Model 609</td>
<td>33755.2743</td>
<td>0.000030</td>
</tr>
<tr>
<td>z196, Model 610</td>
<td>33264.0333</td>
<td>0.000030</td>
</tr>
<tr>
<td>z196, Model 611</td>
<td>32786.8852</td>
<td>0.000031</td>
</tr>
<tr>
<td>z196, Model 612</td>
<td>32323.2323</td>
<td>0.000031</td>
</tr>
<tr>
<td>z196, Model 613</td>
<td>31872.5100</td>
<td>0.000031</td>
</tr>
<tr>
<td>z196, Model 614</td>
<td>31434.1847</td>
<td>0.000032</td>
</tr>
<tr>
<td>z196, Model 615</td>
<td>31007.7519</td>
<td>0.000032</td>
</tr>
<tr>
<td>z196, Model 701</td>
<td>61776.0618</td>
<td>0.000016</td>
</tr>
<tr>
<td>z196, Model 702</td>
<td>58394.1806</td>
<td>0.000017</td>
</tr>
<tr>
<td>z196, Model 703</td>
<td>56939.5018</td>
<td>0.000018</td>
</tr>
<tr>
<td>z196, Model 704</td>
<td>55749.1289</td>
<td>0.000018</td>
</tr>
<tr>
<td>z196, Model 705</td>
<td>54421.7687</td>
<td>0.000018</td>
</tr>
<tr>
<td>z196, Model 706</td>
<td>53691.2752</td>
<td>0.000019</td>
</tr>
<tr>
<td>z196, Model 707</td>
<td>52805.2805</td>
<td>0.000019</td>
</tr>
<tr>
<td>z196, Model 708</td>
<td>51948.0519</td>
<td>0.000019</td>
</tr>
<tr>
<td>z196, Model 709</td>
<td>50793.6508</td>
<td>0.000020</td>
</tr>
<tr>
<td>z196, Model 710</td>
<td>49844.2368</td>
<td>0.000020</td>
</tr>
<tr>
<td>z196, Model 711</td>
<td>48929.6636</td>
<td>0.000020</td>
</tr>
<tr>
<td>z196, Model 712</td>
<td>48048.0480</td>
<td>0.000021</td>
</tr>
<tr>
<td>z196, Model 713</td>
<td>47337.2781</td>
<td>0.000021</td>
</tr>
<tr>
<td>z196, Model 714</td>
<td>46647.2303</td>
<td>0.000021</td>
</tr>
<tr>
<td>z196, Model 715</td>
<td>45845.2722</td>
<td>0.000022</td>
</tr>
<tr>
<td>z196, Model 716</td>
<td>44943.8202</td>
<td>0.000022</td>
</tr>
<tr>
<td>z196, Model 717</td>
<td>44444.4444</td>
<td>0.000023</td>
</tr>
<tr>
<td>z196, Model 718</td>
<td>44077.1350</td>
<td>0.000023</td>
</tr>
<tr>
<td>z196, Model 719</td>
<td>43596.7302</td>
<td>0.000023</td>
</tr>
<tr>
<td>z196, Model 720</td>
<td>43360.4336</td>
<td>0.000023</td>
</tr>
<tr>
<td>z196, Model 721</td>
<td>42895.4424</td>
<td>0.000023</td>
</tr>
<tr>
<td>z196, Model 722</td>
<td>42666.6667</td>
<td>0.000023</td>
</tr>
</tbody>
</table>
Table 20. IBM zEnterprise 196 (z196) processor models (continued)

<table>
<thead>
<tr>
<th>z196 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds task or SRB execution time per service unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z196, Model 723</td>
<td>42328.0423</td>
<td>0.000024</td>
</tr>
<tr>
<td>z196, Model 724</td>
<td>42105.2632</td>
<td>0.000024</td>
</tr>
<tr>
<td>z196, Model 725</td>
<td>41775.4569</td>
<td>0.000024</td>
</tr>
<tr>
<td>z196, Model 726</td>
<td>41558.4416</td>
<td>0.000024</td>
</tr>
<tr>
<td>z196, Model 727</td>
<td>41343.6693</td>
<td>0.000024</td>
</tr>
<tr>
<td>z196, Model 728</td>
<td>40920.7161</td>
<td>0.000024</td>
</tr>
<tr>
<td>z196, Model 729</td>
<td>40712.4682</td>
<td>0.000025</td>
</tr>
<tr>
<td>z196, Model 730</td>
<td>40404.0404</td>
<td>0.000025</td>
</tr>
<tr>
<td>z196, Model 731</td>
<td>40100.2506</td>
<td>0.000025</td>
</tr>
<tr>
<td>z196, Model 732</td>
<td>39900.2494</td>
<td>0.000025</td>
</tr>
<tr>
<td>z196, Model 733</td>
<td>39603.9604</td>
<td>0.000025</td>
</tr>
<tr>
<td>z196, Model 734</td>
<td>39312.0393</td>
<td>0.000025</td>
</tr>
<tr>
<td>z196, Model 735</td>
<td>39119.8044</td>
<td>0.000026</td>
</tr>
<tr>
<td>z196, Model 736</td>
<td>38929.4404</td>
<td>0.000026</td>
</tr>
<tr>
<td>z196, Model 737</td>
<td>38740.9201</td>
<td>0.000026</td>
</tr>
<tr>
<td>z196, Model 738</td>
<td>38554.2169</td>
<td>0.000026</td>
</tr>
<tr>
<td>z196, Model 739</td>
<td>38369.3046</td>
<td>0.000026</td>
</tr>
<tr>
<td>z196, Model 740</td>
<td>38095.2381</td>
<td>0.000026</td>
</tr>
<tr>
<td>z196, Model 741</td>
<td>37914.6919</td>
<td>0.000026</td>
</tr>
<tr>
<td>z196, Model 742</td>
<td>37825.0591</td>
<td>0.000026</td>
</tr>
<tr>
<td>z196, Model 743</td>
<td>37647.0588</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 744</td>
<td>37470.7260</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 745</td>
<td>37296.0373</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 746</td>
<td>37122.9698</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 747</td>
<td>36951.5012</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 748</td>
<td>36866.3594</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 749</td>
<td>36781.6092</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 750</td>
<td>36613.2723</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 751</td>
<td>36446.4692</td>
<td>0.000027</td>
</tr>
<tr>
<td>z196, Model 752</td>
<td>36363.6364</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 753</td>
<td>36281.1791</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 754</td>
<td>36199.0950</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 755</td>
<td>36117.3815</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 756</td>
<td>36036.0360</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 757</td>
<td>35955.0562</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 758</td>
<td>35874.4395</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 759</td>
<td>35794.1834</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 760</td>
<td>35714.2857</td>
<td>0.000028</td>
</tr>
</tbody>
</table>
Table 20. IBM zEnterprise 196 (z196) processor models (continued)

<table>
<thead>
<tr>
<th>z196 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds task or SRB execution time per service unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z196, Model 761</td>
<td>35634.7439</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 762</td>
<td>35555.5556</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 763</td>
<td>35476.7184</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 764</td>
<td>35398.2301</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 765</td>
<td>35320.0883</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 766</td>
<td>35242.2907</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 767</td>
<td>35164.8352</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 768</td>
<td>35087.7193</td>
<td>0.000028</td>
</tr>
<tr>
<td>z196, Model 769</td>
<td>34934.4978</td>
<td>0.000029</td>
</tr>
<tr>
<td>z196, Model 770</td>
<td>34782.6087</td>
<td>0.000029</td>
</tr>
<tr>
<td>z196, Model 771</td>
<td>34632.0346</td>
<td>0.000029</td>
</tr>
<tr>
<td>z196, Model 772</td>
<td>34482.7586</td>
<td>0.000029</td>
</tr>
<tr>
<td>z196, Model 773</td>
<td>34334.7639</td>
<td>0.000029</td>
</tr>
<tr>
<td>z196, Model 774</td>
<td>34188.0342</td>
<td>0.000029</td>
</tr>
<tr>
<td>z196, Model 775</td>
<td>34042.5532</td>
<td>0.000029</td>
</tr>
<tr>
<td>z196, Model 776</td>
<td>33898.3051</td>
<td>0.000030</td>
</tr>
<tr>
<td>z196, Model 777</td>
<td>33648.2105</td>
<td>0.000030</td>
</tr>
<tr>
<td>z196, Model 778</td>
<td>33472.8033</td>
<td>0.000030</td>
</tr>
<tr>
<td>z196, Model 779</td>
<td>33264.0333</td>
<td>0.000030</td>
</tr>
<tr>
<td>z196, Model 780</td>
<td>33057.8512</td>
<td>0.000030</td>
</tr>
</tbody>
</table>

Table 21. IBM System z10 Enterprise Class (z10 EC) processor models

<table>
<thead>
<tr>
<th>System z10 EC processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z10 EC, Model 401</td>
<td>11291.4608</td>
<td>0.000089</td>
</tr>
<tr>
<td>z10 EC, Model 402</td>
<td>10680.9079</td>
<td>0.000094</td>
</tr>
<tr>
<td>z10 EC, Model 403</td>
<td>10315.9252</td>
<td>0.000097</td>
</tr>
<tr>
<td>z10 EC, Model 404</td>
<td>10050.2513</td>
<td>0.000100</td>
</tr>
<tr>
<td>z10 EC, Model 405</td>
<td>9846.1538</td>
<td>0.000102</td>
</tr>
<tr>
<td>z10 EC, Model 406</td>
<td>9673.5187</td>
<td>0.000103</td>
</tr>
<tr>
<td>z10 EC, Model 407</td>
<td>9518.1440</td>
<td>0.000105</td>
</tr>
<tr>
<td>z10 EC, Model 408</td>
<td>9373.1693</td>
<td>0.000107</td>
</tr>
<tr>
<td>z10 EC, Model 409</td>
<td>9243.2120</td>
<td>0.000108</td>
</tr>
<tr>
<td>z10 EC, Model 410</td>
<td>9111.6173</td>
<td>0.000110</td>
</tr>
<tr>
<td>z10 EC, Model 411</td>
<td>8973.6399</td>
<td>0.000111</td>
</tr>
<tr>
<td>z10 EC, Model 412</td>
<td>8834.8978</td>
<td>0.000113</td>
</tr>
<tr>
<td>z10 EC, Model 501</td>
<td>24427.4809</td>
<td>0.000041</td>
</tr>
<tr>
<td>z10 EC, Model 502</td>
<td>23021.5827</td>
<td>0.000043</td>
</tr>
</tbody>
</table>
Table 21. IBM System z10 Enterprise Class (z10 EC) processor models (continued)

<table>
<thead>
<tr>
<th>System z10 EC processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z10 EC, Model 503</td>
<td>22222.2222</td>
<td>0.000045</td>
</tr>
<tr>
<td>z10 EC, Model 504</td>
<td>21621.6216</td>
<td>0.000046</td>
</tr>
<tr>
<td>z10 EC, Model 505</td>
<td>21136.0634</td>
<td>0.000047</td>
</tr>
<tr>
<td>z10 EC, Model 506</td>
<td>20725.3886</td>
<td>0.000048</td>
</tr>
<tr>
<td>z10 EC, Model 507</td>
<td>20356.2341</td>
<td>0.000049</td>
</tr>
<tr>
<td>z10 EC, Model 508</td>
<td>20025.0313</td>
<td>0.000050</td>
</tr>
<tr>
<td>z10 EC, Model 509</td>
<td>19680.1968</td>
<td>0.000051</td>
</tr>
<tr>
<td>z10 EC, Model 510</td>
<td>19370.4600</td>
<td>0.000052</td>
</tr>
<tr>
<td>z10 EC, Model 511</td>
<td>19070.3218</td>
<td>0.000052</td>
</tr>
<tr>
<td>z10 EC, Model 512</td>
<td>18735.3630</td>
<td>0.000053</td>
</tr>
<tr>
<td>z10 EC, Model 601</td>
<td>32989.6907</td>
<td>0.000030</td>
</tr>
<tr>
<td>z10 EC, Model 602</td>
<td>31128.4047</td>
<td>0.000032</td>
</tr>
<tr>
<td>z10 EC, Model 603</td>
<td>30018.7617</td>
<td>0.000033</td>
</tr>
<tr>
<td>z10 EC, Model 604</td>
<td>29143.8980</td>
<td>0.000034</td>
</tr>
<tr>
<td>z10 EC, Model 605</td>
<td>28469.7509</td>
<td>0.000035</td>
</tr>
<tr>
<td>z10 EC, Model 606</td>
<td>27874.5645</td>
<td>0.000036</td>
</tr>
<tr>
<td>z10 EC, Model 607</td>
<td>27350.4274</td>
<td>0.000037</td>
</tr>
<tr>
<td>z10 EC, Model 608</td>
<td>26890.7563</td>
<td>0.000037</td>
</tr>
<tr>
<td>z10 EC, Model 609</td>
<td>26402.6403</td>
<td>0.000038</td>
</tr>
<tr>
<td>z10 EC, Model 610</td>
<td>25931.9287</td>
<td>0.000039</td>
</tr>
<tr>
<td>z10 EC, Model 611</td>
<td>25477.7070</td>
<td>0.000039</td>
</tr>
<tr>
<td>z10 EC, Model 612</td>
<td>25078.3699</td>
<td>0.000040</td>
</tr>
<tr>
<td>z10 EC, Model 701</td>
<td>47619.0476</td>
<td>0.000021</td>
</tr>
<tr>
<td>z10 EC, Model 702</td>
<td>44692.7374</td>
<td>0.000022</td>
</tr>
<tr>
<td>z10 EC, Model 703</td>
<td>43010.7527</td>
<td>0.000023</td>
</tr>
<tr>
<td>z10 EC, Model 704</td>
<td>41666.6667</td>
<td>0.000024</td>
</tr>
<tr>
<td>z10 EC, Model 705</td>
<td>40404.0404</td>
<td>0.000025</td>
</tr>
<tr>
<td>z10 EC, Model 706</td>
<td>39603.9604</td>
<td>0.000025</td>
</tr>
<tr>
<td>z10 EC, Model 707</td>
<td>38834.9515</td>
<td>0.000026</td>
</tr>
<tr>
<td>z10 EC, Model 708</td>
<td>38004.7506</td>
<td>0.000026</td>
</tr>
<tr>
<td>z10 EC, Model 709</td>
<td>37037.0370</td>
<td>0.000027</td>
</tr>
<tr>
<td>z10 EC, Model 710</td>
<td>36281.1791</td>
<td>0.000028</td>
</tr>
<tr>
<td>z10 EC, Model 711</td>
<td>35476.7184</td>
<td>0.000028</td>
</tr>
<tr>
<td>z10 EC, Model 712</td>
<td>34782.6087</td>
<td>0.000029</td>
</tr>
<tr>
<td>z10 EC, Model 713</td>
<td>34188.0342</td>
<td>0.000029</td>
</tr>
<tr>
<td>z10 EC, Model 714</td>
<td>33613.4454</td>
<td>0.000030</td>
</tr>
<tr>
<td>z10 EC, Model 715</td>
<td>33057.8512</td>
<td>0.000030</td>
</tr>
<tr>
<td>z10 EC, Model 716</td>
<td>32454.3611</td>
<td>0.000031</td>
</tr>
</tbody>
</table>
Table 21. IBM System z10 Enterprise Class (z10 EC) processor models (continued)

<table>
<thead>
<tr>
<th>System z10 EC processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z10 EC, Model 717</td>
<td>32064.1283</td>
<td>0.000031</td>
</tr>
<tr>
<td>z10 EC, Model 718</td>
<td>31746.0317</td>
<td>0.000032</td>
</tr>
<tr>
<td>z10 EC, Model 719</td>
<td>31434.1847</td>
<td>0.000032</td>
</tr>
<tr>
<td>z10 EC, Model 720</td>
<td>31128.4047</td>
<td>0.000032</td>
</tr>
<tr>
<td>z10 EC, Model 721</td>
<td>30769.2308</td>
<td>0.000033</td>
</tr>
<tr>
<td>z10 EC, Model 722</td>
<td>30534.3511</td>
<td>0.000033</td>
</tr>
<tr>
<td>z10 EC, Model 723</td>
<td>30303.0303</td>
<td>0.000033</td>
</tr>
<tr>
<td>z10 EC, Model 724</td>
<td>30075.1880</td>
<td>0.000033</td>
</tr>
<tr>
<td>z10 EC, Model 725</td>
<td>29850.7463</td>
<td>0.000034</td>
</tr>
<tr>
<td>z10 EC, Model 726</td>
<td>29629.6296</td>
<td>0.000034</td>
</tr>
<tr>
<td>z10 EC, Model 727</td>
<td>29411.7647</td>
<td>0.000034</td>
</tr>
<tr>
<td>z10 EC, Model 728</td>
<td>29143.8980</td>
<td>0.000034</td>
</tr>
<tr>
<td>z10 EC, Model 729</td>
<td>28933.0922</td>
<td>0.000035</td>
</tr>
<tr>
<td>z10 EC, Model 730</td>
<td>28725.3142</td>
<td>0.000035</td>
</tr>
<tr>
<td>z10 EC, Model 731</td>
<td>28520.4991</td>
<td>0.000035</td>
</tr>
<tr>
<td>z10 EC, Model 732</td>
<td>28318.5841</td>
<td>0.000035</td>
</tr>
<tr>
<td>z10 EC, Model 733</td>
<td>28119.5079</td>
<td>0.000036</td>
</tr>
<tr>
<td>z10 EC, Model 734</td>
<td>27923.2112</td>
<td>0.000036</td>
</tr>
<tr>
<td>z10 EC, Model 735</td>
<td>27777.7778</td>
<td>0.000036</td>
</tr>
<tr>
<td>z10 EC, Model 736</td>
<td>27633.8515</td>
<td>0.000036</td>
</tr>
<tr>
<td>z10 EC, Model 737</td>
<td>27491.4089</td>
<td>0.000036</td>
</tr>
<tr>
<td>z10 EC, Model 738</td>
<td>27303.7543</td>
<td>0.000037</td>
</tr>
<tr>
<td>z10 EC, Model 739</td>
<td>27118.6441</td>
<td>0.000037</td>
</tr>
<tr>
<td>z10 EC, Model 740</td>
<td>26936.0269</td>
<td>0.000037</td>
</tr>
<tr>
<td>z10 EC, Model 741</td>
<td>26755.8528</td>
<td>0.000037</td>
</tr>
<tr>
<td>z10 EC, Model 742</td>
<td>26666.6667</td>
<td>0.000038</td>
</tr>
<tr>
<td>z10 EC, Model 743</td>
<td>26533.9967</td>
<td>0.000038</td>
</tr>
<tr>
<td>z10 EC, Model 744</td>
<td>26402.6403</td>
<td>0.000038</td>
</tr>
<tr>
<td>z10 EC, Model 745</td>
<td>26272.5780</td>
<td>0.000038</td>
</tr>
<tr>
<td>z10 EC, Model 746</td>
<td>26101.1419</td>
<td>0.000038</td>
</tr>
<tr>
<td>z10 EC, Model 747</td>
<td>25974.0260</td>
<td>0.000039</td>
</tr>
<tr>
<td>z10 EC, Model 748</td>
<td>25848.1422</td>
<td>0.000039</td>
</tr>
<tr>
<td>z10 EC, Model 749</td>
<td>25764.8953</td>
<td>0.000039</td>
</tr>
<tr>
<td>z10 EC, Model 750</td>
<td>25641.0256</td>
<td>0.000039</td>
</tr>
<tr>
<td>z10 EC, Model 751</td>
<td>25477.7070</td>
<td>0.000039</td>
</tr>
<tr>
<td>z10 EC, Model 752</td>
<td>25356.5769</td>
<td>0.000039</td>
</tr>
<tr>
<td>z10 EC, Model 753</td>
<td>25236.5931</td>
<td>0.000040</td>
</tr>
<tr>
<td>z10 EC, Model 754</td>
<td>25117.7394</td>
<td>0.000040</td>
</tr>
</tbody>
</table>
Table 21. IBM System z10 Enterprise Class (z10 EC) processor models (continued)

<table>
<thead>
<tr>
<th>System z10 EC processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z10 EC, Model 755</td>
<td>25039.1236</td>
<td>0.000040</td>
</tr>
<tr>
<td>z10 EC, Model 756</td>
<td>24960.9984</td>
<td>0.000040</td>
</tr>
<tr>
<td>z10 EC, Model 757</td>
<td>24883.3593</td>
<td>0.000040</td>
</tr>
<tr>
<td>z10 EC, Model 758</td>
<td>24806.2016</td>
<td>0.000040</td>
</tr>
<tr>
<td>z10 EC, Model 759</td>
<td>24729.5209</td>
<td>0.000040</td>
</tr>
<tr>
<td>z10 EC, Model 760</td>
<td>24615.3846</td>
<td>0.000041</td>
</tr>
<tr>
<td>z10 EC, Model 761</td>
<td>24502.2971</td>
<td>0.000041</td>
</tr>
<tr>
<td>z10 EC, Model 762</td>
<td>24390.2439</td>
<td>0.000041</td>
</tr>
<tr>
<td>z10 EC, Model 763</td>
<td>24279.2109</td>
<td>0.000041</td>
</tr>
<tr>
<td>z10 EC, Model 764</td>
<td>24169.1843</td>
<td>0.000041</td>
</tr>
</tbody>
</table>

Table 22. IBM System z9 Business Class (z9 BC) processor models

<table>
<thead>
<tr>
<th>System z9 BC processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z9 BC, Model A01</td>
<td>1316.9808</td>
<td>0.000759</td>
</tr>
<tr>
<td>z9 BC, Model A02</td>
<td>1265.3223</td>
<td>0.000790</td>
</tr>
<tr>
<td>z9 BC, Model A03</td>
<td>1229.1619</td>
<td>0.000814</td>
</tr>
<tr>
<td>z9 BC, Model B01</td>
<td>1927.0143</td>
<td>0.000519</td>
</tr>
<tr>
<td>z9 BC, Model B02</td>
<td>1851.4233</td>
<td>0.000540</td>
</tr>
<tr>
<td>z9 BC, Model B03</td>
<td>1798.5612</td>
<td>0.000556</td>
</tr>
<tr>
<td>z9 BC, Model C01</td>
<td>2341.5776</td>
<td>0.000427</td>
</tr>
<tr>
<td>z9 BC, Model C02</td>
<td>2249.7188</td>
<td>0.000445</td>
</tr>
<tr>
<td>z9 BC, Model C03</td>
<td>2185.4938</td>
<td>0.000458</td>
</tr>
<tr>
<td>z9 BC, Model D01</td>
<td>3000.1875</td>
<td>0.000333</td>
</tr>
<tr>
<td>z9 BC, Model D02</td>
<td>2882.3635</td>
<td>0.000347</td>
</tr>
<tr>
<td>z9 BC, Model D03</td>
<td>2800.1400</td>
<td>0.000357</td>
</tr>
<tr>
<td>z9 BC, Model E01</td>
<td>3560.3026</td>
<td>0.000058</td>
</tr>
<tr>
<td>z9 BC, Model E02</td>
<td>3420.9964</td>
<td>0.000059</td>
</tr>
<tr>
<td>z9 BC, Model F01</td>
<td>4413.7931</td>
<td>0.000060</td>
</tr>
<tr>
<td>z9 BC, Model F02</td>
<td>4240.6573</td>
<td>0.000062</td>
</tr>
<tr>
<td>z9 BC, Model G01</td>
<td>5588.5435</td>
<td>0.000179</td>
</tr>
<tr>
<td>z9 BC, Model H01</td>
<td>6611.5702</td>
<td>0.000151</td>
</tr>
<tr>
<td>z9 BC, Model I01</td>
<td>7615.4212</td>
<td>0.000131</td>
</tr>
<tr>
<td>z9 BC, Model J01</td>
<td>8743.1694</td>
<td>0.000114</td>
</tr>
<tr>
<td>z9 BC, Model 1way L0x LPAR</td>
<td>3560.3026</td>
<td>0.000281</td>
</tr>
<tr>
<td>z9 BC, Model 2way L0x LPAR</td>
<td>3420.9964</td>
<td>0.000292</td>
</tr>
<tr>
<td>z9 BC, Model L03</td>
<td>3322.9491</td>
<td>0.000301</td>
</tr>
</tbody>
</table>
Table 22. IBM System z9 Business Class (z9 BC) processor models (continued)

<table>
<thead>
<tr>
<th>System z9 BC processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z9 BC, Model L04</td>
<td>3238.8664</td>
<td>0.000309</td>
</tr>
<tr>
<td>z9 BC, Model 1way M0x LPAR</td>
<td>4413.7931</td>
<td>0.000227</td>
</tr>
<tr>
<td>z9 BC, Model 2way M0x LPAR</td>
<td>4240.6573</td>
<td>0.000236</td>
</tr>
<tr>
<td>z9 BC, Model M03</td>
<td>4119.4645</td>
<td>0.000243</td>
</tr>
<tr>
<td>z9 BC, Model M04</td>
<td>4016.0643</td>
<td>0.000249</td>
</tr>
<tr>
<td>z9 BC, Model 1way N0x LPAR</td>
<td>5588.5435</td>
<td>0.000179</td>
</tr>
<tr>
<td>z9 BC, Model N02</td>
<td>5369.1275</td>
<td>0.000186</td>
</tr>
<tr>
<td>z9 BC, Model N03</td>
<td>5215.1239</td>
<td>0.000192</td>
</tr>
<tr>
<td>z9 BC, Model N04</td>
<td>5084.2072</td>
<td>0.000197</td>
</tr>
<tr>
<td>z9 BC, Model 1way O0x LPAR</td>
<td>6611.5702</td>
<td>0.000151</td>
</tr>
<tr>
<td>z9 BC, Model O02</td>
<td>6351.7269</td>
<td>0.000157</td>
</tr>
<tr>
<td>z9 BC, Model O03</td>
<td>6170.4589</td>
<td>0.000162</td>
</tr>
<tr>
<td>z9 BC, Model O04</td>
<td>6015.0376</td>
<td>0.000166</td>
</tr>
<tr>
<td>z9 BC, Model 1way P0x LPAR</td>
<td>7615.4212</td>
<td>0.000131</td>
</tr>
<tr>
<td>z9 BC, Model P02</td>
<td>7315.9579</td>
<td>0.000137</td>
</tr>
<tr>
<td>z9 BC, Model P03</td>
<td>7107.9520</td>
<td>0.000141</td>
</tr>
<tr>
<td>z9 BC, Model P04</td>
<td>6929.4067</td>
<td>0.000144</td>
</tr>
<tr>
<td>z9 BC, Model 1way Q0x LPAR</td>
<td>8743.1694</td>
<td>0.000114</td>
</tr>
<tr>
<td>z9 BC, Model Q02</td>
<td>8398.9501</td>
<td>0.000119</td>
</tr>
<tr>
<td>z9 BC, Model Q03</td>
<td>8159.1025</td>
<td>0.000123</td>
</tr>
<tr>
<td>z9 BC, Model Q04</td>
<td>7952.2863</td>
<td>0.000126</td>
</tr>
<tr>
<td>z9 BC, Model R01</td>
<td>9809.9326</td>
<td>0.000102</td>
</tr>
<tr>
<td>z9 BC, Model R02</td>
<td>9422.8504</td>
<td>0.000106</td>
</tr>
<tr>
<td>z9 BC, Model R03</td>
<td>9153.3181</td>
<td>0.000109</td>
</tr>
<tr>
<td>z9 BC, Model R04</td>
<td>8923.5917</td>
<td>0.000112</td>
</tr>
<tr>
<td>z9 BC, Model S01</td>
<td>10996.5636</td>
<td>0.000091</td>
</tr>
<tr>
<td>z9 BC, Model S02</td>
<td>10568.0317</td>
<td>0.000095</td>
</tr>
<tr>
<td>z9 BC, Model S03</td>
<td>10262.9891</td>
<td>0.000097</td>
</tr>
<tr>
<td>z9 BC, Model S04</td>
<td>10006.2539</td>
<td>0.000100</td>
</tr>
<tr>
<td>z9 BC, Model T01</td>
<td>12298.2321</td>
<td>0.000081</td>
</tr>
<tr>
<td>z9 BC, Model T02</td>
<td>11816.8390</td>
<td>0.000085</td>
</tr>
<tr>
<td>z9 BC, Model T03</td>
<td>11477.7618</td>
<td>0.000087</td>
</tr>
<tr>
<td>z9 BC, Model T04</td>
<td>11188.8112</td>
<td>0.000089</td>
</tr>
<tr>
<td>z9 BC, Model U01</td>
<td>13710.3685</td>
<td>0.000073</td>
</tr>
</tbody>
</table>
Table 22. IBM System z9 Business Class (z9 BC) processor models (continued)

<table>
<thead>
<tr>
<th>System z9 BC processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z9 BC, Model U02</td>
<td>13168.7243</td>
<td>0.000076</td>
</tr>
<tr>
<td>z9 BC, Model U03</td>
<td>12789.7682</td>
<td>0.000078</td>
</tr>
<tr>
<td>z9 BC, Model U04</td>
<td>12470.7716</td>
<td>0.000080</td>
</tr>
<tr>
<td>z9 BC, Model V01</td>
<td>15399.4225</td>
<td>0.000055</td>
</tr>
<tr>
<td>z9 BC, Model V02</td>
<td>14801.1101</td>
<td>0.000055</td>
</tr>
<tr>
<td>z9 BC, Model V03</td>
<td>14375.5615</td>
<td>0.000056</td>
</tr>
<tr>
<td>z9 BC, Model V04</td>
<td>14010.5079</td>
<td>0.000056</td>
</tr>
<tr>
<td>z9 BC, Model W01</td>
<td>17278.6177</td>
<td>0.000058</td>
</tr>
<tr>
<td>z9 BC, Model W02</td>
<td>16597.5104</td>
<td>0.000060</td>
</tr>
<tr>
<td>z9 BC, Model W03</td>
<td>16129.0323</td>
<td>0.000062</td>
</tr>
<tr>
<td>z9 BC, Model W04</td>
<td>15717.0923</td>
<td>0.000064</td>
</tr>
<tr>
<td>z9 BC, Model X01</td>
<td>19347.0735</td>
<td>0.000052</td>
</tr>
<tr>
<td>z9 BC, Model X02</td>
<td>18583.0430</td>
<td>0.000054</td>
</tr>
<tr>
<td>z9 BC, Model X03</td>
<td>18058.6907</td>
<td>0.000055</td>
</tr>
<tr>
<td>z9 BC, Model X04</td>
<td>17601.7602</td>
<td>0.000057</td>
</tr>
<tr>
<td>z9 BC, Model Y01</td>
<td>21419.094</td>
<td>0.000047</td>
</tr>
<tr>
<td>z9 BC, Model Y02</td>
<td>20592.0268</td>
<td>0.000049</td>
</tr>
<tr>
<td>z9 BC, Model Y03</td>
<td>20000.0000</td>
<td>0.000050</td>
</tr>
<tr>
<td>z9 BC, Model Y04</td>
<td>19488.4287</td>
<td>0.000051</td>
</tr>
<tr>
<td>z9 BC, Model Z01</td>
<td>24427.4809</td>
<td>0.000041</td>
</tr>
<tr>
<td>z9 BC, Model Z02</td>
<td>23460.4106</td>
<td>0.000043</td>
</tr>
<tr>
<td>z9 BC, Model Z03</td>
<td>22792.0228</td>
<td>0.000044</td>
</tr>
<tr>
<td>z9 BC, Model Z04</td>
<td>22222.2222</td>
<td>0.000045</td>
</tr>
</tbody>
</table>

Table 23. IBM System z9 Enterprise Class (z9 EC) processor models

<table>
<thead>
<tr>
<th>System z9 EC processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z9 EC, Model 401</td>
<td>10107.3910</td>
<td>0.000099</td>
</tr>
<tr>
<td>z9 EC, Model 402</td>
<td>9708.7379</td>
<td>0.000103</td>
</tr>
<tr>
<td>z9 EC, Model 403</td>
<td>9433.9623</td>
<td>0.000106</td>
</tr>
<tr>
<td>z9 EC, Model 404</td>
<td>9195.4023</td>
<td>0.000109</td>
</tr>
<tr>
<td>z9 EC, Model 405</td>
<td>8958.5666</td>
<td>0.000112</td>
</tr>
<tr>
<td>z9 EC, Model 406</td>
<td>8762.3220</td>
<td>0.000114</td>
</tr>
<tr>
<td>z9 EC, Model 407</td>
<td>8565.3105</td>
<td>0.000117</td>
</tr>
<tr>
<td>z9 EC, Model 408</td>
<td>8368.2008</td>
<td>0.000120</td>
</tr>
<tr>
<td>z9 EC, Model 501</td>
<td>19631.9018</td>
<td>0.000051</td>
</tr>
<tr>
<td>z9 EC, Model 502</td>
<td>18867.9245</td>
<td>0.000053</td>
</tr>
<tr>
<td>z9 EC, Model 503</td>
<td>18327.6060</td>
<td>0.000055</td>
</tr>
<tr>
<td>System z9 EC processor models</td>
<td>Service units per second of task or SRB execution time</td>
<td>Seconds Task or SRB Execution Time Per Service Unit</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------------------------------------------------------</td>
<td>-----------------------------------------------------</td>
</tr>
<tr>
<td>z9 EC, Model 504</td>
<td>17857.1429</td>
<td>0.000056</td>
</tr>
<tr>
<td>z9 EC, Model 505</td>
<td>17391.3043</td>
<td>0.000058</td>
</tr>
<tr>
<td>z9 EC, Model 506</td>
<td>17003.1881</td>
<td>0.000059</td>
</tr>
<tr>
<td>z9 EC, Model 507</td>
<td>16614.7456</td>
<td>0.000060</td>
</tr>
<tr>
<td>z9 EC, Model 508</td>
<td>16227.1805</td>
<td>0.000062</td>
</tr>
<tr>
<td>z9 EC, Model 601</td>
<td>22774.1456</td>
<td>0.000042</td>
</tr>
<tr>
<td>z9 EC, Model 602</td>
<td>22857.1429</td>
<td>0.000044</td>
</tr>
<tr>
<td>z9 EC, Model 603</td>
<td>22191.4008</td>
<td>0.000045</td>
</tr>
<tr>
<td>z9 EC, Model 604</td>
<td>21621.6216</td>
<td>0.000046</td>
</tr>
<tr>
<td>z9 EC, Model 605</td>
<td>21052.6316</td>
<td>0.000048</td>
</tr>
<tr>
<td>z9 EC, Model 606</td>
<td>20592.0206</td>
<td>0.000049</td>
</tr>
<tr>
<td>z9 EC, Model 607</td>
<td>20125.7862</td>
<td>0.000050</td>
</tr>
<tr>
<td>z9 EC, Model 608</td>
<td>19656.0197</td>
<td>0.000051</td>
</tr>
<tr>
<td>z9 EC, Model 701</td>
<td>29520.2952</td>
<td>0.000034</td>
</tr>
<tr>
<td>z9 EC, Model 702</td>
<td>28368.7943</td>
<td>0.000035</td>
</tr>
<tr>
<td>z9 EC, Model 703</td>
<td>27538.7263</td>
<td>0.000036</td>
</tr>
<tr>
<td>z9 EC, Model 704</td>
<td>26845.6376</td>
<td>0.000037</td>
</tr>
<tr>
<td>z9 EC, Model 705</td>
<td>26143.7908</td>
<td>0.000038</td>
</tr>
<tr>
<td>z9 EC, Model 706</td>
<td>25559.1054</td>
<td>0.000039</td>
</tr>
<tr>
<td>z9 EC, Model 707</td>
<td>25000.0000</td>
<td>0.000040</td>
</tr>
<tr>
<td>z9 EC, Model 708</td>
<td>24427.4809</td>
<td>0.000041</td>
</tr>
<tr>
<td>z9 EC, Model 709</td>
<td>23845.0075</td>
<td>0.000042</td>
</tr>
<tr>
<td>z9 EC, Model 710</td>
<td>23391.8129</td>
<td>0.000043</td>
</tr>
<tr>
<td>z9 EC, Model 711</td>
<td>22922.6361</td>
<td>0.000044</td>
</tr>
<tr>
<td>z9 EC, Model 712</td>
<td>22566.9958</td>
<td>0.000044</td>
</tr>
<tr>
<td>z9 EC, Model 713</td>
<td>22099.4475</td>
<td>0.000045</td>
</tr>
<tr>
<td>z9 EC, Model 714</td>
<td>21739.1304</td>
<td>0.000046</td>
</tr>
<tr>
<td>z9 EC, Model 715</td>
<td>21390.3743</td>
<td>0.000047</td>
</tr>
<tr>
<td>z9 EC, Model 716</td>
<td>21052.6316</td>
<td>0.000048</td>
</tr>
<tr>
<td>z9 EC, Model 717</td>
<td>20833.3333</td>
<td>0.000048</td>
</tr>
<tr>
<td>z9 EC, Model 718</td>
<td>20592.0206</td>
<td>0.000049</td>
</tr>
<tr>
<td>z9 EC, Model 719</td>
<td>20356.2341</td>
<td>0.000049</td>
</tr>
<tr>
<td>z9 EC, Model 720</td>
<td>20125.7862</td>
<td>0.000050</td>
</tr>
<tr>
<td>z9 EC, Model 721</td>
<td>19900.4975</td>
<td>0.000050</td>
</tr>
<tr>
<td>z9 EC, Model 722</td>
<td>19777.5031</td>
<td>0.000051</td>
</tr>
<tr>
<td>z9 EC, Model 723</td>
<td>19656.0197</td>
<td>0.000051</td>
</tr>
<tr>
<td>z9 EC, Model 724</td>
<td>19536.0195</td>
<td>0.000051</td>
</tr>
<tr>
<td>z9 EC, Model 725</td>
<td>19417.4757</td>
<td>0.000052</td>
</tr>
</tbody>
</table>
Table 23. IBM System z9 Enterprise Class (z9 EC) processor models (continued)

<table>
<thead>
<tr>
<th>System z9 EC processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>z9 EC, Model 726</td>
<td>19300.3619</td>
<td>0.000052</td>
</tr>
<tr>
<td>z9 EC, Model 727</td>
<td>19070.3218</td>
<td>0.000052</td>
</tr>
<tr>
<td>z9 EC, Model 728</td>
<td>18845.7008</td>
<td>0.000053</td>
</tr>
<tr>
<td>z9 EC, Model 729</td>
<td>18735.3630</td>
<td>0.000053</td>
</tr>
<tr>
<td>z9 EC, Model 730</td>
<td>18626.3097</td>
<td>0.000054</td>
</tr>
<tr>
<td>z9 EC, Model 731</td>
<td>18518.5185</td>
<td>0.000054</td>
</tr>
<tr>
<td>z9 EC, Model 732</td>
<td>18411.9678</td>
<td>0.000054</td>
</tr>
<tr>
<td>z9 EC, Model 733</td>
<td>18285.7143</td>
<td>0.000055</td>
</tr>
<tr>
<td>z9 EC, Model 734</td>
<td>18161.1805</td>
<td>0.000055</td>
</tr>
<tr>
<td>z9 EC, Model 735</td>
<td>18038.3315</td>
<td>0.000055</td>
</tr>
<tr>
<td>z9 EC, Model 736</td>
<td>17917.1333</td>
<td>0.000056</td>
</tr>
<tr>
<td>z9 EC, Model 737</td>
<td>17797.5528</td>
<td>0.000056</td>
</tr>
<tr>
<td>z9 EC, Model 738</td>
<td>17679.5580</td>
<td>0.000057</td>
</tr>
<tr>
<td>z9 EC, Model 739</td>
<td>17563.1175</td>
<td>0.000057</td>
</tr>
<tr>
<td>z9 EC, Model 740</td>
<td>17448.2007</td>
<td>0.000057</td>
</tr>
<tr>
<td>z9 EC, Model 741</td>
<td>17448.2007</td>
<td>0.000057</td>
</tr>
<tr>
<td>z9 EC, Model 742</td>
<td>17334.7779</td>
<td>0.000058</td>
</tr>
<tr>
<td>z9 EC, Model 743</td>
<td>17222.8202</td>
<td>0.000058</td>
</tr>
<tr>
<td>z9 EC, Model 744</td>
<td>17112.2995</td>
<td>0.000058</td>
</tr>
<tr>
<td>z9 EC, Model 745</td>
<td>17003.1881</td>
<td>0.000059</td>
</tr>
<tr>
<td>z9 EC, Model 746</td>
<td>16895.4593</td>
<td>0.000059</td>
</tr>
<tr>
<td>z9 EC, Model 747</td>
<td>16895.4593</td>
<td>0.000059</td>
</tr>
<tr>
<td>z9 EC, Model 748</td>
<td>16771.4885</td>
<td>0.000060</td>
</tr>
<tr>
<td>z9 EC, Model 750</td>
<td>16528.9256</td>
<td>0.000061</td>
</tr>
<tr>
<td>z9 EC, Model 751</td>
<td>16410.2564</td>
<td>0.000061</td>
</tr>
<tr>
<td>z9 EC, Model 752</td>
<td>16293.2790</td>
<td>0.000061</td>
</tr>
<tr>
<td>z9 EC, Model 753</td>
<td>16177.9575</td>
<td>0.000062</td>
</tr>
<tr>
<td>z9 EC, Model 754</td>
<td>15285.7008</td>
<td>0.000052</td>
</tr>
</tbody>
</table>

Table 24. zSeries 990 processor models

<table>
<thead>
<tr>
<th>zSeries 990 processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>zSeries 990, Model 301</td>
<td>21857.9</td>
<td>0.000046</td>
</tr>
<tr>
<td>zSeries 990, Model 302</td>
<td>20752.3</td>
<td>0.000048</td>
</tr>
<tr>
<td>zSeries 990, Model 303</td>
<td>20075.3</td>
<td>0.000050</td>
</tr>
<tr>
<td>zSeries 990, Model 304</td>
<td>19559.9</td>
<td>0.000051</td>
</tr>
<tr>
<td>zSeries 990, Model 305</td>
<td>19047.6</td>
<td>0.000053</td>
</tr>
</tbody>
</table>
### Table 24. zSeries 990 processor models (continued)

<table>
<thead>
<tr>
<th>zSeries 990 processor models</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>zSeries 990, Model 306</td>
<td>18626.3</td>
<td>0.000054</td>
</tr>
<tr>
<td>zSeries 990, Model 307</td>
<td>18202.5</td>
<td>0.000055</td>
</tr>
<tr>
<td>zSeries 990, Model 308</td>
<td>17777.8</td>
<td>0.000056</td>
</tr>
<tr>
<td>zSeries 990, Model 309</td>
<td>17353.6</td>
<td>0.000058</td>
</tr>
<tr>
<td>zSeries 990, Model 310</td>
<td>17003.2</td>
<td>0.000059</td>
</tr>
<tr>
<td>zSeries 990, Model 311</td>
<td>16666.7</td>
<td>0.000060</td>
</tr>
<tr>
<td>zSeries 990, Model 312</td>
<td>16326.5</td>
<td>0.000061</td>
</tr>
<tr>
<td>zSeries 990, Model 313</td>
<td>15984.0</td>
<td>0.000063</td>
</tr>
<tr>
<td>zSeries 990, Model 314</td>
<td>15640.3</td>
<td>0.000064</td>
</tr>
<tr>
<td>zSeries 990, Model 315</td>
<td>15296.4</td>
<td>0.000065</td>
</tr>
<tr>
<td>zSeries 990, Model 316</td>
<td>14953.3</td>
<td>0.000067</td>
</tr>
<tr>
<td>zSeries 990, Model 317</td>
<td>14787.4</td>
<td>0.000068</td>
</tr>
<tr>
<td>zSeries 990, Model 318</td>
<td>14611.9</td>
<td>0.000068</td>
</tr>
<tr>
<td>zSeries 990, Model 319</td>
<td>14532.2</td>
<td>0.000069</td>
</tr>
<tr>
<td>zSeries 990, Model 320</td>
<td>14440.4</td>
<td>0.000069</td>
</tr>
<tr>
<td>zSeries 990, Model 321</td>
<td>14349.8</td>
<td>0.000070</td>
</tr>
<tr>
<td>zSeries 990, Model 322</td>
<td>14260.2</td>
<td>0.000070</td>
</tr>
<tr>
<td>zSeries 990, Model 323</td>
<td>14171.8</td>
<td>0.000071</td>
</tr>
<tr>
<td>zSeries 990, Model 324</td>
<td>14084.5</td>
<td>0.000071</td>
</tr>
<tr>
<td>zSeries 990, Model 325</td>
<td>13998.3</td>
<td>0.000071</td>
</tr>
<tr>
<td>zSeries 990, Model 326</td>
<td>13913.0</td>
<td>0.000072</td>
</tr>
<tr>
<td>zSeries 990, Model 327</td>
<td>13828.9</td>
<td>0.000072</td>
</tr>
<tr>
<td>zSeries 990, Model 328</td>
<td>13745.7</td>
<td>0.000073</td>
</tr>
<tr>
<td>zSeries 990, Model 329</td>
<td>13663.5</td>
<td>0.000073</td>
</tr>
<tr>
<td>zSeries 990, Model 330</td>
<td>13582.3</td>
<td>0.000074</td>
</tr>
<tr>
<td>zSeries 990, Model 331</td>
<td>13490.7</td>
<td>0.000074</td>
</tr>
<tr>
<td>zSeries 990, Model 332</td>
<td>13400.3</td>
<td>0.000075</td>
</tr>
</tbody>
</table>

### Table 25. zSeries 900 processor models

<table>
<thead>
<tr>
<th>zSeries 900 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>zSeries 900, Model 101</td>
<td>11585.8</td>
<td>0.000086</td>
</tr>
<tr>
<td>zSeries 900, Model 102</td>
<td>10891.8</td>
<td>0.000092</td>
</tr>
<tr>
<td>zSeries 900, Model 103</td>
<td>10430.2</td>
<td>0.000096</td>
</tr>
<tr>
<td>zSeries 900, Model 104</td>
<td>10081.9</td>
<td>0.000099</td>
</tr>
<tr>
<td>zSeries 900, Model 105</td>
<td>9732.4</td>
<td>0.000103</td>
</tr>
<tr>
<td>zSeries 900, Model 106</td>
<td>9384.2</td>
<td>0.000107</td>
</tr>
<tr>
<td>zSeries 900 processor model</td>
<td>Service units per second of task or SRB execution time</td>
<td>Seconds Task or SRB Execution Time Per Service Unit</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>--------------------------------------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>zSeries 900, Model 107</td>
<td>9153.3</td>
<td>0.000109</td>
</tr>
<tr>
<td>zSeries 900, Model 108</td>
<td>8805.7</td>
<td>0.000114</td>
</tr>
<tr>
<td>zSeries 900, Model 109</td>
<td>8456.7</td>
<td>0.000118</td>
</tr>
<tr>
<td>zSeries 900, Model 110</td>
<td>9334.9</td>
<td>0.000107</td>
</tr>
<tr>
<td>zSeries 900, Model 111</td>
<td>9211.3</td>
<td>0.000109</td>
</tr>
<tr>
<td>zSeries 900, Model 112</td>
<td>8968.6</td>
<td>0.000112</td>
</tr>
<tr>
<td>zSeries 900, Model 113</td>
<td>8724.1</td>
<td>0.000115</td>
</tr>
<tr>
<td>zSeries 900, Model 114</td>
<td>8602.2</td>
<td>0.000116</td>
</tr>
<tr>
<td>zSeries 900, Model 115</td>
<td>8359.5</td>
<td>0.00012</td>
</tr>
<tr>
<td>zSeries 900, Model 116</td>
<td>8117.7</td>
<td>0.000123</td>
</tr>
<tr>
<td>zSeries 900, Model 1C1</td>
<td>12112</td>
<td>0.000083</td>
</tr>
<tr>
<td>zSeries 900, Model 1C2</td>
<td>11502.5</td>
<td>0.000087</td>
</tr>
<tr>
<td>zSeries 900, Model 1C3</td>
<td>11142.1</td>
<td>0.00009</td>
</tr>
<tr>
<td>zSeries 900, Model 1C4</td>
<td>10781.7</td>
<td>0.000093</td>
</tr>
<tr>
<td>zSeries 900, Model 1C5</td>
<td>10540.2</td>
<td>0.000095</td>
</tr>
<tr>
<td>zSeries 900, Model 1C6</td>
<td>10296</td>
<td>0.000097</td>
</tr>
<tr>
<td>zSeries 900, Model 1C7</td>
<td>10056.6</td>
<td>0.000099</td>
</tr>
<tr>
<td>zSeries 900, Model 1C8</td>
<td>9816</td>
<td>0.000102</td>
</tr>
<tr>
<td>zSeries 900, Model 1C9</td>
<td>9575.1</td>
<td>0.000104</td>
</tr>
<tr>
<td>zSeries 900, Model 210</td>
<td>11165.3873</td>
<td>0.000090</td>
</tr>
<tr>
<td>zSeries 900, Model 211</td>
<td>10869.5652</td>
<td>0.000092</td>
</tr>
<tr>
<td>zSeries 900, Model 212</td>
<td>10723.8606</td>
<td>0.000093</td>
</tr>
<tr>
<td>zSeries 900, Model 213</td>
<td>10430.2477</td>
<td>0.000096</td>
</tr>
<tr>
<td>zSeries 900, Model 214</td>
<td>10139.4170</td>
<td>0.000099</td>
</tr>
<tr>
<td>zSeries 900, Model 215</td>
<td>9993.7539</td>
<td>0.000100</td>
</tr>
<tr>
<td>zSeries 900, Model 216</td>
<td>9696.9697</td>
<td>0.000103</td>
</tr>
<tr>
<td>zSeries 900, Model 2C1</td>
<td>14692.3783</td>
<td>0.000068</td>
</tr>
<tr>
<td>zSeries 900, Model 2C2</td>
<td>13961.6056</td>
<td>0.000072</td>
</tr>
<tr>
<td>zSeries 900, Model 2C3</td>
<td>13377.9264</td>
<td>0.000075</td>
</tr>
<tr>
<td>zSeries 900, Model 2C4</td>
<td>13082.5838</td>
<td>0.000076</td>
</tr>
<tr>
<td>zSeries 900, Model 2C5</td>
<td>12638.2306</td>
<td>0.000079</td>
</tr>
<tr>
<td>zSeries 900, Model 2C6</td>
<td>12345.6790</td>
<td>0.000081</td>
</tr>
<tr>
<td>zSeries 900, Model 2C7</td>
<td>12048.1928</td>
<td>0.000083</td>
</tr>
</tbody>
</table>
### Table 25. zSeries 900 processor models (continued)

<table>
<thead>
<tr>
<th>zSeries 900 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>zSeries 900, Model 2C8</td>
<td>11756.0617</td>
<td>0.000085</td>
</tr>
<tr>
<td>zSeries 900, Model 2C9</td>
<td>11461.3181</td>
<td>0.000087</td>
</tr>
</tbody>
</table>

### Table 26. zSeries 890 processor models

<table>
<thead>
<tr>
<th>zSeries 890 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>zSeries 890, Model 110</td>
<td>1264.9221</td>
<td>0.000791</td>
</tr>
<tr>
<td>zSeries 890, Model 210</td>
<td>1225.2087</td>
<td>0.000816</td>
</tr>
<tr>
<td>zSeries 890, Model 310</td>
<td>1200.3901</td>
<td>0.000833</td>
</tr>
<tr>
<td>zSeries 890, Model 410</td>
<td>1175.6062</td>
<td>0.000851</td>
</tr>
<tr>
<td>zSeries 890, Model 120</td>
<td>2238.0753</td>
<td>0.000447</td>
</tr>
<tr>
<td>zSeries 890, Model 220</td>
<td>2167.7280</td>
<td>0.000461</td>
</tr>
<tr>
<td>zSeries 890, Model 320</td>
<td>2123.7059</td>
<td>0.000471</td>
</tr>
<tr>
<td>zSeries 890, Model 420</td>
<td>2079.8128</td>
<td>0.000481</td>
</tr>
<tr>
<td>zSeries 890, Model 130</td>
<td>4280.3638</td>
<td>0.000234</td>
</tr>
<tr>
<td>zSeries 890, Model 230</td>
<td>4146.1519</td>
<td>0.000241</td>
</tr>
<tr>
<td>zSeries 890, Model 330</td>
<td>4061.9447</td>
<td>0.000246</td>
</tr>
<tr>
<td>zSeries 890, Model 430</td>
<td>3978.1203</td>
<td>0.000251</td>
</tr>
<tr>
<td>zSeries 890, Model 140</td>
<td>5344.0214</td>
<td>0.000187</td>
</tr>
<tr>
<td>zSeries 890, Model 240</td>
<td>5176.3183</td>
<td>0.000193</td>
</tr>
<tr>
<td>zSeries 890, Model 340</td>
<td>5071.3154</td>
<td>0.000197</td>
</tr>
<tr>
<td>zSeries 890, Model 440</td>
<td>4965.8597</td>
<td>0.000201</td>
</tr>
<tr>
<td>zSeries 890, Model 150</td>
<td>8346.3745</td>
<td>0.000120</td>
</tr>
<tr>
<td>zSeries 890, Model 250</td>
<td>8084.8914</td>
<td>0.000124</td>
</tr>
<tr>
<td>zSeries 890, Model 350</td>
<td>7920.7921</td>
<td>0.000126</td>
</tr>
<tr>
<td>zSeries 890, Model 450</td>
<td>7755.6956</td>
<td>0.000129</td>
</tr>
<tr>
<td>zSeries 890, Model 160</td>
<td>10184.5958</td>
<td>0.000098</td>
</tr>
<tr>
<td>zSeries 890, Model 260</td>
<td>9864.3650</td>
<td>0.000101</td>
</tr>
<tr>
<td>zSeries 890, Model 360</td>
<td>9661.8357</td>
<td>0.000104</td>
</tr>
<tr>
<td>zSeries 890, Model 460</td>
<td>9461.8569</td>
<td>0.000106</td>
</tr>
<tr>
<td>zSeries 890, Model 170</td>
<td>17797.5528</td>
<td>0.000056</td>
</tr>
<tr>
<td>zSeries 890, Model 270</td>
<td>17241.3793</td>
<td>0.000058</td>
</tr>
<tr>
<td>zSeries 890, Model 370</td>
<td>16895.4593</td>
<td>0.000059</td>
</tr>
<tr>
<td>zSeries 890, Model 470</td>
<td>16546.0186</td>
<td>0.000060</td>
</tr>
</tbody>
</table>
### Table 27. zSeries 800 processor models

<table>
<thead>
<tr>
<th>zSeries 800 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>zSeries 800, Model 0E1</td>
<td>1955.0</td>
<td>0.000512</td>
</tr>
<tr>
<td>zSeries 800, Model 0A1</td>
<td>3907.2</td>
<td>0.000256</td>
</tr>
<tr>
<td>zSeries 800, Model 0X2 UNI</td>
<td>4239.5</td>
<td>0.000236</td>
</tr>
<tr>
<td>zSeries 800, Model 0X2</td>
<td>3900.5</td>
<td>0.000256</td>
</tr>
<tr>
<td>zSeries 800, Model 0B1</td>
<td>5612.0</td>
<td>0.000178</td>
</tr>
<tr>
<td>zSeries 800, Model 0C1</td>
<td>6980.8</td>
<td>0.000143</td>
</tr>
<tr>
<td>zSeries 800, Model 0A2 UNI</td>
<td>6893.6</td>
<td>0.000145</td>
</tr>
<tr>
<td>zSeries 800, Model 0A2</td>
<td>6341.7</td>
<td>0.000158</td>
</tr>
<tr>
<td>zSeries 800, Model 001</td>
<td>9334.9</td>
<td>0.000107</td>
</tr>
<tr>
<td>zSeries 800, Model 002</td>
<td>8588.3</td>
<td>0.000116</td>
</tr>
<tr>
<td>zSeries 800, Model 003</td>
<td>8121.8</td>
<td>0.000123</td>
</tr>
<tr>
<td>zSeries 800, Model 004</td>
<td>7843.1</td>
<td>0.000128</td>
</tr>
</tbody>
</table>

### Table 28. S/390 9672 processor models

<table>
<thead>
<tr>
<th>S/390 9672 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds of Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>S/390 9672, Model T16</td>
<td>6067.5</td>
<td>0.000165</td>
</tr>
<tr>
<td>S/390 9672, Model T26</td>
<td>5643.7</td>
<td>0.000177</td>
</tr>
<tr>
<td>S/390 9672, Model R36</td>
<td>5460.8</td>
<td>0.000183</td>
</tr>
<tr>
<td>S/390 9672, Model R46</td>
<td>5278.8</td>
<td>0.000189</td>
</tr>
<tr>
<td>S/390 9672, Model R56</td>
<td>5158.0</td>
<td>0.000194</td>
</tr>
<tr>
<td>S/390 9672, Model R66</td>
<td>5036.2</td>
<td>0.000199</td>
</tr>
<tr>
<td>S/390 9672, Model R76</td>
<td>4915.5</td>
<td>0.000203</td>
</tr>
<tr>
<td>S/390 9672, Model R86</td>
<td>4733.7</td>
<td>0.000211</td>
</tr>
<tr>
<td>S/390 9672, Model R96</td>
<td>4490.6</td>
<td>0.000223</td>
</tr>
<tr>
<td>S/390 9672, Model RX6</td>
<td>4247.4</td>
<td>0.000235</td>
</tr>
<tr>
<td>S/390 9672, Model Y16</td>
<td>7246.4</td>
<td>0.000138</td>
</tr>
<tr>
<td>S/390 9672, Model Y26</td>
<td>6739.7</td>
<td>0.000148</td>
</tr>
<tr>
<td>S/390 9672, Model Y36</td>
<td>6449.0</td>
<td>0.000155</td>
</tr>
<tr>
<td>S/390 9672, Model Y46</td>
<td>6230.5</td>
<td>0.000161</td>
</tr>
<tr>
<td>S/390 9672, Model Y56</td>
<td>6086.0</td>
<td>0.000164</td>
</tr>
<tr>
<td>S/390 9672, Model Y66</td>
<td>5941.3</td>
<td>0.000168</td>
</tr>
<tr>
<td>S/390 9672, Model Y76</td>
<td>5797.1</td>
<td>0.000173</td>
</tr>
<tr>
<td>S/390 9672, Model Y86</td>
<td>5578.8</td>
<td>0.000179</td>
</tr>
<tr>
<td>S/390 9672, Model Y96</td>
<td>5361.9</td>
<td>0.000187</td>
</tr>
</tbody>
</table>
### Table 28. S/390 9672 processor models (continued)

<table>
<thead>
<tr>
<th>S/390 9672 processor model</th>
<th>Service units per second of task or SRB execution time</th>
<th>Seconds of Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>S/390 9672, Model YX6</td>
<td>5071.3</td>
<td>0.000197</td>
</tr>
<tr>
<td>S/390 9672, Model RA6</td>
<td>4212.7</td>
<td>0.000237</td>
</tr>
<tr>
<td>S/390 9672, Model RB6</td>
<td>3960.4</td>
<td>0.000253</td>
</tr>
<tr>
<td>S/390 9672, Model R16</td>
<td>5588.5</td>
<td>0.000179</td>
</tr>
<tr>
<td>S/390 9672, Model R26</td>
<td>5141.4</td>
<td>0.000194</td>
</tr>
<tr>
<td>S/390 9672, Model RC6</td>
<td>5029.9</td>
<td>0.000199</td>
</tr>
<tr>
<td>S/390 9672, Model RD6</td>
<td>4918.5</td>
<td>0.000203</td>
</tr>
<tr>
<td>S/390 9672, Model RX7</td>
<td>8346.4</td>
<td>0.000120</td>
</tr>
<tr>
<td>S/390 9672, Model X27</td>
<td>7928.6</td>
<td>0.000126</td>
</tr>
<tr>
<td>S/390 9672, Model X37</td>
<td>7677.5</td>
<td>0.000130</td>
</tr>
<tr>
<td>S/390 9672, Model X47</td>
<td>7511.7</td>
<td>0.000133</td>
</tr>
<tr>
<td>S/390 9672, Model X57</td>
<td>7262.8</td>
<td>0.000138</td>
</tr>
<tr>
<td>S/390 9672, Model X67</td>
<td>7095.3</td>
<td>0.000141</td>
</tr>
<tr>
<td>S/390 9672, Model X77</td>
<td>6762.5</td>
<td>0.000148</td>
</tr>
<tr>
<td>S/390 9672, Model X87</td>
<td>6512.0</td>
<td>0.000154</td>
</tr>
<tr>
<td>S/390 9672, Model X97</td>
<td>6177.6</td>
<td>0.000162</td>
</tr>
<tr>
<td>S/390 9672, Model XX7</td>
<td>6010.5</td>
<td>0.000166</td>
</tr>
<tr>
<td>S/390 9672, Model XY7</td>
<td>5759.5</td>
<td>0.000174</td>
</tr>
<tr>
<td>S/390 9672, Model XZ7</td>
<td>5592.5</td>
<td>0.000179</td>
</tr>
<tr>
<td>S/390 9672, Model Z17</td>
<td>9667.7</td>
<td>0.000103</td>
</tr>
<tr>
<td>S/390 9672, Model Z27</td>
<td>9184.8</td>
<td>0.000109</td>
</tr>
<tr>
<td>S/390 9672, Model Z37</td>
<td>8893.8</td>
<td>0.000112</td>
</tr>
<tr>
<td>S/390 9672, Model Z47</td>
<td>8700.4</td>
<td>0.000115</td>
</tr>
<tr>
<td>S/390 9672, Model Z57</td>
<td>8412.2</td>
<td>0.000119</td>
</tr>
<tr>
<td>S/390 9672, Model Z67</td>
<td>8217.8</td>
<td>0.000122</td>
</tr>
<tr>
<td>S/390 9672, Model Z77</td>
<td>7831.6</td>
<td>0.000128</td>
</tr>
<tr>
<td>S/390 9672, Model Z87</td>
<td>7540.1</td>
<td>0.000133</td>
</tr>
<tr>
<td>S/390 9672, Model Z97</td>
<td>7249.7</td>
<td>0.000138</td>
</tr>
<tr>
<td>S/390 9672, Model ZX7</td>
<td>6959.5</td>
<td>0.000144</td>
</tr>
<tr>
<td>S/390 9672, Model ZY7</td>
<td>6765.3</td>
<td>0.000148</td>
</tr>
<tr>
<td>S/390 9672, Model ZZ7</td>
<td>6475.1</td>
<td>0.000154</td>
</tr>
</tbody>
</table>
Table 29. S/390 3000 processor models

<table>
<thead>
<tr>
<th>S/390 3000, Model</th>
<th>Seconds of Task or SRB Execution Time Per Service Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model A10</td>
<td>1792.1</td>
</tr>
<tr>
<td></td>
<td>0.000558</td>
</tr>
<tr>
<td>Model A20</td>
<td>1582.3</td>
</tr>
<tr>
<td></td>
<td>0.000632</td>
</tr>
</tbody>
</table>

If you plan to use these constants for purposes other than those suggested in this section, observe the following limitations:

- Actual customer workloads and performance may vary. For a more exact comparison of processors, see the internal throughput rate (ITR) numbers in *Large Systems Performance Reference (LSPR)*.
- CPU time can vary for different runs of the same job step. One or more of the following factors might cause variations in the CPU time: CPU architecture (such as storage buffering), cycle stealing with integrated channels, and the amount of the queue searching (see z/OS MVS System Management Facilities (SMF)).
- The constants do not account for multiprocessor effects within logical partitions. For example, a logical 1-way partition in an S/390 9672, Model RX3, has 1090 service units per second, while a 10-way partition on the same machine has 839.3 service units per second.

**Using SMF Task Time**

For installations with no prior service data, the task time reported in SMF record Type 4, 5, 30, 34, and 35 records can be converted to service units using the preceding tables.

**Examples of Resource Groups**

- To give a department a “dedicated 4381” amount of capacity, you can specify the following resource group:

  **Name**  4381-91E
  **Description**  Capacity of a 4381
  **Capacity Minimum**  309
  **Capacity Maximum**  309

- To assure half the capacity of a 9672 Model Y96 system, with 9 physical CPs, to a service class, specify the following:

  **Name**  HALFSYS3
  **Description**  Preserve half of SYS3 (9672 Mod Y96)
  **Capacity Maximum**  24129

Since the 9672 Model Y96 has 5362 raw CPU service units per second, with 9 physical CPs, 0.5 x 9 x 5362 is 24129. The 24129 is captured service units allocated to the address spaces in the resource group.
Appendix C. Return Codes of Sample Job IWMINSTL

This information describes the return codes issued by sample job IWMINSTL. The Install Definition Utility is shipped as member IWMINSTL in SYS1.SAMPLIB. IWMINSTL can be used to install a WLM service definition, or to activate a WLM policy without having to use the ISPF WLM application.

Table 30 describes the return codes that are issued by IWMINSTL. An accompanying message text is written to the job output listing.

Table 30. Return Codes from IWMINSTL

<table>
<thead>
<tr>
<th>Return Code</th>
<th>Message Text and Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful execution.</td>
</tr>
<tr>
<td>104</td>
<td>The service definition was not processed due to a mismatch between the WLM address space level (current.wlm.level) and the level of this utility (IWMARIDU.level). Use the correct level of the IWMARIDU utility.</td>
</tr>
<tr>
<td>204</td>
<td>GetServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>Unable to use (your.definition.dataset) for service definition data. A table has an unrecognized format.</td>
</tr>
<tr>
<td>208</td>
<td>GetServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>Unable to use (your.definition.dataset), data set is in use.</td>
</tr>
<tr>
<td>212</td>
<td>GetServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>The service definition was not read due to a mismatch between the service definition PDS and the WLM Install Definition Utility. The service definition PDS has a higher functionality level than the current version of the WLM Install Definition Utility. Once a service definition was updated by a WLM instance with a higher functionality level it can no longer be updated by a WLM instance with lower functionality level.</td>
</tr>
<tr>
<td>216</td>
<td>GetServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>The service definition was not opened due to an ISPF dialog error.</td>
</tr>
<tr>
<td>304</td>
<td>InstallServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>Install failed, service definition has no name. No service definition name was specified. To fix the error, do the following:</td>
</tr>
<tr>
<td></td>
<td>• Specify a valid service definition name for IWMINSTL parameter SVDEFPDS.</td>
</tr>
<tr>
<td></td>
<td>• Verify in the IWMINSTL JCL, whether the SVDEF DD statement exists.</td>
</tr>
<tr>
<td>308</td>
<td>InstallServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>Install failed, no workloads are defined.</td>
</tr>
<tr>
<td>312</td>
<td>InstallServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>Install failed, no service classes are defined.</td>
</tr>
<tr>
<td>316</td>
<td>InstallServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>Install failed, no service policies are defined.</td>
</tr>
<tr>
<td>Return Code</td>
<td>Message Text and Explanation</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>320</td>
<td>InstallServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>Install failed, access was denied to the WLM couple data set.</td>
</tr>
<tr>
<td></td>
<td>The UserID does not have proper access to MVSADMIN.WLM.POLICY in the FACILITY class. See for more information.</td>
</tr>
<tr>
<td>324</td>
<td>InstallServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>Install failed, service definition was modified</td>
</tr>
<tr>
<td></td>
<td>Every WLM service definition contains a service definition ID. An attempt was made to install a service definition with a different ID than the service definition ID that is installed. A WLM service definition ID consists of the following:</td>
</tr>
<tr>
<td></td>
<td>• The name of the service definition</td>
</tr>
<tr>
<td></td>
<td>• A timestamp when the service definition was installed</td>
</tr>
<tr>
<td></td>
<td>• A user ID that installed the service definition</td>
</tr>
<tr>
<td></td>
<td>• The system name from which the installation was done</td>
</tr>
<tr>
<td></td>
<td>Use the FORCE=Y parameter for the WLM install definition utility to install a service definition with a different ID. The default is FORCE=N.</td>
</tr>
<tr>
<td>336</td>
<td>InstallServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>Install failed, failure in WLM bridge layer.</td>
</tr>
<tr>
<td></td>
<td>Additional messages regarding this error might be written to the job output listing.</td>
</tr>
<tr>
<td>404</td>
<td>ActivateServicePolicy</td>
</tr>
<tr>
<td></td>
<td>The service policy was not activated due to an ISPF dialog error.</td>
</tr>
<tr>
<td>408</td>
<td>ActivateServicePolicy</td>
</tr>
<tr>
<td></td>
<td>Activate failed, activation in progress on another system.</td>
</tr>
<tr>
<td>412</td>
<td>ActivateServicePolicy</td>
</tr>
<tr>
<td></td>
<td>Activate failed, one or more systems were unable to activate the policy.</td>
</tr>
<tr>
<td>416</td>
<td>ActivateServicePolicy</td>
</tr>
<tr>
<td></td>
<td>Activate failed, (POLNAME) was not found on the WLM couple data set.</td>
</tr>
<tr>
<td>420</td>
<td>ActivateServicePolicy</td>
</tr>
<tr>
<td></td>
<td>Activate failed, failure in WLM bridge layer.</td>
</tr>
<tr>
<td></td>
<td>Additional messages regarding this error might be written to the job output listing.</td>
</tr>
<tr>
<td>516</td>
<td>SaveServiceDefinition</td>
</tr>
<tr>
<td></td>
<td>The service definition was not opened due to an ISPF dialog error.</td>
</tr>
</tbody>
</table>
Appendix D. 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 may view the information through the z/OS Internet Library website or the z/OS Information 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
U.S.A.

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. Refer to z/OS TSO/E Primer, z/OS TSO/E User’s Guide, and z/OS ISPF User’s Guide Vol I 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:

http://www.ibm.com/systems/z/os/zos/bkserv/
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 [http://www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other company, product, and service names may be trademarks or service marks of others.
Notices

This information was developed for products and services offered in the U.S.A. or elsewhere.

IBM may not offer 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 character set (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.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites 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 wish 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
2455 South Road
Poughkeepsie, NY 12601-5400
USA

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.

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.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

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.

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.

COPYRIGHT LICENSE:

This information contains 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.

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.
Workload Management Terms

A

**active service policy.** The service policy that determines workload management processing.

**application environment.** A group of application functions requested by a client that execute in server address spaces.

**application owning region (AOR).** In a CICSpex configuration, a CICS region devoted to running applications.

**automatic control.** One of two distinct methods of managing application environments. Under automatic control, the name of the startup JCL procedure has been defined for an application environment, giving workload management the ability to automatically start server address spaces. Contrast with manual control.

C

**channel path identifier.** The channel subsystem communicates with I/O devices by means of a channel path between the channel subsystem and devices. A CHPID is a value assigned to each channel path of the System z® that uniquely identifies that path. Up to 256 CHPIDs are supported for each channel subsystem.

**CICSpex.** A configuration of interconnected CICS systems in which each system is dedicated to one of the main elements of the overall workload. See also application owning region, file owning region, and terminal owning region.

**classification rules.** The rules workload management and subsystems use to assign a service class and, optionally, a report class to a work request. A classification rule consists of one or more of work qualifiers such as subsystem type, subsystem instance, userid, accounting information, transaction name, transaction class, source LU, netid, and LU name.

**compatibility mode.** Prior to z/OS V1R3 a mode of processing, in which the IEAISSxx and IEAICSxx parmlib members determine system resource management. See also goal mode.

**control region.** The main storage region that contains the subsystem work manager or subsystem resource manager control program.

**couple data set.** A data set created through the XCF couple data set format utility. The data set is shared by MVS systems in a sysplex. There are several types of couple data sets for different purposes. See also WLM couple data set.

**CPU service units.** A measure of the task control block (TCB) execution time multiplied by an SRM constant which is CPU model dependent. See also unweighted CPU service units per second, and service unit.

D

**delay monitoring services.** The workload management services that monitor the delays encountered by a work request.

**distributed data facility (DDF).** An optional feature that allows a DB2 application to access data at other DB2s and at remote relational database systems that support IBM’s Distributed Relational Database Architecture™ (DRDA).

**duration.** The length of a service class performance period in service units.

**dynamic alias management.** A service definition option — when enabled, workload management will dynamically reassign parallel access volume aliases to help work meet its goals and to minimize IOS queueing.

E

**enclave.** A transaction that can span multiple dispatchable units (SRBs and tasks) in one or more address spaces and is reported on and managed as a unit.

**execution velocity.** A service goal naming the rate at which you expect work to be processed for a given service class or a measure of the acceptable processor and storage delays while work is running.

F

**fold qualifier names.** When defining classification rules, the “Fold qualifier names” option, when set to the default Y, means that the qualifier names will be folded to upper case as soon as you type them in and then press Enter. If you set this option to N, then the qualifier names will remain in the case they are typed in.


**goal mode.** A mode of processing where the active service policy determines system resource management.

**guest platform management provider (GPMP).** An optional suite of applications that is installed in specific z/OS, Linux, and AIX® operating system images to support platform management functions. For example, the guest platform management provider collects and aggregates performance data for virtual servers and workloads. Users view these reports through the ensemble-management Hardware Management Console (HMC).

**Hardware Management Console (HMC).** A user interface through which data center personnel configure, control, monitor, and manage System z hardware and software resources. The HMC communicates with each central processor complex (CPC) through the CPC’s Support Element (SE). On a zEnterprise 196, using the Unified Resource Manager on the HMCs/SEs, personnel can also create and manage an ensemble.

**HMC.** Accepted acronym for Hardware Management Console.

**hypervisor.** A program that allows multiple instances of operating systems or virtual servers to run simultaneously on the same hardware device. A hypervisor can run directly on the hardware, can run within an operating system, or can be imbedded in platform firmware. Examples of hypervisors include PR/SM, zVM, and PowerVM™.

**IBM System z.** Family name. After first use can be System z.

**IBM System z Application Assist Processor (zAAP).** A specialized processor that provides a Java™ execution environment, which enables Java-based Web applications to be integrated with core z/OS business applications and backend database systems.

**IBM z Enterprise Unified Resource Manager (Unified Resource Manager).** Utilize on “first use”. Then can use z Enterprise Unified Resource Manager or short form Unified Resource Manager. See “Unified Resource Manager”.

**IBM System z Integrated Information Processor (zIIP).** A specialized processor that provides computing capacity for selected data and transaction processing workloads, and for selected network encryption workloads.

**importance level.** The degree of importance of a service goal relative to other service class goals, in five levels: lowest, low, medium, high, highest.

**intranode management network (INMN).** A private 1000BASE-T Ethernet network operating at 1 Gbps that is required for the Unified Resource Manager to manage the resources within a single zEnterprise node. The INMN connects the Support Element (SE) to the z196 and to any attached zBX.

**I/O priority management.** A service definition option — when enabled, I/O priorities will be managed separately from dispatching priorities, according to the goals of the work.

**I/O service units.** A measure of individual data set I/O activity and JES spool reads and writes for all data sets associated with an address space.

**installation.** A particular computing system, including the work it does and the people who manage it, operate it, apply it to problems, service it, and use the results it produces.

**installed service definition.** The service definition residing in the WLM couple data set for WLM.

**logical unit (LU).** In VTAM, the source and recipient of data transmissions. Data is transmitted from one logical unit (LU) to another LU. For example, a terminal can be an LU, or a CICS or IMS system can be an LU.

**LU.** Logical unit.

**LU name.** The second level of the source LU name after the “.” for fully qualified names.

**LU 6.2 session.** A session that is initiated by VTAM* programs on behalf of a logical unit (LU) 6.2 application program, or a session initiated by a remote LU in which the application program specifies that the VTAM programs are to control the session by using the APPCCMD macro instruction.

**manual control.** One of two distinct methods of managing application environments in goal mode. Under manual control, the name of the startup JCL procedure has not been defined for an application environment. The installation must therefore manually start server address spaces when needed. Contrast with automatic control.

**masking.** Using a % for a single character replacement in classification rules. See also wild carding.
MVS image. The system-id of any MVS system included in the sysplex as it appears in SYS1.PARMLIB member.

P

performance administration. The process of defining and adjusting workload management goals and resource groups based on installation business objectives.

performance block. A piece of storage containing workload management's record of execution delay information about work requests.

performance management. The process workload management uses to decide how to match resources to work according to performance goals and processing capacity.

performance period. A service goal and importance level assigned to a service class for a specific duration. You define performance periods for work that has changing performance requirements as work consumes resources.

policy. See service policy.

R

relational database management system (RDBMS). A relational database manager that supports SAA.

report class. A group of work for which reporting information is collected separately. For example, you can have a report class for information combining two different service classes, or a report class for information on a single transaction.

resource. When used as part of a scheduling environment, a resource is an abstract element that can represent an actual physical entity (such as a peripheral device), or an intangible quality (such as a certain time of day). A resource is listed in a scheduling environment along with a required state of ON or OFF. If the corresponding resource state on a given system matches the required state, then the requirement is satisfied for that resource.

resource group. An amount of processing capacity across one or more MVS images, assigned to one or more service classes.

S

scheduling environment. A list of resource names along with their required states. If an MVS image satisfies all of the requirements in the scheduling environment associated with a given unit of work, then that unit of work can be assigned to that MVS image. If any of the requirements are not satisfied, then that unit of work cannot be assigned to that MVS image.

server address space. Any address space that does work on behalf of a transaction manager or a resource manager. For example, a server address space could be a CICS AOR, or an IMS control region.

service administration application. The online ISPF application used by the service administrator to specify the workload management service definition.

service class. A group of work which has the same performance goals, resource requirements, or business importance. For workload management, you assign a service goal and optionally a resource group to a service class.

service coefficient. A value that specifies which type of resource consumption should be emphasized in the calculation of service rate. The types of resource consumption are CPU, IOC, MSO, and SRB.

service definition. A definition of the workloads and classification rules in an installation. The definition includes workloads, service classes, systems, resource groups, service policies, and classification rules.

service level administrator. The user role introduced by workload management whose main task is to make sure overall installation operation is consistent with performance goals and objectives.

service level agreement (SLA). A written agreement of the information systems (I/S) service to be provided to the users of a computing installation.

service policy. A named set of performance goals workload management uses as a guideline to match resources to work. See also active service policy.

service request block (SRB) service units. A measure of the SRB execution time for both local and global SRBs, multiplied by an SRM constant which is CPU model dependent.

service unit. The amount of service consumed by a work request as calculated by service definition coefficients and CPU, SRB, I/O, and storage service units.

single-system sysplex. A sysplex in which only one MVS system is initialized as part of the sysplex. In a single-system sysplex, XCF provides XCF services on the system, but does not provide signalling services between MVS systems. See also multi-system sysplex, XCF-local mode, and monoplex.

source LU. A fully qualified two level name separated by a “.”, where the first level is the network id and the second is the LU name, OR merely a single LU name. See also LU name.
storage service units. A measure of the central storage page frames multiplied by 1/50 of the CPU service units. The 1/50 is a scaling factor designed to bring the storage service component in line with the CPU component.

subsystem instance. 1) For application environments, a unique combination of subsystem type (as specified in the service definition for an application environment) and subsystem name (as specified by the work manager subsystem when it connects to workload management). 2) For classification, a work qualifier used to distinguish multiple instances of a subsystem.

subsystem work manager. An address space defined in the SYS1.PARMLIB member as SUBSYS=nnn.

T

terminal owning region (TOR). A CICS region devoted to managing the terminal network.

U

Unified Resource Manager. Full name: IBM zEnterprise Unified Resource Manager. Licensed Internal Code (LIC), also known as firmware, that is part of the Hardware Management Console. The Unified Resource Manager provides energy monitoring and management, goal-oriented policy management, increased security, virtual networking, and data management for the physical and logical resources of a given ensemble.

unweighted CPU service units per second. The unweighted service units per second of task or SRB execution time. This measure is CPU-model dependent, but is independent of the values of the service coefficients.

V

velocity. A service goal naming the rate at which you expect work to be processed for a given service class or a measure of the acceptable processor and storage delays while work is running.

W

wild carding. The use of an asterisk (*) as a multiple character replacement in classification rules. See also masking.

WLM couple data set. A type of couple data set that is created through the XCF couple data set format utility for the WLM function. The data set contains the service definition information.

workload. A group of work to be tracked, managed and reported as a unit. Also, a group of service classes.

workload management mode. The mode in which workload management manages system resources on an MVS image. Prior to z/OS V1R3 this mode could be either compatibility mode, or goal mode. Starting with z/OS V1R3, compatibility mode has been removed.

work qualifier. An attribute of incoming work. Work qualifiers include: subsystem type, subsystem instance, userid, accounting information, transaction name, transaction class, source LU, netid, and LU name.

work request. A piece of work, such as a request for service, a batch job, an APPC, CICS, or IMS transaction, a TSO LOGON, or a TSO command.

Z

zAAP. Approved acronym for IBM System z Application Assist Processor (zAAP). See full name.

zIIP. Approved acronym for IBM System z Integrated Information (zIIP). See full name.
Index

A
access
restricting 144
accessibility 261
accounting information
nesting 64
qualifier 64
action field 186
active service policy
definition 269
adjusting velocity goals 53
administer WLM
with z/OS Management Facility 229
administration application
definition 271
alias management, dynamic 96
APPC/MVS scheduler (ASCH) 76
application
action field 186
classification groups 205
command line 186
commands 186
create a group panel 205
create workload panel 194
customizing keylists 240
definition menu 188
function keys 187
scrollable area 185
starting 235, 238, 239
using the menu bar 184
workload selection list 194
workloads 194
application environment
definition 269
application environments 120
authorizing 123
CB (WebSphere Application Server) 113
classification
creating for a subsystem type 200
definition 269
classification rules
creating for a subsystem type 200
definition 269
collection name
definition 65
compliance mode
definition 269
connection type
definition 66
correlation information
definition 269
couple data set
allocating 148
calculating the size of 148
definition 269
increasing the size of 150
installing service definition 219
restricting access to 144
average response time (continued)
limit 48
couple data set (continued)
SEXTXCF command 151
updating COUPLExx 151
COUPLExx
member
DATA keyword 151
coupling facility
defining a structure 157, 161
CPSM environment 20
CPU protection 48, 100
CPU service units
definition 269, 272
customer information control system (CICS) 76

B
balancing of WLM managed batch initiators 20, 155
batch initiator balancing 20, 155
batch initiator management 156
batch initiators 21
business importance
definition 7
coupling facility
definition 269
coupling facility structure 161
discretionary goal
48
discretionary goal management
migration considerations 154
discretionary goals
using 54
dispatch mode 23
distributed data facility (DDF) 76
duration
definition 49
in a performance period 49
dynamic alias management 96
definition 269
migration considerations 154
dynamic channel path management 163
coupling facility structure 161
disability 261
discretionary goal 48
discretionary goal management
migration considerations 154
discretionary goals
using 54
dispatch mode 23
distributed data facility (DDF) 76
duration
definition 49
in a performance period 49
dynamic alias management 96
definition 269
migration considerations 154
dynamic channel path management 163
coupling facility structure 161

E
enclave
definition 269
Enterprise Storage Server 96
execution velocity
calculation 96
definition 269
with I/O priority management 96

© Copyright IBM Corp. 1994, 2011
fold qualifier names
definition 269
option explained 201
functionality level 146

goal
definition 47
goal types 18
goals, performance
defining 6, 50
GPMG (guest platform management provider) 135
group capacity 30
guest platform management provider (GPMP) 135

Hardware Management Console (HMC)
definition 270
heterogeneous report class 91
HiperDispatch mode 23
HMC
definition 270
homogeneous report class 91
hypervisor
definition 270

I/O priority
management 96
I/O priority management
definition 270
I/O priority queueing
defining 33
dynamic 33
I/O service units
definition 270
I/O storage management 25
IBM zEnterprise (zEnterprise)
processor models z196 241
importance
definition 270
importance levels
in performance periods 55
IMS (information management system)
overview of work 60
work qualifiers supported by 76
information management system (IMS) 76
inheritance
in classification rules 79
initiators, batch 21
Install Definition Utility
member IWMINSTL 142
member IWMSSDEF 142
installation exit (continued)
IWMAREX2 235
installed service definition
definition 270
Integrated Information Processor (zIIP) 179
intelligent resource director
coupling facility priority queueing 164
coupling facility structure 161
dynamic channel path management 163
LPAR CPU Management 163
making it work 166
intranode management network (INMN)
definition 270
IRD
coupling facility structure 161
IWEB
application environments 113
overview of work 60
work qualifiers supported by 76
IWMAM040 message 225
IWMAM041 message 225
IWMAM042 message 225
IWMAM043 message 225
IWMAM044 message 225
IWMAM046 message 225
IWMAM050 message 225
IWMAM051 message 226
IWMAM052 message 226
IWMAM054 message 226
IWMAM055 message 226
IWMAM058 message 226
IWMAM072 message 226
IWMAM077 message 227
IWMAM098 message 227
IWMAM313 message 227
IWMAREX1 installation exit 234
IWMAREX2 installation exit 235
IWMINSTL 142
IWMINSTL, return codes 259
IWMSSDEF 142
JES (job entry subsystem)
overview of work 60
work qualifiers supported by 76
JES2 (job entry subsystem 2)
overview of work 60
work qualifiers supported by 76
JES2 batch initiators 21
JES3 (job entry subsystem 3)
overview of work 60
work qualifiers supported by 76
JES3 batch initiators 21
job entry subsystem (JES) 76

keyboard 261

nesting
accounting information 80
subsystem parameter 80
service definition 13
    base 13
    contents 33
    creating for the first time 141
    defining 33
    definition 5, 271
    hierarchy 13
    installing 142, 219
    printing 223
    printing as GML 223
    restricting access to 144
    storing in MVS PDS 183
    storing in WLM couple data set 183
-service level agreement (SLA) 271
-service policy 36
    activating 223
    create service policy panel 193
    defining 6, 35
    defining overrides 198
    definition 271
    in service definition 35
    overview 14
    printing 223
-service unit 271
    processor model 241
    task/SRB execution time 241
-SETXCF command 151
    example 151
    shark 96
    shortcut keys 261
-SOM 113
    application environments 113
    overview of work 60
    work qualifiers supported by 76
-source LU definition 271
-SRB (service request block) execution time 241
    processor model 241
    service units 241
-SRB service units definition 271
    start position 84
    using 84
-started task control (STC) 76
    started tasks 84
        defining goals for 54
        servers 54
    started tasks (STC) 87
        defining classification rules 87
        defining service classes 87
        overview of work 60
        work qualifiers supported by 76
-STC (started task control) definition 272
    overview of work 241
        work qualifiers supported by 76
-storage coefficient 95
    using 95
-storage management I/O 25
-storage protection 99
-storage service units definition 272
-substring notation examples 85
    subsystem collection name qualifier 68
    subsystem instance definition 272
    qualifier 68
    subsystem parameter qualifier 69
    subsystem type creating 204
    deleting 205
    modifying rules for 200
    subsystem types in classification 64
    SYS_ 126
    SYSI.PARMLIB member IEAOPT 24
    SYSI.SAMPLIB member IVMINSTL 142
        member IMSSDEF 142
    SYSH overview of work 60
        work qualifiers supported by 76
    SYSOTHER service class 84
    sysplex couple data set upgrading 141
    sysplex name qualifier 70
    SYSTC service class 83
    SYSTC1-SYSTC5 service class 84
    SYSTEM service class 82
    system name qualifier 70
    system response time goals determining 51
    System z10 EC processor models 244
    System z9 BC processor models 247
    System z9 EC processor models 249
-task processor model 241
    terminal owning region (TOR) definition 272
    transaction class qualifier 70
    transaction name qualifier 71
    transaction server management, modifications of 101
    TSO overview of work 60
    work qualifiers supported by 76
-U Unified Resource Manager definition 272
    Unified Resource Manager performance policy, service class qualifier 73
-user ID qualifier 72
-using group capacity 30
-z/OS Management Facility 229
-V velocity definition 272
    formula 50
    limit 48
    migration considerations 153
    velocity goals adjusting 53
    defining 53
    samples included in 53
-W WebSphere Application Server (CB) 76
    wildcarding 272
    wildcard notation examples 86
    WLM couple data set allocating 148
        calculating the size of 148
        increasing the size of 150
        installing service definition 219
        restricting access to 144
        SEXTXCF command 151
        updating COUPLExx 151
        work environments 17
        work qualifier definition 272
        work request definition 272
        workload definition 272
        in service definition 39
        workload balancing 19
        definition 5
        Workload Management task overview 229
-Z z/OS Basic Skills information center xv
    z/OS Management Facility 229
    z/OS UNIX System Services (OMVS) 76
    z196 processor models 241
    zAAP definition 270
    zAAP (zSeries Application Assist Processor) 173
zIIP
- definition 270
- zIIP (Integrated Information Processor) 179
  - zSeries 800 processor models 255
  - zSeries 890 processor models 254
  - zSeries 900 processor models 252
  - zSeries 990 processor models 251
- zSeries Application Assist Processor 173