Note

Before using this information and the product it supports, be sure to read the general information under “Notices” on page 525.

This edition applies to Version 1 Release 11 of z/OS (5694-A01) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC33-7988-07.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address below.

IBM welcomes your comments. A form for readers’ comments is provided at the back of this publication. If the form has been removed, address your comments to:
IBM Deutschland Research & Development GmbH
Department 3248
Schönaicher Str. 220
D-71032 Böblingen
Federal Republic of Germany

Or by FAX:
From Germany: 07031-16-3456
Other countries: +(49)+7031-16-3456

Or electronically:
Internet e-mail: s390id@de.ibm.com


If you would like a reply, be sure to include your name, address, telephone number, or FAX number.

Make sure to include the following in your comment or note:
• Title and order number of this book
• Page number or topic related to your comment

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
Figures

1. Multiple data set configuration definition without HCD ............ 2
2. Single data set configuration definition with HCD .................. 3
3. Configuration definition with HCD ................................ 6
4. Objects managed by HCD ..................................... 6
5. Building a production IODF and an IOCDS ......................... 7
6. Relationship of data sets used by HCD ............................... 8
7. Processor configuration with coupling facility implemented .......... 10
8. Sample ISPF Master Application Menu ................................ 21
9. HCD Primary Task Selection panel .................................. 23
10. HCD Profile Options ............................................. 25
11. Example of an HCD Profile ........................................ 33
12. Create Work I/O Definition File .................................... 37
13. IODF Distribution/Merge Process .................................... 41
14. Maintain I/O Definition Files ........................................ 43
15. Create New I/O Definition File ....................................... 44
16. View IODF information .............................................. 45
17. Define JCL for IODF Import .......................................... 46
18. Specify Target IODF and User Password ............................. 47
19. Upgrade an IODF .................................................... 48
20. Create Work I/O Definition File ....................................... 49
21. Sample Configuration Package List — left panel .................. 50
22. Add Configuration Package .......................................... 51
23. Transmit Configuration Package ...................................... 53
24. Example of an Activity Log without automatic logging .......... 54
25. View Activity Log ................................................... 56
26. Example of a Panel Layout .......................................... 59
27. Panel Layout of a Data Entry Panel .................................. 61
28. Primary Task Selection panel ....................................... 62
29. Example of a single selection list .................................... 63
30. Example of a Multiple Selection List .................................. 63
31. Action List Panel with a Context Menu ............................. 64
32. Action List Panel with Action Codes ................................. 65
33. Message List .......................................................... 66
34. Explanation Message ................................................. 67
35. Example of a Prompt Selection Panel ................................ 67
36. Moving within a Multi-Step Task ..................................... 70
37. Using the Goto Action Bar Choice .................................... 71
38. Operating System Configuration List as example of a Selection List 71
39. Filter example .......................................................... 74
40. Filter I/O Device List ................................................ 74
41. Reduced I/O Device List (using a filter) ............................ 75
42. Navigation Map ....................................................... 80
43. Add a processor ....................................................... 92
44. Available support levels ............................................. 93
45. CTC Connection Update List ....................................... 103
46. Result of the Transfer (Move) Partition Configs Action ......... 110
47. Confirm Delete Partition ............................................. 111
48. Add Channel Path ................................................... 114
49. Channel Path/Partition Matrix ....................................... 116
50. CF Channel Path Connectivity List ................................... 120
51. Connect to CF Channel Path ......................................... 120
52. Add CF Control Unit and Devices .................................... 121
53. Select Processor / Control Unit (1) .................................. 131
54. Select Processor / Control Unit (2) .................................. 131
55. Change Channel Path Link Addresses ............................... 133
56. I/O Device List with device groups .................................. 139
57. I/O Device List with single devices .................................. 140
58. Add Device ............................................................. 141
59. Device / Processor Definition ........................................ 142
60. Define Device / Processor ............................................ 142
61. Define Device Candidate List ......................................... 144
62. Specify Subchannel Set ID .......................................... 145
63. Specify Subchannel Set ID .......................................... 151
64. Eligible Channel Subsystems ....................................... 152
65. Eligible Operating System Configurations .......................... 152
66. CTC/CNC connection established using a dynamic connection .... 163
67. CTC/CNC connection established using shared channels.......... 164
68. Configuration with one switch ....................................... 170
69. Configuration with two switches (1) ................................ 170
70. Configuration with two switches (2) ................................ 171
71. Configuration with two cascading FICON switches ............... 171
72. Switch List (left part) ............................................... 173
73. Switch List (right part) .............................................. 173
74. Add Switch ........................................................... 174
75. Change Switch Definition ............................................ 175
76. Confirm Priming Switch Data List ................................... 176
77. Move Ports to a Target Switch ....................................... 177
78. Confirm Delete Switch ............................................... 178
79. Port List ............................................................. 179
80. Activate or Process Configuration Data ............................. 197
81. Build Production I/O Definition File ................................. 198
82. Confirm Production IODF Name ..................................... 199
83. Confirm Delete I/O Definition File ................................... 199
84. Define Descriptor Fields ............................................. 199
85. IOCDS List ........................................................... 201
86. S/390 Microprocessor Cluster List .................................. 204
87. IOCDS List ........................................................... 205
88. Build IOCP Input Data Set .......................................... 208
89. Example of an input data set for migration enhancements ....... 212
90. Identify System I/O Configuration .................................. 217
91. Activate New Configuration: Hardware and Software Changes . 221
92. Activate Sysplex Member List ....................................... 224
93. Activate Software Configuration Only ............................... 225
94. Message panel with ACTIVATE messages ......................... 226
95. View Configuration Status .......................................... 227
96. Activate Hardware and Software Configuration ..................... 227
97. Restrict Ports Eligible for Dynamic CHPID Management .......... 229
98. Build CONFIGxx Member .......................................... 229

© Copyright IBM Corp. 1994, 2009
Tables

1. Syntax examples ............................... xvi
2. IODF compatibility overview .............. 18
3. Batch Jobs Used by the HCD Dialog ...... 34
4. Size considerations when upgrading a back-level IODF .......... 49
5. Coexistence Considerations for IODFs. ...... 57
6. Online Help Information ................. 68
7. Actions on IODF Objects. ................. 292
8. Standard DD names Used by HCD .......... 337
9. CBD.CPC.IOCDs access authority and HCD IOCDS management functions .......... 341
10. Load Module Libraries to be defined to program control. ............... 353
11. APF authorized Load Module Libraries .... 354
12. Symptoms of system problems ............ 447
13. Search Argument ............................ 449
14. Problem Data ............................... 449
15. Search Argument ............................ 450
16. Problem Data ............................... 450
17. Search Argument ............................ 450
18. Problem Data ............................... 451
19. Search Argument ............................ 451
20. Problem Data ............................... 451
21. Search Argument ............................ 453
22. Search Argument ............................ 453
23. Problem Data ............................... 453
24. Problem Data ............................... 454
25. Problem Data ............................... 455
26. Search Argument ............................ 457
27. Problem Data ............................... 457
28. Search Argument ............................ 457
29. Problem Data ............................... 458
30. Search Argument ............................ 458
31. Problem Data ............................... 458
32. Problem Data ............................... 462
33. Used registers and passed parameters .... 475
34. Summary of Request Block Names and Related Constants .......... 476
About this document

This document explains how to use Hardware Configuration Definition (HCD) to accomplish the following:
- Define new hardware configurations
- Modify existing hardware configurations
- View existing hardware configurations
- Activate configurations
- Query supported hardware
- Maintain IODFs
- Compare two IODFs
- Compare an IODF with an actual configuration
- Print reports of configurations
- Create graphical reports of a configuration
- Migrate existing configuration data

Note

Unless otherwise noted, the term “MVS” as used in this document refers collectively to the older MVS operating system as well as to its successors OS/390 and z/OS, in which MVS is one of the basic components. For purposes of this document, "MVS" used alone and without reference to a specific release is to be understood as the generic operating system type supported by HCD.

Who this document is for

This document is for the person who is responsible for defining the hardware configuration for a z/OS system. It is assumed that:
- The person has a basic knowledge of z/OS, and hardware configuration.
- Configuration planning has been completed before HCD is used to enter definition data. For information on configuration planning, refer to z/OS HCD Planning.

For the person responsible for problem determination, this document also explains what to do if a problem arises with HCD.

This information is for system programmers and system operators.

How to use this document

Before you start to use HCD, you should read
- Chapter 1, “Hardware configuration definition - What is it?,” on page 1
- Chapter 5, “How to use the dialog,” on page 59

These topics provide information about the general concepts and facilities of HCD.

When you want to use HCD to define a new configuration, you should read
- Chapter 5, “How to use the dialog,” on page 59
- Chapter 6, “How to define, modify, or view a configuration,” on page 77
- Chapter 7, “How to work with switches,” on page 169
Chapter 5 explains how to use the HCD panels, get online help information, enter data and select items such as tasks, objects and actions. Chapter 6 and Chapter 7 explain how to define operating system (OS) configurations, processors, control units, I/O devices, and switches.

**How this document is organized**

- **Chapter 1**, “Hardware configuration definition - What is it?,” on page 1 explains how HCD fits into the context of hardware configurations and systems management. It also explains the environment in which HCD is used.

- **Chapter 2**, “Migration,” on page 13 discusses how to move from a previous HCD release to HCD under z/OS. It also refers to other sections in this document dealing with migration and conversion tasks.

- **Chapter 3**, “How to set up, customize and start HCD,” on page 19 provides information on how to install, customize, and start HCD, and how to set up an HCD installation for the first time in z/OS.

- **Chapter 4**, “How to work with I/O definition files (IODF),” on page 35 explains how to work with I/O definition files (IODFs), for example, creating, changing, viewing, and deleting them. It also explains how to use configuration packages to create subset IODFs for distribution.

- **Chapter 5**, “How to use the dialog,” on page 59 explains the general facilities of the HCD dialog, that is panels, online help, navigation, making selections, and entering data.

- **Chapter 6**, “How to define, modify, or view a configuration,” on page 77 explains the navigation through the HCD dialog and how to define, change, copy, delete, and view operating system configurations, processors, control units, and devices. It also explains how to prime processor, control unit, and device data.

- **Chapter 7**, “How to work with switches,” on page 169 includes basic information about switches and explains how to define, change, prime, and delete switches, how to define and prime switch connections (ports), and how to work with switch configurations. It also describes how to migrate, activate, and save switch configuration data.

- **Chapter 8**, “How to activate or process configuration data,” on page 195 explains how to make a configuration available for use by the system. It further explains how to compare a configuration defined in an IODF with the configuration sensed on the system. It includes information on activating a configuration dynamically, activating a configuration sysplex-wide, and on remote IOCDS management functions.

- **Chapter 9**, “How to print and compare configuration data,” on page 235 explains how to build textual and graphical reports about channel subsystem, switch, and operating system configuration, I/O paths, and CTC definitions and how to compare IODFs. It also explains how to print the data that is currently displayed on a list panel.

- **Chapter 10**, “How to query supported hardware and installed UIMs,” on page 253 explains how to use HCD to view system data.
Chapter 11, “How to migrate existing input data sets,” on page 261 contains information for migrating existing IOCP/MVSCP/HCPRIO definitions and explains the steps in the migration process.

Chapter 12, “How to invoke HCD batch utility functions,” on page 307 describes the HCD programming interface.

Chapter 13, “Security and other considerations,” on page 339 provides information on various HCD-related topics.

Chapter 14, “How to provide LDAP support for HCD,” on page 347 explains the provision for retrieving and updating IODF data via LDAP protocol.

“Appendix A. How to navigate through the dialog” on page 371 illustrates the flow from the HCD main panel options and the various actions that can be taken from each option.

“Appendix B. Configuration reports” on page 377 contains examples of the various reports that can be printed by using HCD.

“Appendix C. Problem determination for HCD” on page 447 explains what to do if problems occur with HCD.

“Appendix D. HCD object management services” on page 475 explains how to use the HCD application programming interface to retrieve configuration data, such as switch data, device type, or control unit type, from the IODF.

“Appendix E. Scenarios” on page 483 contains scenarios which cover the main definition tasks required to produce the IODF illustrated in the diagram at the beginning of the appendix.

“Appendix F. IODF data model” on page 505 describes the IODF data model in terms of object classes and attribute definitions as used by the HCD LDAP backend.

How to read syntax diagrams

This section describes how to read syntax diagrams. It defines syntax diagram symbols, items that may be contained within the diagrams (keywords, variables, delimiters, operators, fragment references, operands) and provides syntax examples that contain these items.

Syntax diagrams pictorially display the order and parts (options and arguments) that comprise a command statement. They are read from left to right and from top to bottom, following the main path of the horizontal line.

Symbols

The following symbols may be displayed in syntax diagrams:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Symbol 1" /></td>
<td>Indicates the beginning of the syntax diagram.</td>
</tr>
<tr>
<td><img src="image2.png" alt="Symbol 2" /></td>
<td>Indicates that the syntax diagram is continued to the next line.</td>
</tr>
<tr>
<td><img src="image3.png" alt="Symbol 3" /></td>
<td>Indicates that the syntax is continued from the previous line.</td>
</tr>
<tr>
<td><img src="image4.png" alt="Symbol 4" /></td>
<td>Indicates the end of the syntax diagram.</td>
</tr>
</tbody>
</table>
Syntax items

Syntax diagrams contain many different items. Syntax items include:

- **Keywords** - a command name or any other literal information.
- **Variables** - variables are italicized, appear in lowercase, and represent the name of values you can supply.
- **Delimiters** - delimiters indicate the start or end of keywords, variables, or operators. For example, a left parenthesis is a delimiter.
- **Operators** - operators include add (+), subtract (-), multiply (*), divide (/), equal (=), and other mathematical operations that may need to be performed.
- **Fragment references** - a part of a syntax diagram, separated from the diagram to show greater detail.
- **Separators** - a separator separates keywords, variables or operators. For example, a comma (,) is a separator.

**Note:** If a syntax diagram shows a character that is not alphanumeric (for example, parentheses, periods, commas, equal signs, a blank space), enter the character as part of the syntax.

Keywords, variables, and operators may be displayed as required, optional, or default. Fragments, separators, and delimiters may be displayed as required or optional.

<table>
<thead>
<tr>
<th>Item type</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required</td>
<td>Required items are displayed on the main path of the horizontal line.</td>
</tr>
<tr>
<td>Optional</td>
<td>Optional items are displayed below the main path of the horizontal line.</td>
</tr>
<tr>
<td>Default</td>
<td>Default items are displayed above the main path of the horizontal line.</td>
</tr>
</tbody>
</table>

Syntax examples

The following table provides syntax examples.

<table>
<thead>
<tr>
<th>Item</th>
<th>Syntax example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required item.</td>
<td><img src="image" alt="Syntax example" /></td>
</tr>
<tr>
<td>Required items appear on the main path of the horizontal line. You must specify these items.</td>
<td><img src="image" alt="Syntax example" /></td>
</tr>
<tr>
<td>Required choice.</td>
<td><img src="image" alt="Syntax example" /></td>
</tr>
<tr>
<td>A required choice (two or more items) appears in a vertical stack on the main path of the horizontal line. You must choose one of the items in the stack.</td>
<td><img src="image" alt="Syntax example" /></td>
</tr>
<tr>
<td>Optional item.</td>
<td><img src="image" alt="Syntax example" /></td>
</tr>
<tr>
<td>Optional items appear below the main path of the horizontal line.</td>
<td><img src="image" alt="Syntax example" /></td>
</tr>
<tr>
<td>Item</td>
<td>Syntax example</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Optional choice.</td>
<td>An optional choice (two or more items) appears in a vertical stack below the main path of the horizontal line. You may choose one of the items in the stack.</td>
</tr>
<tr>
<td>Default.</td>
<td>Default items appear above the main path of the horizontal line. The remaining items (required or optional) appear on (required) or below (optional) the main path of the horizontal line. The following example displays a default with optional items.</td>
</tr>
<tr>
<td>Variable.</td>
<td>Variables appear in lowercase italics. They represent names or values.</td>
</tr>
<tr>
<td>Repeatable item.</td>
<td>An arrow returning to the left above the main path of the horizontal line indicates an item that can be repeated. A character within the arrow means you must separate repeated items with that character. An arrow returning to the left above a group of repeatable items indicates that one of the items can be selected, or a single item can be repeated.</td>
</tr>
<tr>
<td>Fragment.</td>
<td>The fragment symbol indicates that a labelled group is described below the main syntax diagram. Syntax is occasionally broken into fragments if the inclusion of the fragment would overly complicate the main syntax diagram.</td>
</tr>
</tbody>
</table>
Summary of changes

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

What is new in HCD for z/OS V1R11

Summary of changes for SC33-7988-08 issued September 2009

Here is a summary of updates that have been introduced to HCD for z/OS® V1R11.

Note: To see a description of new functionality of the current release, select option 9. What’s new in this release from the HCD Primary Task Selection panel.

Here you may find information about SPEs that are delivered after the completion of this document.

HCD Profile Options dialog

HCD provides a new Profile Options dialog which displays the current values of all possible HCD profile keywords as they are either explicitly set in the HCD profile data set or as they are defaulted by HCD. This dialog allows you to verify or modify HCD profile options.

New profile option

There are new profile keywords for the following purposes:

• Showing partition information in IOCP decks: Use profile option SHOW_IOCP_DEFAULTS to explicitly show and document the partition defaults of CHPID and IODEVICE statements in generated IOCP decks. The option causes HCD to write this additional information as comments into the IOCP deck.

• Extend the change log data set to provide space for updates: Use profile option CHLOG_EXTENSION to specify the percentage of additional space that is to be allocated beyond the default size of the IODF when a change log data set is created.

Changes in HCD reports

The following HCD reports are enhanced:

• The Control Unit Summary Report now shows the number of logical paths per control unit.

• The Control Unit Detail Report now shows the number of logical paths for a control unit per channel subsystem and the number of logical paths per control unit port.

• The EDT Report now shows values for columns NAME, NAME TYPE, VIO and TOKEN also for an esoteric that has no device numbers assigned to it.

• For the Processor Compare Report, HCD now also compares the processor tokens of work/production or work/work IODFs. In previous releases, HCD compared only the tokens of production IODFs.
In the CSS/OS Device Compare Report, devices with adjacent ranges are joined before printing. Additionally the device range is increased to show a maximum of 4 digits.

**Indicating channel path mode changes**

When working with channel paths in the HCD dialog, users might specify a partition assignment that does not match the specified channel path mode, or vice versa. For example, a user sets the operation mode of a channel path to SHR (shared), but partitions of multiple channel subsystems are assigned to it. In such a case, HCD does not accept the user’s channel path mode specification but implicitly adjusts it to the defined partition assignment. In the given example, HCD changes the channel path mode to SPAN (spanned).

Starting with this release, HCD issues a new informational message to inform the user that the channel path mode is adjusted to reflect the specified partition assignment.

**Specifying multi-user access mode when creating a work IODF from a production IODF**

You can now choose multi-user access mode for a new work IODF that you create from a production IODF.

**Support of the WWPN Prediction Tool**

From the currently accessed IODF, you can export the FCP channel/device specific part of the I/O configuration into a comma-separated value (CSV) output format. The generated file can be used as input to the WWPN Prediction Tool to assign world-wide port names to virtualized FCP ports for configuring SAN devices.

**Miscellaneous dialog enhancements**

**Enhanced prompting for available IODFs**

The Available IODFs panel invoked by prompting for IODFs indicates whether an IODF can be accessed in multi-user access or in single-user access mode only.

**Enhanced Channel Path List**

The Channel Path List is enhanced to show the processor ID when the user scrolls to the right to see which partitions are in the access and candidate lists for a channel path (partition matrix).

**Enhanced deleting of partitions with CHPIDs uniquely assigned**

A partition, that has CHPIDs uniquely assigned to only this partition in either the access or candidate list, cannot be deleted until users first remove the CHPIDs from the partition. This is indicated by a message. However, this message identifies only one CHIPID at a time, that is exclusively assigned to this partition, and may reoccur several times, for each CHIPID uniquely assigned to the partition. Thus, deleting a partition could become a cumbersome and inefficient task.

Therefore, the Confirm Delete Partition dialog is enhanced to indicate all CHPIDs exclusively assigned to this single partition by flagging them with an ‘*’. Thus, users can remove the flagged CHPIDs in one step and then delete the partition more efficiently.
Functions available as SPE since z/OS V1R10

The following features have been available as SPEs since z/OS V1R10, but are now described in detail in this edition for the first time.

**Hardware support**
- HCD supports the IBM z10 BC processor (processor type 2098-E10).
- For the z/TPF operating system, 3215 consoles are supported on channel paths of type OSC. IOCP needs to distinguish between OSC-3270 and OSC-3215 attachments. This is done via the CHPARM keyword on the CHPID statement.

**History of changes**

**What is new in HCD for z/OS V1R10**

**Multi-user access**
When creating an IODF, you can specify a multi-user access option, so that multiple users can simultaneously update this IODF. The IODF is kept in exclusive update mode only for the duration of a single transaction. If the updates of the transaction are committed, another user may update the IODF without requiring the first user to release it.

Users can switch between single-user and multi-user access using a new option in the Change I/O Definition File Attributes dialog.

**Changed HCD LDAP support**
The HCD LDAP backend now works as a plug-in to the IBM Tivoli Directory Server for z/OS. Support of the z/OS Integrated Security Services LDAP Server has been discontinued.

**CUADD value is shown on the Control Unit List**
A new column CUADD is introduced on the Control Unit List to display the defined logical address for a control unit, where available.

**Changes to the ‘Switch IOCDS’ request**
Whenever an HCD Switch IOCDS request is performed, an informational message (CBDG323I) will be written from the HCD session to the console.

**Enhanced spanning of channel paths**
Up to z/OS V1R9 HCD, connected control units and devices were not defined for the new CSS when spanning a channel path to a new CSS. Users had to do this explicitly by invoking an action from the Control Unit List.

Starting with this release, when spanning a channel path with connected control unit(s) (and devices) to a new CSS, HCD invokes a dialog asking whether the control units should also be reachable from the new CSS.

**New profile option**
There is one new keyword that you can specify in the HCD profile for the following purpose:
- **Delay device regrouping**: You can specify that HCD should perform any necessary device regrouping after a device group split only when the IODF is closed. This gives a better response time in the HCD dialog for large IODFs, because HCD per default performs a necessary device regrouping each time when exiting the I/O Device List.
Functions available as SPE since z/OS V1R9
The following functions have been available as SPEs since z/OS V1R9, but are now described in detail in this edition for the first time.

Hardware support:  HCD supports the IBM z10 EC processor family (processor types 2097-E12, -E26, -E40, -E56, -E64).

Selecting a 2097-type processor creates a maximum logical processor configuration. That is, HCD generates a processor configuration with four channel subsystems and 15 reserved partitions in each channel subsystem. Also, HCD generates two subchannel sets in each of the four channel subsystems and provides for the maximum number of devices in each subchannel set.

New channel path type for coupling over InfiniBand:  The use of InfiniBand technology is supported by a new channel path type CIB to emulate coupling connections on the Host Communication Adapter (HCA). Support is provided on type 2094, 2096 and 2097 processors to define coupling links and STP links between CPCs.

What is new in HCD for z/OS V1R9
Summary of changes for SC33–7988–06 issued September 2007

Here is a summary of updates that have been introduced to HCD for z/OS V1R9.

New and enhanced profile options
There are several new keywords that you can specify in the HCD profile for the following purposes:

• Allow or prohibit mixed esoterics: In previous releases, HCD allowed a mix of DASD and TAPE devices under the same esoteric name. Though HCD continued to build the production IODF, it issued warning message CBDA332I. Starting with this release, using the new profile keyword MIXED_ESOTERIC, users can decide whether HCD should prohibit mixed esoterics by issuing CBDA332I as error message rather than a warning and thus stop building the production IODF.

• Select a HLQ for exporting IODFs: When exporting an IODF, the generated data set is normally written with the high-level qualifier of the user ID that issued the Export IODF function. To allow the specification of a different HLQ, you can use the keyword EXPORTED_HLQ in the HCD profile. The HLQ (up to 8 characters) specified with this keyword is used for the exported IODF data set name rather than the user ID.

This function is available as SPE and needs to be installed as APAR OA18724 starting with z/OS 1.4.

• Import IODF in data space: The setting of keyword IODF_DATA_SPACE is now also applied to the Import I/O definition file function. To avoid size restrictions imposed on large imported IODFs by address space limitations, the specification of IODF_DATA_SPACE=YES (which is the default) causes HCD to import the IODFs into a data space instead of importing them into an address space.

This function is available as SPE and needs to be installed as APAR OA16701 starting with z/OS 1.4.

• Extend MCF data set to provide space for updates: The setting of keyword MCF_EXTENSION allows users to specify the percentage of additional space that is to be allocated when defining an MCF data set. Per default an MCF data set is allocated with 30 percent additional space than actually needed to hold the MCF data, thereby providing relief for updates that consume data space.
• Select a volume for the HCM master configuration file: Keyword MCF_VOL lets you specify a volume serial number where to store the HCM master configuration file (MCF) if you want to exploit HCM’s MCF concept. For information on the MCF concept, refer to the z/OS and z/VM HCM User’s Guide.

• Overwriting the default allocation for data sets allocated due to HCM requests: The new HCD profile option ALLOC_SPACE allows users to overwrite the default allocation for host data sets that are temporarily allocated due to HCM requests, for example, HCDASMP, HCDRPT, HCDIN.

  This function is available as SPE and needs to be installed as APAR OA19771 for z/OS 1.7 and 1.8.

Checking for multiple extents of production IODFs
A z/OS IPL using a production IODF with multiple extents results in a wait state. Starting with this release, HCD issues warning message CBDA009I, if dynamic activation encounters an IODF with multiple extents.

  This function is available as SPE and needs to be installed as APAR OA17546 starting with z/OS 1.4.

Filter for occupied ports
On the Port List for switches, you can now filter for occupied ports.

Show/Hide setting in the I/O Device List
The Show/Hide setting of parameters or features in an I/O Device List is now saved across HCD sessions.

Enhancements for transmitting configuration packages
When defining a configuration package, the target user ID and node ID as well as the status of the receiving operating system (attended or unattended) are remembered in the IODF as part of the package definition. In earlier releases, you could not change the target during package submission.

  Starting with this release, the Transmit Configuration Package dialog allows you to overwrite the target user ID, target node ID and transmission mode (attended or unattended).

Removed support
Support for processor types 9021 and 9121 has been removed.

  If you access processor support modules for these processor types, you get message CBDA046I during HCD initialization.

Functions available as SPE since z/OS V1R7
The following functions have been available as SPEs for z/OS 1.7, but are now described in detail in this edition for the first time.

  Hardware support: HCD supports new IBM System z9 Business Class, type 2096-R07 and 2096-S07 processors.

  Server Time Protocol (STP) link support: HCD supports Server Time Protocol (STP) links between two zSeries (z890, z990, or higher) processors. In the Connect to CF Channel Path dialog, users can select two CHPIDs capable for coupling facilities, and then specify the Timing-only link option to create an STP link.
Automatic activity log generation: You can enable automatic logging of change activities on IODFs. For this purpose, use the new HCD profile entry \texttt{CHANGE\_LOG = YES}, and request activity logging when creating the IODF. You can edit the generated log entries before you exit the IODF.

With the new profile options \texttt{CHLOG\_VOL} and \texttt{ACTLOG\_VOL}, in non SMS-managed environments, you can specify the volume serial numbers where to allocate a new change log or activity log.

Enhancements of HCD configuration reports: The following configuration reports have been enhanced:

- The \textit{Processor Summary Report} displays the support level of the listed processors.
- The \textit{IOCDS Report} includes more information, which previously was only available from the \textit{IOCDS List} in the HCD dialog. New columns indicate whether the IOCDS token matches the current HSA token as well as the processor token in the IODF, whether it is write-protected and what is its status.
- The \textit{CF Channel Path Connectivity Report} now includes a new column showing the type of the connecting control unit.

What is new in HCD for z/OS V1R7

This section summarizes the updates that have been introduced to HCD for z/OS V1R7.

Hardware support

HCD supports \texttt{@server zSeries IBM System z9 Enterprise Class (z9\textsuperscript{©} EC) processors (type 2094)} with multiple subchannel sets.

Support of multiple subchannel sets

Starting with z9 EC processors, users can define an additional subchannel set with ID 1 (SS 1) on top of the existing subchannel set (SS 0) in a channel subsystem. With this additional subchannel set, you can configure more than 2*63K devices for a channel subsystem. With z/OS V1R7, you can define Parallel Access Volume (PAV) alias devices (device types 3380A, 3390A) of the 2105, 2107 and 1750 DASD control units to SS 1.

Support of OSA NCP channel path type

A new channel path type OSN (OSA NCP) is introduced which provides support for OS logical partitions that need to communicate with an external network using SNA protocols. This functionality was earlier provided by the 374x control units, which have been withdrawn from market. The NCP function is now being emulated in Linux via the Communication Controller for Linux (CCL) on zSeries.

Supporting more than 160 TCP/IP stacks for OSD channels

You can now decide to have more than 160 TCP/IP stacks supported with OSD channels. This is done by disabling priority queueing for these channels. If priority queuing is disabled, the channel can support four times as many queues (4 \* 480 = 1920 subchannels) corresponding to four times as many TCP/IP stacks (4 \* 160 = 640) as with enabled queue prioritization.

When defining or changing channel paths of type OSD for XMP processors, HCD prompts you with a dialog where you can decide whether you require more than 160 TCP/IP stacks for the channel in question.

Also, you can now connect more than one control unit to an OSD channel.
IODF size reduction
Due to the fact of increasing device definitions in the IODF, the IODF size limit of 2 GB is about to be approached. In addition, large IODFs involve a series of disadvantages. Therefore, z/OS V1.7 HCD starts using the new V5 IODF format, representing devices in device groups rather than containing individual device definitions. This significantly reduces the size of IODFs and improves the processing performance of large configurations.

How HCD maintains device groups within an IODF when you apply changes to single devices of a group is described in “How HCD arranges devices into groups in an IODF” on page 41.

Working with device groups
The HCD dialog reflects the new IODF structure in the I/O Device List (see “Defining device data” on page 138). The initial I/O Device List shows the device groups and lets you perform actions on the device groups as well as navigate to the single devices.

Enhanced CHPID aggregate function
The CHPID aggregate function now allows you to aggregate just a subset of control units from a source to a target CHPID. Also, the target CHPID may be connected to a different dynamic switch than the source CHPID.

Miscellaneous enhancements
Local download of an IOCDS: An IOCDS download can now be performed locally even if a SNA address has been defined to the processor.

Definition of FICON loopback port configuration: For a FICON switch matrix, HCD now allows you to define a dynamic port connection from a FICON port to itself (loopback connection).

Enhanced CTC Connection List and Report: The CTC Connection List/Report now also displays point-to-point CTC connections.

Enhanced View IODF panel: Besides the IODF version, the View IODF panel now shows the percentage of the used space that is actually utilized.

Enhanced Available IODFs panel: The Available IODFs panel, invoked by prompting for IODFs on the HCD Primary Task Selection panel, provides sort keys which you can use to sort the IODF list by IODF name, allocated size or creation date.

Improved PFSHOW handling: The PFSHOW command setting within HCD is retained across HCD sessions. In addition, the PFSHOW setting that is active before the invocation of HCD is saved and reset upon exit of HCD.

Automatic IODF check: There is a new profile option, CHECK_IODF, which you can specify to perform an automatic check for consistent IODF data when the currently allocated IODF is switched or the HCD dialog is terminated.

Counting rows of filtered lists: On panels that provide the Filter action bar choice, you can now use a new pull down choice Count rows on (filtered) lists to receive a message that displays the number of rows matching the current filter criteria.
Prompt for unused device numbers: On the Add Device panel, you can use a prompt (PF4) to retrieve unused device numbers and ranges in the current IODF. You can select a free device number and range from the displayed list for the definition of new devices.

CSS / OS Device Compare Report enhancement: The CSS / OS Device Compare report now provides an indication for devices that relate to the limiting LPAR via CHPIDs, but which are excluded from this LPAR with an explicit device candidate list and which therefore are not accessible from the current CSS.

What is new in HCD for z/OS V1R5

This section summarizes the updates that have been introduced to HCD for z/OS V1R5. To obtain this new functionality, you need to install PTF UA90070 for HCD if not mentioned otherwise.

You must also upgrade the support level of existing 2084 processor definitions.

Hardware support
HCD supports IBM @server zSeries 990 processors (type 2084 with the appropriate support level) or zSeries 890 processors (type 2086) in the following way:

- a maximum of 30 logical partitions per CPC
- a maximum of 4 channel subsystems (2084 processors only)
- a maximum of 2 channel subsystems (2086 processors only)
- spanned physical channel paths
- dynamic partitions

Note: On the Available Support Levels panel you can retrieve an explanation of the processor support level for zSeries processors: Position the cursor on the processor support level description and press PF1 to get an enumeration of functions provided by this support level.

Support of OSA Integrated Console Controller
A new channel path type OSC (OSA Integrated Console Controller) is introduced which provides 2074 control unit functions in an OSA Express2 channel.

Support of dynamic partitions
For IBM @server zSeries 990 and zSeries 890 processors with the appropriate hardware and software support levels, HCD provides the capability to add or remove LPARs via dynamic I/O configuration after a POR. You can define partitions with a placeholder ("*") in the IODF and change the "*" to a partition name later or vice versa. The "*" indicates that you plan to add a partition or to remove an existing partition dynamically by activating the configuration at a later point in time.

Support of spanned physical channels
On XMP processors, besides the internal channels ICP and IQD, the following physical channels may be spanned:

- CF peer and sender channels:
  - CBP, CFP, CBS, CFS
- OSA channels:
  - OSC, OSD, OSE
- FICON channels:
  - FC, FCP
Spanned channels have the same CHPID and PCHID values defined in all channel subsystems.

**Support for multiple control units on OSD channels**
Up to 16 control units may be defined on an OSD channel path, provided that each has a unique CU logical address (CUADD value). This will allow a single partition to use all 480 supported valid subchannels.

**Support of null device candidate lists for XMP processors**
If devices are connected to a control unit which is shared between multiple channel subsystems, some (not all) of these devices may specify a null device candidate list for one or more CSSs.

**Over-defining channel paths on an XMP processor**
For an XMP processor, you can define a channel path that is not physically installed on the machine. This may be useful if you want to migrate from a machine which had more channels defined than the target XMP processor has currently installed, or if you want to prepare a configuration for future upgrades of the channel cards. Such over-defined channel paths are excluded from the IOCDS and from dynamic activations. For more information, refer to "Over-defining channel paths on an XMP processor" on page 119.

**Automatic change of CTC connections during Copy action**
When copying channel subsystems or partitions within the same IODF, valid CTC connections in the source are not automatically copied and changed in the target configuration. To facilitate these changes, you get a list of affected CTC connections from which you can select those that should be updated. For the selected CTC connections, HCD performs the corresponding updates and copies them to the target configuration. For more information, refer to "Copying/repeating channel subsystems with CTC connections" on page 103.

**What is new in HCD for z/OS V1R4**
This section summarizes the updates that have been introduced to HCD in z/OS V1R4.

- "Support of multiple logical channel subsystems (LCSS)"
- "Support of physical channel identifiers" on page xxviii
- "Miscellaneous enhancements" on page xxviii
- "Hardware support" on page xxix

**Support of multiple logical channel subsystems (LCSS)**

<table>
<thead>
<tr>
<th>A note on terminolog:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Throughout this document, the following terms are used:</td>
</tr>
<tr>
<td><strong>XMP processor and SMP processor</strong></td>
</tr>
<tr>
<td>The term XMP processor designates processors that support multiple logical channel subsystems. It is used in contrast to the term SMP processor, which designates processors of previous generations that support only one channel subsystem.</td>
</tr>
</tbody>
</table>

More than 256 channel paths are supported on an XMP machine. This is achieved by allowing the definition of multiple logical channel subsystems (LCSS).
The HCD dialog is enhanced in a way that for XMP processors the definition of multiple logical channel subsystems is supported. A new object called channel subsystem is introduced into the object hierarchy below the processor object. For such processors, partitions and channel paths now pertain to a channel subsystem. For previous processor generations (SMP processors), the object hierarchy remains unchanged.

With XMP processors, supporting multiple logical channel subsystems, some types of channel paths can be shared across partitions belonging to different channel subsystems. Such a channel path is called a spanned channel path. For more information, refer to “Working with channel paths” on page 112.

Support of physical channel identifiers
Real I/O hardware is attached to a processor via physical channels. For XMP processors, the physical channels need a physical channel identifier (PCHID) which determines the physical location of a channel in the processor. For these processors, you have to specify the physical channel identifier (PCHID) related to the channel path identifier (CHPID).

The task of adding the physical channel path information to an IODF is eased in a way that HCD can cooperate with the CHPID Mapping Tool (CMT). Input to the CMT is the hardware configuration file of your machine (CFReport, see also in the “Glossary” on page 529) and a valid IOCP input file. Output from the CMT is again an IOCP input file that has the PCHID values filled in. This IOCP input file can be reimported into the IODF. For more information see “How to interact with the CHPID Mapping Tool” on page 213.

Miscellaneous enhancements

Enhanced IODF prompt: Besides IODF name and the volume serial number, the IODF prompt now shows the following attributes:
• the VSAM allocated blocks for the data object
• the creation date of the VSAM cluster

HCD profile changes: The default of the following keywords in the HCD profile have changed from NO to YES:
• IODF_DATA_SPACE
• SHOW_IO_CHANGES
• BATCH_IODF_NAME_CHECK

Redesigned Switch Configuration Detail Report: The Switch Configuration Detail Report was previously too extensive and not clearly arranged. Its format is now redesigned such that via grouping the information contained in that report will be highly condensed.

Limitation of CSS Report: When limiting a Channel Subsystem Report to a single partition, the report will show channel paths, control units and devices attached by the access list as well as those attached by the candidate list.

Default SIZE parameter of INITIODEF utility: For the SIZE parameter of the Initialize IODF (INITIODEF) utility, you can now specify zero (0) to get the default. HCD then tries to get the number of allocated blocks of the VSAM data set from the catalog and uses that value as the default. If you specify a size value greater than zero, HCD checks whether this value does not exceed the allocated size of the VSAM data set.
**Enhanced checking:** The following new checks are introduced:

- When copying/merging channel paths, HCD checks whether an existing target channel path is connected to a different switch port than the source channel path. HCD also checks whether an existing dedicated channel path in the target is reconnected to a different partition during copying/merging, because the CHPID already existed in the target. Appropriate warning messages are issued if necessary.
- For esoteric groups, HCD issues a warning message if:
  - you do not specify a token for esoterics in an EDT
  - you mix DASDs and TAPEs into a single esoteric group.
- A warning message is issued while building the production IODF in case more than one channel path is connected to the same switch port. Also a port that is already connected to a channel path will no longer be presented in the prompt when connecting a channel path with the same ID to the switch.

**Hardware support**

HCD supports IBM @server zSeries 990 (Type 2084) processors. These processors support:

- more than one channel subsystem within the processor complex
- more than 15 partitions throughout all defined channel subsystems
- more than 256 channel paths throughout all defined channel subsystems
- spanning for specific channel types

Each single channel subsystem has the following limitations:

- 256 channel paths
- 15 logical partitions
- 64K devices

**What is new in HCD for z/OS V1R3**

This section summarizes the updates that have been introduced to HCD in z/OS Version 1 Release 3 (z/OS V1R3).

- “Coupling facility duplexing” on page xxx
- “iQDIO and FCP channel support” on page xxx
- “PCTC support” on page xxx
- “Support for FICON cascade switching” on page xxx

**Note:**

This release of z/OS does not include a new release of HCD. For this reason, HCD is presented with its FMIDs of OS/390 Release 9.

An appendix with z/OS product accessibility information has been added.

**Coupling facility duplexing**

With coupling facility duplexing, a CF logical partition can use the CF sender to communicate with another CF logical partition. That means, you can define sender channel paths (CFS, CBS, ICS) besides the receiver channel paths (CFR, CBR, ICR) in a CF partition. CF Duplexing is supported starting with the 9672 Parallel Enterprise Servers G5 and G6 models and the 2064 zSeries models.
iQDIO and FCP channel support
HCD supports two new channel types:

- **IQD**: In zSeries processors, a virtual internal queued direct I/O (iQDIO) transport layer enables memory-to-memory message delivery across logical partitions. So, IQD channels will be used for Fast Message Passing between z/OS logical partitions and the Linux for zSeries partition.

- **FCP**: This channel path type is introduced to allow access to SCSI devices, for example, a DVD device, via the Fibre Channel Protocol from a Linux for zSeries image.

FCTC support
IBM @server zSeries 900 exploits the FICON capability to provide channel-to-channel host communication between processors. FICON CTC (FCTC) provides CTC communication at a higher bandwidth and with greater connectivity than with ESCON CTC.

In order to be usable as an FCTC connection channel, an FC channel path must be defined to an FCTC control unit which is connected to FCTC devices. At least one end of an FCTC connection must be a 2064 zSeries processor, since the FCTC control unit function is only contained in an FC channel of a 2064 zSeries machine at the corresponding EC level and follow-on machines.

Support for FICON cascade switching
HCD supports fabrics containing cascade switching using FICON switches. Such a fabric consists of two or more FICON switches. Within a fabric, the connection from a channel path to a control unit is dynamically established using the link address of the target control unit provided.

For addressing control units in FICON cascade switching environments, HCD supports the use of two-byte link addresses. The first byte specifies the switch address and the second byte specifies the port address of the FC switch to which the control unit is attached.

Hardware support
HCD supports zSeries 900 (Type 2066) processors.

What is new in HCD for z/OS V1R1
This section summarizes the updates that have been introduced to HCD in z/OS V1R1. If you are migrating from OS/390, you should review the information in the detailed section for each item.

Note:
This release of z/OS does not include a new release of HCD. For this reason, HCD is presented with its FMIDs of OS/390 Release 9.

Hardware support
- zSeries 900 (Type 2064) processor
- Support of native FICON (FC) channels and up to 288K HSA subchannels for System/390 Parallel Enterprise Server Generations 5 and 6.

Dynamic Channel Path Management (DCM)
As part of the z/OS Intelligent Resource Director, Dynamic Channel Path Management allows the system to dynamically re-assign channel paths to...
connected control units based on the current work load and its service goals. HCD allows channel paths to be designated as static (fixed) or managed (re-assignable) when they are defined.

**Support for peer coupling channels**
For the zSeries 900 processor, a new type of coupling channel, a *peer coupling channel*, has been introduced. HCD supports peer coupling channels via 3 new channel path types:
- CFP - Peer mode ISC-3, defined on ISC-3 links
- CBP - Peer mode ICB-3, defined on 1 GB STI link
- ICP - Peer mode IC-3, defined on uninstalled channel path numbers

**Migration support from ESCON to native FICON**
HCD allows intermixing both ESCON and native FICON channels on the same control unit. Thus, it is possible to switch from ESCON to native FICON channel channels on the control unit via two subsequent dynamic activates without the need to vary the devices off-line.

**Support of a generic fibre channel switch**
HCD supports a generic fibre channel switch (FC switch, type FCS), with available port addresses 00 to FF. This switch type does not support a switch control unit port.
Chapter 1. Hardware configuration definition - What is it?

Overview

This topic explains:

- What HCD is and how it differs from MVSCP and IOCP
- What HCD offers you
- How HCD works
- The environment in which HCD operates

What HCD is and how it differs from MVSCP and IOCP

The channel subsystem (CSS) and the IBM z/OS operating system need to know what hardware resources are available in the computer system and how these resources are connected. This information is called hardware configuration.

Hardware Configuration Definition (HCD) provides an interactive interface that allows you to define the hardware configuration for both a processor’s channel subsystems and the operating system running on the processor.

Before HCD was available, you had to use IOCP to define the hardware to the channel subsystem and the MVS Configuration Program (MVSCP) to define the hardware to the MVS operating system. The following sections explain in what way HCD differs from MVSCP and IOCP when defining, validating and reconfiguring configuration data.

Definition of configuration data

This topic informs about the differences between MVSCP and IOCP on the one hand and HCD on the other hand when performing the task of defining configuration data.

How MVSCP and IOCP worked

With MVSCP and IOCP you were limited to defining one processor or operating system per input data set. This meant that you needed more than one data set when you used MVSCP or IOCP.

Figure 1 on page 2 illustrates the definition process using several sources for writing and modifying the hardware configuration using IOCP and MVSCP data sets.
What HCD does

The configuration you define with HCD may consist of multiple processors with multiple channel subsystems, each containing multiple partitions. HCD stores the entire configuration data in a central repository, the *input/output definition file (IODF)*. The IODF as single source for all hardware and software definitions for a multi-processor system eliminates the need to maintain several independent MVSCP or IOCP data sets. That means that you enter the information only once using an interactive dialog.

Figure 2 on page 3 illustrates the definition process using one source for writing and modifying configuration data in the IODF.
Validation of configuration data

How MVSCP and IOCP worked
MVSCP and IOCP were separately running independent programs. Prior to IPL it was not checked whether the MVSCP output matched the configuration in the I/O configuration data set (IOCDS). Even if the definitions of both programs were not identical, it was possible for an IPL to be successful if the devices needed to start the system were included in both programs. Therefore, discrepancies would be detected only after the system had been running for some time. Such a discovery could have happened at a very inconvenient moment.

What HCD does
The data entered with HCD is validated and checked for consistency and completeness. Because the check is performed when the data is defined, rather than when the device is accessed, inconsistencies can be corrected right away, and unplanned system outages resulting from inconsistent definitions can be avoided.

If you include ESCON or FICON Director definitions in the IODF, HCD also validates the switch port usage and connection information for all devices and channels connected to each of the directors in the configuration. In addition, HCD validates the complete path from the processor through the switch to the control unit and device.

Planned I/O paths, as defined in an IODF, can be checked against the active configuration on the system. Discrepancies are indicated and can be evaluated before and after the IODF is activated.
Reconfiguration of configuration data

How MVSCP and IOCP worked

The IOCP updated the input/output configuration data set (IOCDS) that resided in the hardware support processor. This information was loaded into the hardware system area during power-on reset (POR). If the configuration was changed, it was necessary to write a new IOCDS using IOCP and to load it into the hardware system area with a POR.

MVSCP created the control information (such as UCBs, EDTs, and NIPCONs) needed by MVS to describe the hardware configuration and stored this information in the SYS1.NUCLEUS data set. The nucleus information was loaded at IPL time into storage. If a change was made to the I/O configuration, it was necessary to IPL to make the information available to MVS.

What HCD does

Dynamic reconfiguration management is the ability to select a new I/O configuration during normal processing and without the need to perform a power on reset (POR) of the hardware or an initial program load (IPL) of the z/OS operating system.

The ability of HCD to provide equivalent hardware and software I/O definitions and to detect when they are not in sync is essential for dynamic I/O reconfiguration management. HCD compares both the old and the new configuration and informs the hardware and software about the differences. You may add, delete, and modify definitions for channel paths, control units and I/O devices without having to perform a POR or an IPL.

What HCD offers you

This section summarizes what you can do with HCD and how you can work with HCD.

Single Point of Control: With HCD you have a single source, the IODF, for your configuration data. This means that hardware and software definitions as well as ESCON or FICON director definitions can be done from HCD and can be activated with the data stored in the IODF.

Increased System Availability: HCD checks the configuration data when it is entered and therefore reduces the chance of unplanned system outages due to inconsistent definitions.

Changing Hardware Definitions Dynamically: HCD offers dynamic I/O reconfiguration management. This function allows you to change your hardware and software definitions on the fly - you can add devices, or change devices, channel paths, and control units, without performing a POR or an IPL. You may also perform software-only changes, even if the hardware is not installed.

Sysplex Wide Activate: HCD offers you a single point of control for systems in a sysplex. You can dynamically activate the hardware and software configuration changes for systems defined in a sysplex.

Migration Support: HCD offers a migration function that allows you to migrate your current configuration data from IOCP, MVSCP, and HCPRIO data sets into HCD. Migration support also allows you to make bulk changes to the configuration using an editor on the IOCP/MVSCP/HCPRIO macro statements.
Accurate Configuration Documentation: The actual configuration definitions for one or more processors in the IODF are the basis for the reports you can produce with HCD. This means that the reports are accurate and reflect the up-to-date definition of your configuration.

HCD provides a number of textual reports and graphical reports, that can be either printed or displayed. The printed output can be used for documentation purposes providing the base for further configuration planning tasks. The display function allows you to get a quick overview of your logical hardware configuration.

Guidance through Interactive Interface: HCD provides an interactive user interface, based on ISPF, that supports both the hardware and the software configuration definition functions. The primary way of defining the configuration is through the ISPF dialog. HCD consists of a series of panels that guide you through all aspects of the configuration task. The configuration data is presented in lists.

HCD offers extensive online help and prompting facilities. Help includes information about panels, commands, data displayed, available actions, and context-sensitive help for input fields. A fast path for experienced users is also supported.

Batch Utilities: In addition to the interactive interface, HCD also offers a number of batch utilities. You can use these utilities, for instance, to migrate your existing configuration data; to maintain the IODF; or to print configuration reports. For a complete list of batch utility functions, refer to Chapter 12, “How to invoke HCD batch utility functions,” on page 307.

Cross Operating System Support: HCD allows you to define both MVS type (for example OS/390 or z/OS) and VM type configurations from z/OS and to exchange IODFs between z/OS HCD and z/VM HCD.

Support of S/390 Microprocessor Clusters: HCD provides functions for IOCDS and IPL attributes management, which simplify the configuration and operational support for those processors that are configured in an S/390 microprocessor cluster.

LDAP Interface Capability: HCD provides you with search and update capabilities for IODF data via an LDAP interface.

How HCD works

HCD stores the hardware configuration data you defined in the IODF. A single IODF can contain definitions for several processors (or LPARs) and several MVS or VM operating systems. It contains all information used to create IOCDSs and the information necessary to build the UCBs and EDTs. When HCD initiates the function to build the IOCDS, the IODF is used as input. The IOCDS with the channel subsystem definitions of a processor is then used to perform POR. The same IODF is used by MVS to read the configuration information directly from the IODF during IPL. If your environment includes z/OS and z/VM on different processors or as logical partitions on the same processor, the IODF can also be used to document the z/VM configuration.

Figure 3 on page 6 shows an example of a configuration using HCD.
Objects managed in the IODF

HCD lets you define the configurations as objects and their connections. The following objects and their connections are managed by HCD:

For all these objects the HCD dialog provides action lists where you can define the characteristics and the relation between the objects.

IODF used at IPL

After you complete the input of your configuration data, you have to build a production IODF. The production IODF is used by the operating system to build
the configuration data (for example, control blocks) at IPL time. This active production IODF is also used for building the IOCDS.

Figure 5 illustrates the build phase of a production IODF and of an IOCDS.

![Diagram of building a production IODF and an IOCDS]

The production IODF cannot be updated (read-only). This ensures that the data in the production IODF used at IPL remains the same during the run time of that system.

**Relationship of data sets used by HCD**

Figure 6 on page 8 shows the relationship between the data sets used by HCD and how you can work with or change these data sets.

By using the define and modify tasks or by migrating MVSCP, IOCP, and HCPRIO input data sets you create a work IODF. After finishing the definition, you build a production IODF from your work IODF, which you can use to IPL your system or to activate your configuration dynamically. The data sets shown at the bottom of the figure are created by the appropriate tasks like Build IOCDS, Build IOCP input data set, Build HCPRIO input data set, and Activate switch configuration.
HCD is part of z/OS. It needs a running z/OS system before it can be used to define a hardware configuration. Therefore, an installation should first load a z/OS system, using an old IODF, or a ServerPac Starter IODF to IPL the z/OS system for the first time. HCD can then be used on that system to define the full configuration.

HCD uses the unit information modules (UIMs) of z/OS. UIMs contain device dependent information, such as parameters and features of devices. The UIMs

---

Figure 6. Relationship of data sets used by HCD
must be installed in the z/OS system before you use HCD to define a configuration. The UIMs are also used at IPL time to build the UCBs. That is why they have to be installed in SYS1.NUCLEUS at IPL time.

UIMs are provided for the IBM devices supported by z/OS, OS/390, or MVS by the device product owners. You can write your own UIMs for non-IBM devices. To get information about UIMs, see \[z/OS MVS Device Validation Support\].

**HCD and I/O Operations**

The ESCON and FICON architectures are supported by a class of devices called Directors (switches) that manage the switching functions. These switches may be connected to a processor (via channel path), a control unit, or another switch. The switches connect channel paths and control units only for the duration of an I/O operation. The “internal” configuration of a switch is called switch configuration.

The I/O Operations component of System Automation for z/OS (formerly known as ESCON Manager) maintains and manages switch configurations.

I/O Operations functions, such as the activation of switch configurations and the retrieval of the active configuration data, can be invoked from HCD. This gives you a single point of control for all switch configuration activities, as well as the possibility to check whether a certain data path (from processor to device) is fully configured or not.

Moreover, you can use HCD to migrate switch configurations into HCD from three different sources: directly from the switches, from a saved switch file, or from ISPF tables saved by I/O Operations. You also can save switch configuration data in a switch file, as well as activate the switch configuration. Thus, the switch is activated using the switch configuration stored in the IODF.

I/O Operations permits retrieval of information from the active system. Parts of this information (serial numbers, VOLSERs, port names) can be provided by HCD via the functions “Priming”, “Sensing”, “Activate Verification” and “Prompting for VOLSER”; I/O Operations must be installed for this purpose, even if there are no switches in the configuration.

**Prerequisites**

For migrating existing switch configurations and activating switch configurations:

- System Automation for z/OS (I/O Operations), must be installed and active.
- If you want to prime or verify I/O definitions or generate an I/O Path report, then in order to obtain the data for a system in a sysplex, there must also be a VTAM session between the local system and the target system, and the target system must have I/O Operations installed and running.

For more information on installation requirements, refer to the \[z/OS Program Directory\].

In addition, refer to the PSP Bucket for the latest information about the prerequisites.

**HCD and the coupling facility**

HCD provides the user interface to support processors that have coupling facility capability.
The coupling facility itself is implemented as an extension to PR/SM features on selected processors (refer to z/OS HCD Planning for a list) and runs in a PR/SM partition. It enables direct communication between processors through a specific communication partition (coupling facility partition), connected by coupling facility channels.

You use HCD to specify whether a logical partition is running a coupling facility or an operating system. New channel path definitions in the IODF are used to connect a coupling facility-capable processor to a coupling facility partition:

- The coupling facility receiver channel (CFR channel) path that accesses the partition the coupling facility is running on.
- The coupling facility sender channel (CFS channel) path that accesses the partition the operating system is running on.
- The coupling facility peer channel path that accesses either partition bidirectionally on IBM zSeries processors or their successors.

HCD automatically generates the coupling facility (CF) control unit and devices that are necessary for the IOP. Figure 7 shows a processor configuration with coupling facility implemented.

HCD enables you to dynamically reconfigure the coupling facility channels that are connected to the operating system partition.

**Note:** With CF duplexing, a CF logical partition can use the coupling facility sender function to communicate with another CF logical partition. That means, you can define sender channel paths (CFS, CBS, ICS) besides the receiver channel paths (CFR, CBR, ICR) in a CF partition. CF duplexing is supported starting with the 9672 Parallel Enterprise Servers G5 and G6 models and the 2064 zSeries models.
Support of S/390 microprocessor clusters

Note on terminology:

The term S/390 microprocessor cluster, used in the HCD panels and throughout this book, refers to central processor complexes (CPCs) controlled through the Hardware Management Console.

HCD allows you to define and control configuration data for each CPC that is configured in an S/390 microprocessor cluster. You use HCD for those CPCs that can have their IOCDS and IPL attribute management functions controlled remotely:

- Writing IOCDSs
- Managing write-protection
- Marking the IOCDS as active POR IOCDS
- Updating IPL address and LOAD parameter values

HCD displays all CPCs that are configured in an S/390 microprocessor cluster and controlled by the Hardware Management Console. The CPCs are connected to the same network as the Hardware Management Console through a Token-Ring LAN. A CPC is identified by the system network architecture (SNA) address of its support element, which is specified when the processor complex is configured on the Hardware Management Console. HCD uses the SNA address to be able to write and manage IOCDSs from any processor in an S/390 microprocessor cluster, and to view and update IPL attributes.

Support of the sysplex environment

If you have interconnected systems, it is important to have a single point of control for systems in such a sysplex environment. HCD offers the support of the sysplex environment in several ways:

- You can define all processor and operating system configurations in one IODF.
- After a complete definition you can download the IOCDSs for all processors. This can be initiated from the controlling HCD.
- If you want to change configurations dynamically within the sysplex, you can initiate the activation of the hardware and software configuration changes for systems defined in a sysplex from the controlling HCD.

In addition, HCD offers a function that compares the active I/O configuration against the defined configuration in an IODF. This function can produce reports or lists of I/O paths to show the sensed data against the logical definitions of the paths in the IODF. HCD provides this function to get the data for your local system and for systems in a sysplex.

Based on a production IODF, HCD can also build CONFIGxx members for your local system or for systems in the sysplex.

For systems which are members of a sysplex, CONFIGxx members can be verified against selected systems. Responses are displayed in a message list.
Refer to Chapter 12, “How to invoke HCD batch utility functions,” on page 307, if you want to run batch jobs in a sysplex environment.
Chapter 2. Migration

Overview

This topic explains:

- An overview of the migration process
- Steps needed for migrating to HCD for z/OS Version 1 from a lower level HCD release

Migration overview

Your plan for migrating to the new level of HCD should include information from a variety of sources. These sources of information describe topics such as coexistence, service, hardware and software requirements, installation and migration procedures, and interface changes.

The following documentation provides information about installing your z/OS system. In addition to specific information about HCD, this documentation contains information about all of the z/OS elements.

- **z/OS Planning for Installation**
  This book describes the installation requirements for z/OS at a system and element level. It includes hardware, software, and service requirements for both the driving and target systems. It also describes any coexistence considerations and actions.

- **z/OS Program Directory**
  This document, which is provided with your z/OS product order, leads you through the specific installation steps for HCD and the other z/OS elements.

- **ServerPac Installing Your Order**
  This is the order-customized, installation book for using the ServerPac Installation method. Be sure to review “Appendix A. Product Information”, which describes data sets supplied, jobs or procedures that have been completed for you, and product status. IBM may have run jobs or made updates to PARMLIB or other system control data sets. These updates could affect your migration.

  “Migration roadmap” on page 14

- identifies the migration paths that are supported with the current level of HCD

- describes the additional publications that can assist you with your migration to the current level

Developing a migration strategy

The recommended steps for migrating to HCD under z/OS are:

1. Become familiar with the supporting migration and installation documentation for the release.
   You should determine what updates are needed for products that are supplied by IBM, system libraries, and non-IBM products. Review **z/OS Planning for Installation** and the **z/OS Introduction and Release Guide** for information about HCD and other z/OS elements.

2. Develop a migration plan for your installation.
When planning to migrate to a new release of HCD, you must consider high-level support requirements, such as machine and programming restrictions, migration paths, and program compatibility.

3. Obtain and install any required program temporary fixes (PTFs) or updated versions of the operating system.

Call the IBM Software Support Center to obtain the preventive service planning (PSP) upgrade for HCD, which provides the most current information about PTFs for HCD. Check RETAIN again just before testing HCD. For information about how to request the PSP upgrade, refer to the z/OS Program Directory. Although the z/OS Program Directory contains a list of the required PTFs, the most current information is available from the IBM Software Support Center.

4. Install the product using the z/OS Program Directory or the ServerPac Installing Your Order documentation.

5. Contact programmers who are responsible for updating applications at your installation.

Verify that your installation’s applications will continue to run, and, if necessary, make changes to ensure compatibility with the new release.

6. Use the new release before initializing major new function.

7. If necessary, customize the new function for your installation.

8. Exercise the new functions.

For additional migration information

For information about migrating to z/OS, see z/OS Planning for Installation GA22-7504.

For information about migrating to MVS/ESA SP V5, see:

- MVS/ESA SP V5 Planning: Installation and Migration JES2, GC28-1428.
- Conversion Notebook for System Product Version 5, GC28-1436
- Conversion Notebook for System Product Version 4, GC28-1608
- Conversion Notebook for System Product Version 3, GC28-1568
- Conversion Notebook for System Product Version 2, GC28-1567

Migration roadmap

This section describes the migration paths that are supported by the current release of HCD. It also provides information about how to migrate to the current HCD release from previous releases.

You can find further migration information in z/OS Migration.

Migration tasks

The following sections contain additional migration procedures or information:

- “Upgrade an IODF” on page 48
- “IODF release level compatibility” on page 57
- “Migrating existing switch configurations” on page 189
- “IOCP input data sets using extended migration” on page 210
- Chapter 11, “How to migrate existing input data sets,” on page 261
- “Upgrade IODF” on page 310
- “Migrate I/O configuration statements” on page 311
Information on IODF coexistence can be found in “IODF release level compatibility” on page 57.

**Required changes when migrating to z/OS V1.7 HCD**

z/OS V1.7 HCD starts using the new V5 IODF format, representing devices in device groups rather than containing individual device definition records.

z/OS V1.7 HCD can access an IODF from previous HCD releases and can perform view/read functions (for example, copy, activate) without permanently upgrading the IODF. However, when you try to change the configuration contained in a lower version IODF, upgrading is required before the change. A message will inform you about the necessity of a permanent upgrade. “Upgrade an IODF” on page 48 provides information on how to upgrade from previous IODF versions to a V5 IODF.

Earlier HCD releases provide limited read access to V5 IODFs. There is a coexistence support for V5 IODFs on back-level z/OS releases for read-only functions like the ACTIVATE function. You need to install the corresponding PTF, and you must have z/OS V1.4 HCD (FMID HCS7708) running on your previous z/OS release.

If you need to update a V5 IODF using an HCD version earlier than z/OS V1.7 HCD, you can use a STEPLIB or JOBLIB allocation for the z/OS V1.7 libraries (SYS1.LINKLIB, SYS1.NUCLEUS and SYS1.SCBDHENU).

If you want to share an IODF among multiple z/OS or OS/390 systems that are at different release levels, you have to consider some restrictions concerning IPL, IODF usage, and dynamic reconfiguration. For more information on compatibility considerations for IODF data sets, refer to “IODF release level compatibility” on page 57.

**Upgrading an IODF to z/OS V1.7 HCD**

If you used HCD in a previous z/OS or OS/390 release, you can IPL the z/OS operating system with your old IODF but you must upgrade your back-level IODF if you want to update it with z/OS V1.7 HCD. You need not upgrade your back-level IODF if you want to access it read-only.

The following procedure is a step-by-step instruction on how to upgrade a back-level IODF to a V5 IODF using the HCD dialog:

1. Invoke HCD, specify the IODF name and select the task **Maintain I/O definition files** on the Primary Task Selection panel. HCD detects the back-level V4 IODF format and upgrades the IODF in-storage to a V5 IODF. Depending on the size of the IODF, this task may be time-consuming.

2. Select the task **Upgrade I/O definition file to new format**. Select the default options for Target of upgrade and Condense IODF.

3. On the following panel, HCD suggests an appropriate space allocation value. The proposed value is generally significantly smaller than the size of the source IODF.

4. After the IODF has been successfully upgraded, you can enter additional configuration changes into the V5 work IODF.

5. Create a production IODF from the work IODF and a corresponding LOADxx member.

6. Do one of the following:
   - Activate the z/OS production IODF dynamically.
Note: Depending on the changes that you made in step 4 on page 15 before activating the new configuration, you have to configure offline the affected channel paths or vary offline the affected devices. See z/OS HCD Planning for information about avoiding disruptions to I/O operations during dynamic changes.

- Use the z/OS production IODF to create an IOCDS, then perform a power-on reset with the new IOCDS and IPL using the new IODF.

Setup considerations for migration
If you want to use HCD functions to prepare a V5 IODF, z/OS V1.7 HCD must be available on your back-level operating system. You can either install z/OS V1.7 HCD on your back level z/OS system libraries, or you can share the DASD volume where the z/OS V1.7 system libraries reside. Note that when sharing the DASD volume, the HCD libraries are not cataloged in your running system. You have to access the libraries by specifying the volume and unit parameter.

If you want to prepare a V5 IODF using the HCD batch utility:
- Ensure that the products containing the necessary UIMs are installed.
- Access the SYS1.LINKLIB of the V1.7 z/OS operating system through the STEPLIB or JOBLIB statement.
- Access the parsing macro in SYS1.MACLIB of z/OS V1.7 through the HCDLIB statement.
- For IOCP input data set migration, ensure you know the support level ID of your processor. Use the batch utility to print the Supported Hardware report for assistance when checking the support level ID. An example is shown in “Supported Hardware Report” on page 401.
- Ensure that SYS1.NUCLEUS of z/OS V1.7 is used. Specify the following statement in your batch jobs:
  ```
  //HCDPROF DD DSN=BPAN.HCD.PROF,DISP=SHR
  ```

In the HCD profile (in our example BPAN.HCD.PROF) you have to specify the following statements to access the z/OS UIMs:

Example:

```
UIM_LIBNAME=SYS1.NUCLEUS
UIM_VOLSER=510D18
```

If you want to prepare a V5 IODF using the HCD dialog:
- Ensure that the products containing the necessary UIMs are installed.
- Specify the library containing the UIMs in the z/OS V1.7 HCD profile.

Example:

```
UIM_LIBNAME=SYS1.NUCLEUS
UIM_VOLSER=510D18
```

- Access SYS1.LINKLIB and SYS1.SCBDHENU of the z/OS V1.7 operating system using a TSO logon procedure that contains the following DD statements to allocate the load libraries.

Example:

```
//ISPLLIB DD DSN=SYS1.LINKLIB,DISP=SHR,UNIT=3380,VOL=SER=510D18
// DD DSN=SYS1.SCBDHENU,DISP=SHR,UNIT=3380,VOL=SER=510D18
```

With this TSO logon procedure you can use your TSO session for HCD only. Before using another program than HCD, start a TSO session without the HCD DD statements.
If your target LINKLIB contains an IDCAMS level incompatible with the driver system, HCD may terminate with the error message IDC3009I. In this case, create a new library and copy the modules starting with CBD from the target system into this library. Then, change the ISPLLIB to point to this new library.

- For the migration of the input data sets, specify the volume serial number of the DASD, which contains the MACLIB, on the panel Migrate IOCP / MVSCP / HCPRIIO Data to access the parsing macro.

- Copy the HCD CLISTS CBDCCHC1, CBDCCHCD1, and CBDCACTL from SYS1.SCBDCCLST of the z/OS V1.7 system with a new name (for example, HCDCHC1 and CDCCHCD1) to SYS1.SCBDCCLST of your old system. Note that you also have to replace the old name of CBDCCHCD1 in the new HCD CLIST (in our example, CDCCHCD1) with the new name (HCDCHCD1). In the new CLIST (HCDCHCD), change the allocation of the Panel, Table, and Message library:

  /* Allocate Panel library */
  ALLOCATE F(PLIB) DA('SYS1.SCBDPENU') SHR +
  UNIT(3380) VOLUME(510D18)
  ISPEXEC LIBDEF ISPPLIB LIBRARY ID(PLIB)

  /* Allocate Table library */
  ALLOCATE F(TLIB) DA('SYS1.SCBDTENU') SHR +
  UNIT(3380) VOLUME(510D18)
  ISPEXEC LIBDEF ISPPLIB LIBRARY ID(TLIB)

  /* Allocate Message library */
  ALLOCATE F(MLIB) DA('SYS1.SCBDMENU') SHR +
  UNIT(3380) VOLUME(510D18)
  ISPEXEC LIBDEF ISPPLIB LIBRARY ID(MLIB)

Use this changed CLIST to start the HCD dialog.

**Note:** If you want to use the HCD utility functions
- Print report
- Build IOCP
- Build IOCDS
- Compare IODFs
- Compare CSS / operating system views,

then specify GO.STEPLIB in the offered EXEC procedures (see "Customizing HCD EXEC procedures" on page 34) to access the new data sets:

//GO.STEPLIB DD DSN=SYS1.LINKLIB,DISP=SHR,VOL=SER=510D18,UNIT=3380
//GO.HCDPROF DD DSN=BPA.N.HCD.PROF,DISP=SHR

In the HCD profile (in our example BPAN.HCD.PROF) you have to specify the following statements to access the z/OS V1.7 UIMs.

Example:

UIM_LIBNAME=SYS1.NUCLEUS
UIM_VOLSER=510D18

If you want to migrate IODEVICE configuration statements into an IODF, you might need temporarily more disk space than the final IODF consumes, because HCD can migrate one device at a time only and perform grouping only at the end of the migration.

**Coexistence overview of functionality based on IODF version**

Table 2 on page 18 summarizes the available functionality on z/OS releases based on the IODF level.
### Table 2. IODF compatibility overview

<table>
<thead>
<tr>
<th>IODF created</th>
<th>System running z/OS R7 or later</th>
<th>System running z/OS R4 to R6</th>
<th>System running z/OS pre-R4</th>
</tr>
</thead>
</table>
| with z/OS V1R7 or later releases (Version 5 IODF) | • No restriction to HCD use (read/update)  
• IPL  
• Dynamic Activate | Compatibility SPE installed:  
• Only read / no update in HCD  
**Note:** View Configuration Data (part of option 1 from the HCD Primary Task Selection panel) is not supported.  
• IPL  
• Dynamic Activate | • No access possible in HCD  
• No IPL  
• No Dynamic Activate |
| IODF created  | Read access possible w/o upgrade  
• Update access requires IODF upgrade  
• IPL  
• Dynamic Activate | • No restriction to HCD use (read/update)  
• IPL  
• Dynamic Activate | • No restriction to HCD use (read/update)  
• IPL  
• Dynamic Activate |
| with z/OS pre-V1R7 (Version 4 IODF) | | Compatibility SPE not installed:  
• No access possible in HCD  
• No IPL  
• No Dynamic Activate  
STEPLIB/JOBLIB approach using z/OS V1R7 HCD libraries for full HCD access | |

**IODF format conversions between version 4 and version 5:**
- z/OS V1R7: IODF upgrade function available to migrate from version 4 to version 5 IODF
- z/OS V1R4 - V1R6: Fall-back solution to downgrade from version 5 IODF to version 4 IODF via Export I/O definitions (Build I/O configuration data) and Import I/O definitions (Migrate I/O configuration statements)
Chapter 3. How to set up, customize and start HCD

Overview

This information unit handles the following topics:

- “Setting up HCD”
- “Tailoring the CLIST CBDCCHCD” on page 21
- “Starting and ending HCD” on page 22
- “Defining an HCD profile” on page 23
- “Customizing HCD EXEC procedures” on page 34

Setting up HCD

HCD is a base element of z/OS and therefore installed with the z/OS product. For more information, refer to z/OS Planning for Installation.

The installation of HCD is carried out using SMP/E. The install logic and the JCLIN are provided by HCD. For the installation, refer to the z/OS Program Directory.

Setting up HCD requires the following steps:

1. Install z/OS with the HCD FMIDs.

2. Install other products that are required for HCD (refer to z/OS Planning for Installation).

3. Before you start HCD, you have to set up the load libraries that contain the HCD help modules. You can achieve this in one of the following ways:
   - Include SYS1.SCBDHENU (or SYS1.SCBDHJPN for Kanji) in the linklist concatenation (LNKLSTxx member), or
   - Allocate data set SYS1.SCBDHENU (or SYS1.SCBDHJPN for Kanji) to ISPLLIB.

   If you choose to access the libraries through the ISPLLIB concatenation, ISPLLIB must be allocated prior to invoking ISPF with the TSO ALLOC command or through a CLIST. ISPLLIB is used as a tasklib by ISPF as it is searched first.

   When using the View graphical configuration report:
   - Include the GDDM load library in the linklist concatenation (LNKLSTxx member).

   - Allocate the GDDM sample data set:
     For GDDM 2.1 and 2.2:
     ALLOCate F(ADMPM) DSN('pplib.GDDM.GDDMSAM') SHR REUSE
     For GDDM 2.3 or later:
     ALLOCate F(ADMPM) DSN('pplib.GDDM.SGDDMSAM') SHR REUSE

   If you are using a programmable workstation and you communicate with the host using a 3270 emulator session, the GDDM-OS/2 link files must be installed on your workstation. Note that the high-level qualifier for the GDDM data set might vary from installation to installation.
4. Allocate the data set SYS1.SCBDCLST to the SYSPROC ddname concatenation.

   **Note:** SYS1.SCBDCLST has a fixed record format (RECFM=FB). If your other
   SYSPROC data sets have a variable record format (RECFM=V or VB),
   copy SYS1.SCBDCLST to a data set with variable record format. You
   have to remove sequence numbers (in the CLIST) after copying the
   members to a data set with variable record format.

5. For processing of large IODFs, the size of your TSO region may not be
   sufficient. When you specify the region size on the TSO logon panel, calculate
   as follows:

   $2 \times \text{IODF size} + 4 \text{MB}$

   **Example:**

   Assumed IODF size: 8000 blocks, 4 KB per block = 32 MB

   Suggested region size: 68 MB

6. For setup requirements when using HCD via the HCD LDAP Backend, see
   Chapter 14, “How to provide LDAP support for HCD,” on page 347.

To run HCD, the modules in SYS1.SCBDHENU (containing HCD help members)
and SYS1.NUCLEUS (containing the UIMs) must be accessible. For the HCD
dialog, you can achieve this in three ways:

1. Include SYS1.SCBDHENU in the linklist concatenation (LNKLSTxx member).
2. Include SYS1.SCBDHENU in the JOBLIB/STEPLIB concatenation of the TSO
   logon procedure.
3. Include SYS1.SCBDHENU in the ISPLLIB load library concatenation. If you
   include SYS1.SCBDHENU into the ISPLLIB concatenation, ISPLLIB must be
   allocated prior to invoking ISPF (in TSO or through JCL in the logon
   procedure). ISPLLIB is used as a tasklib by ISPF and is searched first. A pure
   LIBDEF for ISPLLIB does not suffice to invoke HCD.

HCD allocates SYS1.NUCLEUS automatically at initialization time if the keyword
UIM_LIBNAME is not specified in the HCD profile. You may use the HCD profile to
specify a different name and the volume serial number of the library that contains
the UIMs (see also “Defining an HCD profile” on page 23). If you do not specify a
name in the profile, SYS1.NUCLEUS is assumed as default name for the UIMs. For
IPL, however, the UIMs and UDTs must be in SYS1.NUCLEUS.

The following HCD libraries will get defined via the ISPF ‘ISPEXEC LIBDEF’
command if HCD is invoked via CLISTS CBDCHCD and CBDCHCD1:

- SYS1.SCBDPENU for panels
- SYS1.SCBDMENU for messages
- SYS1.SCBDTENU for tables

### How to invoke HCD in dialog mode

To invoke HCD in dialog mode, ISPF must be active. After you have invoked ISPF,
you can use the CLIST CBDCDHCD to activate the HCD function. You may add
HCD to an ISPF selection menu, for example, the ISPF/PDF Master Application
Menu (ISP@MSTR), and invoke HCD using the CLIST CBDCDHCD. Figure 8 on
page 21 shows a sample panel that illustrates how to include HCD on the main
ISPF/PDF panel. Alternatively, the CLIST can be invoked from the ISPF option 6,
or from the command line.
Note that HCD must be invoked with the “NEWAPPL(CBD)” parameter in the CLIST CBDCHCD.

To ensure that the CBDCHCD CLIST can successfully allocate the following libraries, make sure that these libraries are cataloged:

- SYS1.SCBDPENU --> HCD Panel Library
- SYS1.SCBDMENU --> HCD Message Library
- SYS1.SCBDTENU --> HCD Table Library

**Note:** The HCD Panel, Message, and Table libraries are allocated by the CBDCHCD CLIST using the LIBDEF function of ISPF. If other ISPF Dialogs are using the LIBDEF function of ISPF, and you do not want HCD to overlay their allocations, you can update your ISPF startup by adding the HCD data sets to the ISPF ISPPLIB, ISPMLIB, and ISPTLIB concatenations.

---

**Tailoring the CLIST CBDCHCD**

A sample CLIST CBDCHCD (CBDCHCDJ for Kanji) is provided in SYS1.SCBDCLST to assist you in invoking HCD from the ISPF dialog. It allocates the HCD message log file (HCDMLOG), the trace data set (HCDTRACE), and the HCD term file (HCDTERM) with a default high-level qualifier of the userID (&SYSUID). The CLIST also allocates the HCD data sets SYS1.SCBDPENU, SYS1.SCBDMENU, and SYS1.SCBDTENU (or SYS1.SCBDPJPN, SYS1.SCBDMJPN, and SYS1.SCBDTJPN for Kanji).

In CBDCHCD and CBDCHCDJ, the ISPEXEC LIBDEF statement for ISPPLIB, ISPTLIB and ISPMLIB is done with the STACK option. This leaves existing LIBDEFs untouched, so that after exit, the existing HCD libraries ISPPLIB, ISPTLIB and ISPMLIB are freed again.

---

Figure 8. Sample ISPF Master Application Menu
If the prefixes for message, trace, and term data sets do not conform to the installation conventions, you may tailor the CLIST to match your installation defaults. If you want HCD to use your TSO prefix as the high level qualifier, you can call CBDCHCD with the parameter NOPREF(YES). This causes HCD to use the qualifiers &PREFIX..&SYSUID.

CBDCHCD invokes another CLIST, CBDCHCD1. You may tailor this CLIST as well.

In any case, use the application ID for HCD: NEWAPPL(CBD).

The CLIST also tailors the ISPF environment by:

- Setting PFSHOW on. This forces all 24 function keys to be shown (if ISPF is defined to show 24 function keys).
- Setting lower PFKEYS as primary function keys.

Note: HCD can be invoked with the activated TRACE option, when you have specified parameter TRACE(YES) in the default CLIST. The size of the trace data set can be changed by modifying the CLIST. In addition, you can delete the HCD provided trace data set and allocate one according to your specific needs.

Starting and ending HCD

You start HCD like any other ISPF application in your TSO/E system. The procedure for starting an application is different for each installation but you can probably select HCD from a menu of applications that are available in your system. This causes a TSO/E CLIST to be executed. The sample CLIST that is supplied with HCD is CBDCHCD in library SYS1.SCBDCLST.

After you start HCD, the first panel that you see contains a menu of the HCD primary tasks:
If you select task "1. Define, modify, or view configuration data" as shown in Figure 9 and press the Enter key, you trigger this task using the IODF 'SYS1.IODF00.HCD.WORK'.

To end an HCD session, either return to the Primary Task Selection panel and press the F12=Cancel key or the F3=Exit key twice or use the fast path command GOTO X.

Notes:
1. Chapter 12, “How to invoke HCD batch utility functions,” on page 307 explains how you can invoke HCD from another program using the HCD programming interface.
2. To see a description of new functionality of the current release, select option 9. What's new in this release. Here you may find information about SPEs that are delivered after the completion of this document.

Defining an HCD profile

Before you start HCD, you can define an HCD profile to tailor HCD supplied defaults and processing options to your specific installation needs. Using a profile is optional and will remain in effect for the entire HCD session. At initialization time, HCD reads the profile and processes each statement in turn.

The profile statements are contained in a data set allocated to the DD name HCDPROF. The following statement allocates the profile data set to HCDPROF:

```
//HCDPROF DD DSN=&SYSUID..HCD.PROFILE,DISP=SHR
```

The data set must have the following characteristics:
- Be either a sequential data set or a member of a partitioned data set
- Have fixed-length, fixed-blocked record format
- Have 80 character records.
You can use the **HCD Profile Options** dialog (see “Working with the HCD Profile Options dialog”) to edit the profile keywords after having created and allocated the profile data set to HCDPROF.

In addition, you can define profile options manually in the profile data set.

You can extend a comment to the next line by using an asterisk (*) as a continuation character in column 1, as shown in the example:

```
MAP_CUTCUTYPE=9000,NOCHECK /* map CU type 9000 to type NOCHECK */
```

or:

```
MAP_CUTYPE=9000,NOCHECK /* map CU type 9000 to type
* NOCHECK */
```

However, be aware that the **HCD Profile Options** dialog truncates comments longer than 32 characters.

It is also possible to define profile options using inline statements in a batch job. The following example shows an inline profile definition:

```
//HCDPROF DD *
MIGRATE_EXTENDED = YES
VM_UIM = NO
/*
```

**Note:** When starting an HCD batch job from the dialog, the HCD profile data set is not passed automatically to the job but, if required, has to be specified in the JCL.

The following syntax rules apply to a profile statement:

```
<keyword>_<value>
```

**keyword**
- is the name of the HCD keyword; each keyword starts on a new line.

**value**
- specifies one or more values to be assigned to the keyword.

**Notes:**
1. A single statement must not exceed 72 characters.
2. Do not use sequence numbers in your HCD profile.
3. Use /* and */ as delimiters for comments in a profile statement.

**Working with the HCD Profile Options dialog**

You can maintain your HCD profile values in your profile data set using the HCD dialog. Select **Edit profile options** from the HCD Primary Task Selection panel to invoke the **HCD Profile Options** dialog (Figure 10 on page 25).

You can add, delete or modify keywords in your profile data set via this dialog. When you leave the HCD session, HCD writes all changes specified in this dialog to the profile data set.
In this dialog, all HCD profile keywords are listed in alphabetical order, followed to the right by their value and possibly a description (user comment). To see the Description column, scroll to the right (Shift + PF8).

- If a profile data set exists, HCD reads the contained keywords with their values and, if available, their descriptions. For keywords that are not explicitly defined in your profile data set, HCD shows the defaults.
- If you have no own profile data set allocated, this dialog lists the HCD default values. You cannot change these settings.

**Note:** The remainder of this section describes the use of the HCD Profile Options dialog with an allocated profile data set.

---

<table>
<thead>
<tr>
<th>Command ===&gt;</th>
<th>Scroll ===&gt; CSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCD Profile : DOCU.HCD.PROFILE</td>
<td></td>
</tr>
<tr>
<td>/ Profile keyword</td>
<td>A Value +</td>
</tr>
<tr>
<td># ACTLOG_VOL</td>
<td>Y *</td>
</tr>
<tr>
<td># ALLOC_SPACE</td>
<td>Y HCDASMP,60</td>
</tr>
<tr>
<td># ALLOC_SPACE</td>
<td>Y HCDRPT,60</td>
</tr>
<tr>
<td># BATCH IODF NAME CHECK</td>
<td>Y NO</td>
</tr>
<tr>
<td># BYPASS_UPD_IODF_FOR_SNA</td>
<td>Y YES</td>
</tr>
<tr>
<td># CHANGE LOG</td>
<td>N NO</td>
</tr>
<tr>
<td># CHECK_IODF</td>
<td>Y NO</td>
</tr>
<tr>
<td># CHLOG_EXTENSION</td>
<td>Y O</td>
</tr>
<tr>
<td># CHLOG_VOL</td>
<td>Y *</td>
</tr>
<tr>
<td># COLOR_BACKGROUND</td>
<td>Y</td>
</tr>
<tr>
<td># COLOR_HIGH</td>
<td>Y RED</td>
</tr>
</tbody>
</table>

F1=Help  F2=Split  F3=Exit  F4=Prompt  F5=Reset  F7=Backward  F8=Forward  F9=Swap  F12=Cancel  F20=Right  F22=Command

---

**Figure 10. HCD Profile Options**

If you change a displayed default value, HCD writes the changed entry into your data set.

The **HCD Profile Options** dialog accepts a description (comment) of maximum 32 characters. A comment must start on the same line as the keyword assignment, using the ‘/’ notation.

If you specified a longer comment in a manually edited profile data set, HCD shows the truncated comment only and also truncates the text in the profile data set the next time, HCD writes any modifications from the dialog.

For profile keywords which may occur multiple times with different value assignments, the dialog offers actions to add or delete selected entries (action codes ‘a’ or ‘d’). Profile keywords which can only occur with a single value assignment are disabled for any action by a # sign in the action column.

**Column A:** This column is set to Y(es), if a change of the keyword value will become active immediately. Value N(o) denotes that the value change does not become active until the next start of HCD.

**Column Value:** You can overtype the current values. Also, if you position the cursor on a value in this column, you have the following options:
• Pressing PF4 lets you prompt for available values (PF4) where applicable, and
select one value from the offered list.
• By pressing PF1, you can obtain an explanation of the selected keyword.

Figure 11 on page 33 shows the contents of a sample profile data set.

Keywords
The following keywords are supported for a profile statement:

**Volume serial number to allocate output data sets**
To place the IOCP, HCPRIO, and JES3 INISH stream checker data sets to a specific
volume within HCD, you can specify this target volume via two profile options:

*HCDDECK_VOL*
Specifies the volume serial number for allocating a new IOCP, HCPRIO or
other data set containing I/O configuration statements.

*HCDJES3_VOL*
Specifies the volume serial number for allocating a new JES3 initialization
stream checker input data set.

In a non-SMS managed environment, the generated corresponding output data set
is placed on the indicated volume. The profile option is ignored if the specified
output data set already exists on a different volume. In this case, the new output
data set replaces the existing data set on this volume. In an SMS-managed
environment this option is ignored.

The two options have no effect in an SMS managed environment since SMS
overrules the VOLUME parameter.

If the keywords are omitted, the placements of the IOCP, HCPRIO, JES3 INISH,
and other I/O configuration data sets are controlled via SMS or ESOTERIC system
defaults (ALLOCxx of SYS1.PARMLIB or the UADS, respectively).

**Volume serial numbers to allocate log data sets and HCM MCF
data set**

*ACTLOG_VOL*
If the dataset names are not managed by SMS, this keyword specifies the
up to 6 characters long volume serial number to allocate a new activity log.
Using an asterisk (*) indicates that the activity log file will be placed on the
same volume where the associated IODF resides.

*CHLOG_VOL*
If the dataset names are not managed by SMS, this keyword specifies the
up to 6 characters long volume serial number where to allocate the change
log data set. Using an asterisk (*) indicates that the change log file will be
placed on the same volume where the associated IODF resides.

*MCF_VOL*
If the data set names are not managed by SMS, this keyword specifies the
up to 6 characters long volume serial number where to allocate the MCF
data set. Using an asterisk (*) indicates that the MCF data set will be
placed on the same volume where the associated IODF resides.

**Automatic activity logging**

*CHANGE_LOG*
YES/NO. Specifies whether you want to activate change logging (YES).
The default is NO.
If enabled, and HCD additionally maintains an activity log file for the IODF, then HCD logs all updates applied to the IODF in a change log file, and automatically generates activity log entries for updates on HCD objects, for example, add, delete, update or connect, disconnect. These entries are proposals and are shown in the activity log panel where you can modify them before you exit the IODF (see also “Activity logging and change logging” on page 54).

A change of the value setting for this keyword will not be active until the next start of HCD.

**Allow or prohibit mixed esoterics**

*MIXED_ESOTERIC*

YES/NO. Specifies whether you want to allow or prohibit mixed devices (DASD and TAPE) under the same esoteric name. If you specify NO, which is the default, and your configuration contains an esoteric with mixed DASD and TAPE devices, the request to build a production IODF will fail with error message CBDA332I. If you specify YES, HCD issues message CBDA332I as warning message and continues the request.

**HLQ for exporting IODFs**

*EXPORTED_HLQ*

By default, when exporting an IODF, the generated sequential data set is written with the high-level qualifier (HLQ) of the userID that issued the Export IODF function. If this convention is not suitable for your installation, you can use the keyword EXPORTED_HLQ to specify a different HLQ (up to 8 characters).

**Allocation space for data sets allocated due to HCM requests**

*ALLOC_SPACE*

This HCD profile option lets you overwrite the default allocation (CYL,50,50) for data sets that are temporarily allocated in response to HCM requests, such as HCDASMP, HCDRPT, HCDIN. For example, specify:

```
ALLOC_SPACE = HCDASMP,nn
```

where

```
nn
```

is the size (decimal number) used for primary and secondary allocation (in CYL).

**Extending allocation space**

*MCF_EXTENSION*

This HCD profile option lets you define additional space when allocating the MCF data sets to allow for updates.

With this keyword, you specify the percentage of additional space that is to be allocated when defining an MCF data set. Per default an MCF data set is allocated with 30 percent additional space than actually needed to hold the MCF data. You can use this space for updates that consume data space without the need to allocate a new MCF and delete the old one. For example, MCF_EXTENSION = 50 allocates 50% additional space.

**CHLOG_EXTENSION**

This HCD profile option lets you define additional space when allocating the change log data set.

With this keyword, you specify the percentage of additional space that is to be allocated when defining a change log data set. By default, a change log data set is allocated with the same size as the associated IODF. For example, a value set to 50 allocates 50% additional space.

The default extension is 0.
Name and volume serial number for UIM library

**UIM_LIBNAME**
Specifies the name of the data set containing the UIMs, the associated UDTs, and any help members for the UIMs. If the keyword is omitted, SYS1.NUCLEUS is assumed (Note: only UIMs residing in SYS1.NUCLEUS are read during IPL!).

When UIM_LIBNAME is specified, HCD does not implicitly access SYS1.NUCLEUS for loading the UIMs.

If you specify an asterisk (*) as data set name, HCD assumes that the UIM data set (including SYS1.NUCLEUS) is part of the ISPF load library concatenation chain, contained in the JOBLIB/STEPLIB concatenation chain, or specified in the active LNKLSTxx member.

You can only define one data set with the UIM_LIBNAME statement. If you want to specify several data sets, specify an asterisk (*) as data set name and specify the data sets in the JOBLIB/STEPLIB concatenation chain.

**UIM_VOLSER**
Specifies the volume serial number of the UIM library. Required only if the data set is specified via keyword UIM_LIBNAME and is uncataloged.

Load VM UIMs

**VM_UIM**
YES/NO. Specifies whether VM UIMs will be loaded. The default is YES. Installations without VM should specify NO to gain some performance improvement during HCD initialization.

Options for text reports

**LINES_PER_REPORT_PAGE**
Specifies the maximum number of lines per page for reports. The default value is 55.

**UPPERCASE_ONLY**
YES/NO. Specifies whether all HCD reports will be written in uppercase or not. This is useful when using printers that do not have the English codepage. The default is NO.

Layout of graphical reports

**GCR_SCALE**
Specifies the scaling factor for graphical reports when using BookMaster. The default is GCR_SCALE=.6.

**GCR_COMPACT**
YES/NO. Allows more objects to be displayed in a graphical report. The default is NO. Depending on the report type, a different maximum number of objects is shown on one page:

<table>
<thead>
<tr>
<th>Report Type</th>
<th>COMPACT=NO</th>
<th>COMPACT=YES</th>
</tr>
</thead>
<tbody>
<tr>
<td>CU</td>
<td>12 channels, 8 control units</td>
<td>16 channels, 10 control units</td>
</tr>
<tr>
<td>LCU</td>
<td>8 channels, 8 control units</td>
<td>8 channels, 8 control units</td>
</tr>
<tr>
<td>CHPID</td>
<td>8 channels, 64 control units</td>
<td>16 channels, 64 control units</td>
</tr>
<tr>
<td>Switch</td>
<td>1 switch</td>
<td>1 switch</td>
</tr>
<tr>
<td>CF</td>
<td>1 coupling facility</td>
<td>1 coupling facility</td>
</tr>
</tbody>
</table>

**GCR_FORMAT**
Specifies the formatting type:
BOOKIE
For BookMaster. This is the default.

DCF creates a data set for DCF containing script commands.
GML creates a data set for DCF containing GML tags.
GDF creates one or more members in GDF format for printing with
GDDM (not for batch).

GCR_FONT
Specifies the font to be used for printing; applicable only if
GCR_FORMAT=DCF or GCR_FORMAT=GML was specified. Specify the
appropriate font supported by your installation. For information on how to
create a graphical report, see “Create or view graphical configuration
reports” on page 238.

Color settings for graphical display function
(The following colors are valid specifications: green, red, blue, black, purple, pink,
yellow, brown, and white.)
COLOR_NORM
Specifies the color used when drawing the picture. Make sure that the
color is visible on defaulted or specified background. The default is
GREEN.
COLOR_TEXT
Specifies the color used for any text in the picture. The default is GREEN.
COLOR_HIGH
Specifies the color to be used when identifying a focused object. The
default is RED.
COLOR_BACKGROUND
Specifies the background color. If nothing is specified, the graphical display
function uses the standard background of the terminal.

Support of TSO option NOPREFIX

TSO_NOPREFIX
YES/NO. Specifies whether the TSO profile option NOPREFIX is
recognized by HCD. The default is NO. If you specify
TSO_NOPREFIX=YES, and the TSO option NOPREFIX is active, all data
set names specified in the HCD dialog are taken as is, i.e. HCD does not
add a high-level qualifier. Data set names that are generated by HCD act
independent of the TSO option NOPREFIX, and the user ID is added as
the high-level qualifier.

Esoteric token when migrating MVSCP input data sets

ESOTERIC_TOKEN
YES/NO. YES specifies that HCD will assign esoteric tokens in ascending
order when migrating an MVSCP input data set. NO (which is the default)
specifies that no tokens will be assigned.

Control unit type when migrating IOCP input data sets

MAP_CUTYPE
Specifies how a control unit type in an IOCP input data set is mapped to a
control unit type in the IODF. Specify one or more of the following
mappings:
MAP_CUTYPE = xxxx,yyyy-yy

xxxx
is the control unit type specified in an IOCP input data set

yyyy-yy
is the control unit type and model to be used in the IODF.
For example parameters, see the sample profile in Figure 11 on page 33.

**Extended migration**

* MIGRATE_EXTENDED

YES/NO. Specify YES to exploit the extended migration possibilities as described in “Changing I/O configurations by editing data sets” on page 273.

If you specify NO (which is the default), the additional keywords are not generated during IOCP build and when re-migrating IOCP input data sets, the migration function ignores the commented *$HCDC$ and *$HCD$ tags.

**Bypass IODF information update for SNA processor**

*BYPASS_UPD_IODF_FOR_SNA*

YES/NO. This provides a possibility of bypassing the attempt to update the IODF information for SNA processors after having successfully built the IOCDS.

If you specify YES, then no attempts will be made to update the IODF with IOCDS status information, which will usually fail as a result of the IODF being in an exclusive access mode with the dialog.

If you specify NO (which is the default), then for SNA and non-SNA processors, an attempt is made to update IOCDS information in the IODF after the IOCDS has been built successfully.

**Display information during ACTIVATE**

*SHOW_IO_CHANGES*

YES/NO. This option applies to dynamic activate. When performing both a hardware and software change, specify YES (which is the default) to get information about the channel paths, control units, and devices that are deleted, modified, or added.

**Loading an IODF into a data space**

*IODF_DATA_SPACE*

YES/NO. If you specify YES (which is the default), the IODF is loaded into a data space.

If you specify NO, the IODF is loaded into the user address space.

**IODF name verification for batch jobs**

*BATCH_IODF_NAME_CHECK*

YES/NO. If you specify YES (which is the default), HCD checks if the IODF specified for a batch job conforms to the naming convention as described in “IODF naming convention” on page 35. Processing of IODFs with invalid names is limited to deletion.

If you specify NO, HCD does not check the IODF names specified for batch jobs.

**IODF checker automation**

*CHECK_IODF*

YES/NO. If you specify YES, HCD checks an IODF for consistency and structural correctness whenever the IODF accessed in update mode is unallocated. This corresponds to the TRACE ID=IODF command and will consume processing time depending on the size of the IODF.

If you specify NO (which is the default), HCD does not check the IODF automatically.
**Delay device regrouping**

*DELAYED_GROUPING*

YES/NO. If you specify YES, HCD performs any necessary device regrouping after a device group split only when the IODF is closed. This gives a better response time in the HCD dialog for large IODFs.

If you specify NO (which is the default), HCD performs a necessary device regrouping each time when users exit the *I/O Device List*, or, in case the *I/O Device List* was called from either the *Operating System Configuration List* or the *Channel Subsystem List*, when leaving these lists.

**Default settings for OS parameters**

*OS_PARM_DEFAULT*

This keyword overrides a parameter default value set by the UIM. The value is used as a default on the HCD Define Device Parameters/Features panel. The syntax is:

```
OS_PARM_DEFAULT = xxxxxx,yyyyyy
```

where:

- `xxxxxx` is the parameter keyword
- `yyyyyy` is the new parameter default value

Example: The LOCANY default value is NO. You can change it to YES by specifying the HCD profile option:

```
OS_PARM_DEFAULT = LOCANY,YES
```

**Note:** Default values cannot be set differently for different device types. Features, for example SHARED, cannot be defaulted using this keyword.

**Extension of the attachable device list of a control unit**

*CU_ATTACHABLE_DEVICE*

This keyword allows the attachable device list of a control unit to be extended to include additional device types. Both the control unit type and the device type must be defined via UIMs. The value syntax is:

```
xxxxxx,yyyyyy
```

where:

- `xxxxxx` is the control unit type
- `yyyyyy` is the additional device type

Note that more than one device type can be added to the same control unit type.

Example:

```
CU_ATTACHABLE_DEVICE = RS6K,3174
CU_ATTACHABLE_DEVICE = RS6K,3274
```

A change of the value settings for this keyword will not be active until the next start of HCD.

**Show partition defaults in IOCP statements**

*SHOW_IOCP_DEFAULTS*

YES/NO. Use this option to write comment lines into the generated IOCP deck which show the partition assignments in effect for those CHPID and IODEVICE statements which make use of the IOCP defaults for the PARTITION and NOTPART values.
Setting this option to YES causes the suppressed PARTITION / NOTPART keywords to be generated as comments prefaced by the tag *$DFLT$*.

The default is NO.
Example
The following figure shows a profile with sample data:

```c
ACTLOG_VOL = * /* ACTlog on same volume as IODF */
ALLOC_SPACE = HCDASMP,60
BATCH_IODF_NAME_CHECK = NO /* changed to non-default NO */
BYPASS_UPD_IODF_FOR_SNA = YES /* No IODF update with IOCDs data */
CHLOG_VOL = *
COLOR_NORM = BLACK /* default was GREEN */
COLOR_TEXT = BLUE /* default was GREEN */
CU_ATTACHABLE_DEVICE = RS6K,3274 /* extend attachable device list */
ESOTERIC_TOKEN = YES /* Esoteric token: ascending order */
GCR_FONT = X0GT20
HCDDECK_VOL = D83WL2 /* Vol for configuration data set */
HCDJES3_VOL = D83WL4 /* Vol for JES3 output data set */
LINES_PER_REPORT_PAGE = 60 /* Max. number of lines per page */
MAP_CUTYPE = 3705,3745 /* Replace CU type during migration */
MAP_CUTYPE = 3880,3880-23 /* Replace CU type during migration */
MIGRATE_EXTENDED = YES /* Enable migration enhancements */
OS_PARM_DEFAULT = DYNAMIC,NO /* Default for parameter DYNAMIC */
OS_PARM_DEFAULT = LOCANY,YES /* Default for parameter LOCANY */
TSO_NOPREFIX = YES /* Enable TSO Noprefix (Default NO) */
```

Figure 11. Example of an HCD Profile
Customizing HCD EXEC procedures

Some of the HCD tasks, invoked from the dialog, generate batch jobs. These batch jobs use EXEC procedures, as shown in Table 3.

Your installation can use normal ISPF or TSO/E facilities to change the job control statements in these EXEC procedures. They are stored in the library SYS1.PROCLIB. You can customize these procedures according to your own needs.

You can also modify the EXEC procedures by using JCL overwrite statements in the HCD dialog. Thus, you can, for example, add a statement that refers to the HCD profile. See “Job statement information used in panels” on page 75 on how to specify JCL statements in the HCD dialog.

Table 3. Batch Jobs Used by the HCD Dialog

<table>
<thead>
<tr>
<th>HCD Task</th>
<th>EXEC Procedure</th>
<th>Job Step Name</th>
<th>More Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build an IOCDS</td>
<td>CBDJIOCP</td>
<td>GO</td>
<td>see “Build an IOCDS or an IOCP input data set” on page 316</td>
</tr>
<tr>
<td>Build an IOCP input data set</td>
<td>CBDJIOCP</td>
<td>GO</td>
<td>see “Build an IOCDS or an IOCP input data set” on page 316</td>
</tr>
<tr>
<td>Print a configuration report</td>
<td>CBDJRPTS</td>
<td>GO</td>
<td>see “Print configuration reports” on page 324</td>
</tr>
<tr>
<td>Compare IODFs and CSS/OS views</td>
<td>CBDJCMPR</td>
<td>GO</td>
<td>see “Compare IODFs or CSS/OS Reports” on page 328</td>
</tr>
<tr>
<td>Import an IODF</td>
<td>CBDJIMPT</td>
<td>IMP</td>
<td>see “Import an IODF” on page 332</td>
</tr>
<tr>
<td>Transmit part of an IODF</td>
<td>CBDJXMIT</td>
<td>GO</td>
<td>see “Transmit a configuration package” on page 53</td>
</tr>
</tbody>
</table>
Chapter 4. How to work with I/O definition files (IODF)

Overview

This information unit includes:
- IODF naming convention
- Working with I/O definition files (specify, change, create, view, backup, delete, copy, export, import, and upgrade IODFs)
- Working with large IODFs
- Activity Logging
- Using an IODF among different release levels

When you start an HCD session, you need to specify the IODF that HCD is to use. How to do this, how to change to another IODF, and how to use HCD tasks to maintain your IODFs is described below.

Before you can activate your configuration, you must build a production IODF. This task is described in "Build a production IODF" on page 196.

Note: The IODF data sets must be cataloged so that you can use them with HCD.

IODF naming convention

You need to comply to naming conventions for work IODFs, production IODFs, and further data sets associated to an IODF (activity log, change log and HCM MCF data set).

The IODF is a VSAM LINEAR data set with different names for the cluster component and the data component. The name of the data set with a cluster component has the format:

'hhhhhhhh.IODFcc{.yyyyyyyy. ... .yyyyyyyy}.CLUSTER'

The name of the data set with a data component has the format:

'hhhhhhhh.IODFcc{.yyyyyyyy. ... .yyyyyyyy}'

Work IODF

The data set name for a work IODF has the format of:

'hhhhhhhh.IODFcc{.yyyyyyyy. ... .yyyyyyyy}'

'hhhhhhhh' is the high-level qualifier; up to 8 characters long.

'cc' is any two hexadecimal characters (that is, 0-9 and A-F).

'yyyyyyyy' are optional qualifiers, separated by a . and up to 8 characters long. The following qualifiers must not be used as last qualifier: CLUSTER, ACTLOG, CHLOG or MCF.
You can use any number of optional qualifiers but do not make the total name longer than 35 characters because, in some circumstances, HCD appends an additional qualifier.

If you use a change log or an HCM master configuration file, the total IODF name must not exceed 29 characters.

If you omit the high-level qualifier and the enclosing single quotation marks, HCD automatically adds your user prefix (your user ID is the default).

**Production IODF**

The data set name for a production IODF has the same format as a work IODF. You may specify additional qualifiers to differentiate among IODFs (for example for backup reasons). However, the optional qualifiers must be omitted if the IODF is to be used for IPL or dynamic activation. Thus, the format would be:

`'hhhhhhhh.IODFc'`

`hhhhhhhh` is the high-level qualifier; up to 8 characters long.

`cc ` is any two hexadecimal characters (that is, 0-9 and A-F).

**Associated data sets**

Files associated to an IODF, if used, also must conform to the IODF naming conventions, plus a required last qualifier:

*Activity log* (a sequential fixed 80 character data set):

`'hhhhhhhh.IODFc{.yyyyyyyy. ... .yyyyyyyy}.ACTLOG'`

*Change log* (a VSAM LINEAR data set with cluster component and data component):

`'hhhhhhhh.IODFc{.yyyyyyyy. ... .yyyyyyyy}.CHLOG'`

*HCM master configuration file* (MCF, a VSAM LINEAR data set with cluster and data components):

`'hhhhhhhh.IODFc{.yyyyyyyy. ... .yyyyyyyy}.MCF'`

**Create or specify an IODF**

You specify the name of the IODF that you want to use on the Primary Task Selection panel. The first time you use the dialog, HCD puts the default name `SYS1.IODF00.WORK` in the IODF name field (see Figure 28 on page 62). You can type over this name to specify the name you want to use.

If you specify an IODF name that does not exist, HCD assumes that you intend to create a new IODF and displays a panel to let you specify the required attributes. HCD then creates and automatically catalogs the IODF.
Volume serial number (of the data volume the IODF will reside on)
This entry is ignored if your system is managed by SMS, otherwise it is mandatory.

Space allocation
The online HELP gives advice on how much space to allocate. If you run out of space while working with an IODF, you can use the Copy IODF task to copy the IODF to a larger data set.

Activity logging
You have to decide now whether you want HCD to maintain an activity log for the IODF; you cannot specify it later. If you want to use an activity log, your system must have Program Development Facility (PDF) installed.

Multi-user access
Specify whether or not you want to enable the IODF for multi-user access.

Description
Here you may enter any useful additional information concerning the IODF, for example, the system it applies to, a special purpose of the IODF, or the author.

The IODF remains in effect throughout all tasks of your current session and later HCD sessions, until you change it.

Multi-user access
Up to z/OS V1R9 HCD, multiple users could read an IODF simultaneously, but no user could read an IODF while it was accessed in update mode by another user. Also, a user could only update an IODF, if no other user accessed the IODF, neither in read nor update mode.

Starting with release z/OS V1R10 HCD, when creating an IODF, you can specify a multi-user access option in the Create Work I/O Definition File dialog. The default is single-user access.

Having exclusive access to an IODF, users can also switch between single-user mode and multi-user access using an option in the Change I/O Definition File Attributes dialog.
Notes:
1. To enable an IODF for multi-user access you need ALTER access authority.
2. You can check whether the multi-user access property is enabled for an IODF using View I/O definition file information from the Maintain I/O Definition Files task.

With the multi-user access option specified, an IODF is kept in exclusive update mode for a user only for the duration of a single transaction. If the updates of this transaction are committed, another user may update the IODF without requiring the first user to release it.

If a user is updating a multi-user access enabled IODF, HCD implements the following processing: HCD locks the IODF. If multiple subsequent users now also want to apply concurrent updates to the same IODF, they must wait in a queue. However, since the first user’s single transaction may last a split-second only, HCD repeats all other users’ subsequent update requests a couple of times. If all attempts fail, for example, because the first user updates the IODF using the dialog and is delaying required input, HCD will notify all other requesting users with a message, telling who is currently updating the IODF.

Associated change log files inherit the multi-user access ability from the IODFs.

**Note:**
When several users simultaneously work on the same IODF, exploiting the multi-user access capability, it is recommended that they co-ordinate their activities in order to ensure the consistency and integrity of the changes made to the IODF.

**How to release a lock after an abnormal termination:** In most cases of abnormal termination while working with IODFs in multi-user access, HCD invokes a recovery routine that deletes a pending lock. If, however, HCD cannot enter this routine, the lock remains active, and any user, when trying to access the IODF next time, receives a message about who is holding the lock.

In such a case, a user with ALTER access right must re-access the IODF exclusively and select action Change I/O definition file attributes to set back the multi-user access capability to NO. This action deletes the lock, and multi-user access can now be reactivated for that IODF.

**Sharing IODFs**
If you want to share an IODF across two or more systems, you must:
1. Catalog the IODF in the user catalog that is shared by those systems.
2. Define an ALIAS to that catalog in the master catalog of each system that uses the IODF (for details, refer to "Catalog considerations" on page 343).

**Note:** Control of sharing the IODF resource between multiple systems is achieved via Global Resource Serialization (GRS).

**Important:**
If you update an IODF simultaneously from different systems that are not in the same GRS complex, you may destroy data in the IODF.
Deciding on the number of IODFs

The decision whether to create one IODF for each processor, or to combine the I/O definitions for two or more processors in a single IODF, depends on your environment. This section explains when it is advantageous or even necessary to keep the I/O definitions of two or more processors in the same IODF.

**Shared control units and devices**
If control units and devices are shared by different processors, the I/O definitions for these processors should be kept in the same IODF to keep change effort to a minimum and to avoid conflicting definitions.

**Processor and related OS configuration(s)**
For a full dynamic reconfiguration, the IPLed OS configuration must be in the same production IODF as the processor configuration selected for POR.

**Coupling facility support**
For coupling facility support, you have to maintain your coupling facility definitions for the processors to be connected in the same IODF.

**Switch connections**
It is recommended that you maintain your switch configurations in the same IODF as the hardware and software configuration definitions to provide complete validation of the data path.

In order to lookup the port connections of a switch, all connected objects to the ports of a switch have to be defined in the same IODF.

**CPCs of an S/390 microprocessor cluster**
To manage IODFs and IPL parameters within the CPCs of an S/390 microprocessor cluster from a focal-point HCD, the corresponding processor configurations have to be kept in the same IODF.

**Dynamic sysplex reconfiguration**
To dynamically reconfigure the I/O configuration of a system within a sysplex from a focal-point HCD, the processor and OS configuration of the sysplex system have to be defined in the same IODF.

**CTC connection report**
All CTC connections listed in a CTC connection report must be defined in the same IODF. Misconfigurations can be detected only within the scope of one IODF.

**Reporting**
The scope of the reports (textual or graphical) is a single IODF. All I/O definitions required for a report must be kept in the same IODF.

**Validation**
The scope of the validation function is a single IODF.

**HCM**
The scope of the configuration shown by HCM is a single IODF.

These requirements and recommendations may lead to a large IODF, depending on the size of the installation. The number of elements in the single IODF may be too large for effective management. HCD provides the possibility of creating manageable *subset IODFs* with a scope limited to a part of the I/O configuration from a *master IODF* describing the entire configuration. For details on this IODF management strategy refer to “The master IODF concept” on page 40.
The master IODF concept

In order to take full advantage of the available HCD functions and to keep a maximum of freedom for reconfigurations it is proposed to keep a master IODF. A master IODF may contain the I/O definitions for an entire enterprise structure. However, a user may decide to divide the I/O configuration definitions of the whole enterprise into several master IODFs, where little or no interference is expected.

Major configuration changes are done in the master IODF. These include:
- processor configurations,
- OS configurations,
- switch configurations, and
- definitions spanning multiple configurations, e.g. coupling facility connections.

After such changes have been made, a production IODF is built. From this IODF comprehensive reports can be obtained.

HCD provides a set of functions which allows the management of very large IODFs for their activation on individual systems, like POR, IPL, dynamic I/O changes, or for S/390 microprocessor management tasks. Using these functions, subset IODFs containing I/O definitions relevant to only one particular system may be built from the master IODF.

There are no strict rules about what a subset IODF must consist of. Typically it contains:
- a processor configuration with its related OS configuration, or
- all I/O configurations describing the CPCs in an S/390 microprocessor cluster, or
- all I/O configurations describing the systems of a sysplex.

The content of a subset IODF is specified in a configuration package (see “Work with configuration packages” on page 50).

The subset IODF is transferred to the corresponding target system where it is imported and used as the IODF for that system. A subset IODF constitutes a fully functional IODF. When it is built from a master IODF, the processor tokens are preserved. If necessary, updates concerning the target system alone may be carried out using the subset IODF. Subsequently, the subset IODF can be sent back to the system administering the master IODF and merged back into the master IODF, thereby updating it with the changes made at the target system.

The following chart illustrates the possible flow of I/O information according to the master IODF concept.
How HCD arranges devices into groups in an IODF

IODF versions up to V4 contained a separate device definition record for each single device. To reduce the size of IODFs and to improve the processing performance of large configurations, z/OS V1.7 HCD uses a new IODF format V5, arranging single devices into a device group, if they have the following characteristics in common:

- The device numbers of all devices are in consecutive sequence.
- All devices of the group have the same device type (unit, model) and attribute values (Serial-#, Description, VOLSER).
- All devices of the group are attached to the same control unit(s).
- All devices of the group are connected to the same processors/channel subsystems and have the same corresponding processor-specific attributes.
- All devices of the group are connected to the same operating systems and have the same corresponding OS-specific attributes (device type, parameters, features, console definition, subchannel set number).

Figure 13. IODF Distribution/Merge Process
For each OS and each eligible device table (EDT) in the OS, all devices of the

group are connected to the same esoterics.

Devices that adhere to the these rules are aggregated into device groups containing

the maximum number of applicable devices. If you apply a change on one or more
devices from a group, HCD checks how to rearrange the devices and device
groups contained in the IODF in order to achieve the best possible organisation of
devices into groups again according to the specified rules.

Change to another IODF

You can work with only one IODF at a time. If, during an HCD session, you want
to change to another IODF, you must return to the Primary Task Selection panel
and specify the new IODF. If the old IODF has an activity log and has been
modified, a panel is displayed to let you add a comment into the activity log
(“Activity logging and change logging” on page 54 tells you how). The new IODF
now becomes the current IODF.

Change a production IODF / Create a work IODF based on a
production IODF

HCD allows you to perform all tasks on the data stored in a production IODF as
long as you do not try to change it. If you try though, HCD displays the Create
Work I/O Definition File panel where you can define a new work IODF based on
the current production IODF.

HCD then copies the production IODF to that new work IODF, makes the work
IODF the currently accessed IODF, and applies to it all further changes.

You must specify a new data set for the work IODF; you cannot use an existing
one. HCD creates a default work IODF name by appending the qualifier WORK to
the production IODF name. You can change this default work IODF name.

When you have completed the changes, you can use HCD to build a new
production IODF from the work IODF.

View active IODF

HCD provides information about the IODF that has been used for last OR/P or
for dynamic activation (that is, the currently active IODF); in addition the
operating system ID and EDT ID used for IPL are shown, and the configuration
token that is currently active in the HSA (hardware system area). For a description
of this function, see “View active configuration” on page 219.

Backup work or production IODFs

You can use the Backup action bar choice on any action list panel to copy a work
or production IODF to a backup data set. Thus you can keep track of different
stages of the configuration as well as retrieve data lost by accident. When you do
the first backup, you must also specify the volume serial number, if applicable, and
the space allocation for the backup data set. HCD uses the normal catalog process
to catalog copies of the IODFs.

If you backup a work IODF, you need to specify the name of the backup data set
only once for each IODF (when you do the first backup). HCD saves the name of
the backup IODF data set, so you can reuse this data set for each subsequent backup (or use a different one if you want).

If you backup a production IODF, HCD does not save the name of the backup data set, because a production IODF cannot be edited. It is suggested that you maintain a backup copy of your production IODF on a separate volume that is accessible from all systems that will be sharing the backup. When the primary IODF volume is inaccessible or the IODF itself is corrupted, the system can be IPLed through a backup IODF on the alternate volume.

It is also recommended that you choose an alternate high level qualifier for your backup IODF since a lost IODF volume may imply a lost IODF catalog. This high level qualifier can be cataloged in either the master catalog or in an alternate user catalog.

As an alternative method to create a backup IODF, you can use the following procedure:
1. Select **Maintain I/O definition files** from the **Primary Task Selection panel** (Figure 9 on page 23).
2. Select **Copy I/O definition file** from the **Maintain I/O definition Files** panel (Figure 14).
3. Specify the name of the backup IODF.

### Maintain IODFs

HCD provides the tasks listed on the Maintain I/O Definition Files panel (Figure 14) to help you maintain your IODFs. You can reach this panel from the Primary Task Selection panel (see Figure 28 on page 62).

#### Delete an IODF

This task deletes an IODF. If the IODF has an activity log, that log is also deleted. Also, if an HCM master configuration file (MCF) is associated with the IODF, it is deleted along with the IODF. HCD asks you for confirmation before actually deleting the IODF.

#### Copy an IODF

You can invoke the task **Copy I/O definition file from the Maintain I/O Definition Files panel** (Figure 14). This task copies any IODF to another IODF (either existing or new). You must specify the name, volume serial number, if applicable, and the
space allocation of the target data set. If the IODF has an activity log, that log may also be copied. Also, if an HCM master configuration file (MCF) is associated with the IODF, it is copied along with the IODF. However, a change log file (CHLOG), if available, is not copied.

During a definition task, you can use **Copy IODF** to copy the existing data to a larger data set if you have allocated insufficient space to a work IODF. In this case, you need to return to the Primary Task Selection panel afterwards to specify the new data set as the IODF you are working with from now on.

You can also use the **Copy I/O definition file** task to upgrade a V4 IODF to a V5 IODF. The result of the copy process will always be a V5 IODF.

In the **Copy I/O Definition File** dialog, if you specify a target IODF that does not yet exist, HCD displays the dialog from [Figure 15](#) where you can create a new target IODF. The space allocation default depends on the source IODF:

- for a V5 source IODF, the allocation default is the number of allocated blocks of the source IODF.
- for a V4 source IODF, the allocation default is the number of used blocks of the source IODF.

If you copy an IODF which is enabled for multi-user access, this property is not inherited by a target IODF. However, an existing target IODF defined with the multi-user access property will always preserve this property, independent from the source IODF.

You can also invoke the **Copy I/O definition file** task in batch mode. For details refer to [“Copy IODF” on page 323](#).

### Change IODF attributes

With this task, you can change certain attributes of an IODF. You can change the description and you can enable or disable the IODF for multi-user access.

**Note:** To enable or disable an IODF for multi-user access you need ALTER access authority.
View an IODF

This task displays information about the currently accessed IODF. The information includes the type and version of the IODF, its description, the creation date, the last update, and how much of the allocated space for the IODF data set has been used.

You can also invoke the View IODF task from the View action bar, and by issuing the SHOWIODF command from the command line.

![View I/O Definition File Information](image)

**Figure 16. View IODF information**

Export an IODF

This task sends an IODF, and optionally, its activity log file to another (local or remote) system. On the Export IODF panel, specify or revise the IODF name you want to export, the user ID, or the nickname (only for an attended target system), node ID, and status (attended or unattended) of the operating system (OS) to which the IODF is to be sent.

If the target system is unattended, the IODF is sent as a job to the target system, which must be a system of type MVS. In this case no explicit action on the target system is required. You can specify whether to replace an existing IODF with the same name. If the existing IODF is the active IODF for the remote system HCD is running on, replace will not be possible.

If the target system is attended (receiving to be done by the user on the remote system), the IODF is sent to the target system as a sequential data set. There it has to be received by using the TSO RECEIVE command. As a second step the IODF is imported on the target system (see also “Import an IODF” on page 48).

To export an IODF, HCD uses the TSO command TRANSMIT. Therefore, HCD creates a cataloged sequential data set named tsoid.EXPORTED.iodfname, where
tsoid is the sending TSO user ID or the TSO prefix, or is determined by the profile option EXPORTED_HIQ, and iodfname is the part of the IODF data set name after the high-level qualifier.

After processing the TRANSMIT command, the sequential data set is deleted.

If you have specified to send the selected IODF to a system with an operating system running in unattended mode, use the panel from Figure 17 to define the job control language (JCL) statements for importing the IODF on that unattended system.

You can also invoke the Export IODF task by using the Export task in batch mode. For details refer to “Export an IODF” on page 333.

Prerequisites:
1. Network Job Entry (NJE) must be active.
2. The target user ID and password, and, if the target system is not SMS managed, the volume serial number of the receiving data volumes have to be known when the IODF is exported to an unattended system.

User authentication for unattended mode
If you export an IODF to a target system running in unattended mode, there are two alternatives for user authentication:
- sending the target user ID and password to the target node (that is, the user ID which receives the IODF)
- defining a surrogate user ID on the target to act on behalf of the receiver

Sending user ID and password:
To authenticate the receiving user with its user ID and password at the receiving system, enter the correct password on the panel from Figure 18 on page 47 twice.
Defining a surrogate for the receiving user ID:

If you want to avoid sending passwords across the net, on the receiving system, you can define a surrogate user ID for the receiving user. The import job, submitted by the surrogate user, will run with the identity and authorization of the receiving user, without a password being sent.

To enable user authentication without sending a password, perform the following steps:

1. Define a surrogate user ID for the receiving user and the appropriate access rights for the sending and receiving users as shown in the example hereafter. This step is required as a setup only once.

2. When you export an IODF in unattended mode, on the Specify Target IODF and User Password panel (Figure 18), enter a dummy character for the password and password confirmation (for example, an "*") to suppress sending of a password to the receiving target system.

3. On panel Define JCL for IODF Import panel (Figure 17 on page 46), replace the statement

   /*LEAVE THIS JOB CARD UNCHANGED: USERID AND PASSWORD ARE GENERATED

   with the target/receiving user ID to provide the following JCL:

   //JOBNAME JOB (ACCT,BOX),'USER',CLASS=CLASS,
   //MSGCLASS=MSGCLASS,MSGLEVEL=(1,1),
   //USER=user_r
   /*ROUTE XEQ HCD3

   In the example shown below, user_s and node_s denote user ID and node ID of the sender and user_r and node_r denote the respective ids of the receiving system (running an operating system in unattended mode).

On node_r issue:

RDEFINE NODES node_s.USER*,user_s UACC(UPDATE)
RDEFINE SURROGAT user_r.SUBMIT UACC(NONE) OWNER(user_r)
PERMIT user_r.SUBMIT CLASS(SURROGAT) ID(user_s) ACCESS(READ)

- The first statement controls whether jobs coming from node_s are allowed to enter the system from node_s. It also controls, whether jobs that enter the system from node_s nodes have to pass user identification and password verification checks.
• The second statement defines a \texttt{user\textunderscore r.SUBMIT} profile in the \texttt{SURROGAT} general resource class for \texttt{user\textunderscore r} who requires a surrogate user to act on his behalf.
• The third statement authorizes \texttt{user\textunderscore s} to act as a surrogate for \texttt{user\textunderscore r}.

**Import an IODF**

This task imports previously received IODF data into HCD. You can export and import IODFs between different HCD versions. You can, for example, export an IODF from HCD 1.4 and import it with HCD 1.7. Note that in this case you have to upgrade the lower version IODF before using it with HCD, because the export/import function does not change the format of the IODF.

You can also invoke the \texttt{Import IODF} task by using the \texttt{Import} task in batch mode. For details refer to "Import an IODF" on page 332.

At an unattended target system, the IMPORT batch utility is invoked automatically when an IODF, with its associated job control, arrives at the system.

**Upgrade an IODF**

This task upgrades an IODF from a back-level format to the new format that is required for the current release of HCD.

Invoke the \texttt{Upgrade I/O definition file to new format} task as follows:
1. On the HCD Primary Task Selection panel (Figure 9 on page 23), specify the IODF to be upgraded at the bottom of the menu and then select option 6, \texttt{Maintain I/O definition files}.
2. From the Maintain I/O Definition Files panel (Figure 14 on page 43), invoke option 8, \texttt{Upgrade I/O definition file to new format}.

HCD displays the following dialog:

```
Table 4 on page 49 shows the options you have. You can either:
• Upgrade into a new work IODF
• Upgrade in place
```
Table 4. Size considerations when upgrading a back-level IODF

<table>
<thead>
<tr>
<th>Upgrade IODF</th>
<th>Upgrade into New IODF</th>
<th>Upgrade in Place</th>
</tr>
</thead>
<tbody>
<tr>
<td>From</td>
<td>To</td>
<td></td>
</tr>
<tr>
<td>V4 IODF</td>
<td>V5 IODF</td>
<td>1. with Condense option: Default size of new IODF is double the size of the related production IODF. You can change this default size, if necessary.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. without Condense option: Default size of new IODF is the size of V4 IODF.</td>
</tr>
</tbody>
</table>

Notes:
1. The Upgrade in place option cannot be requested for a production IODF (as a production IODF cannot be altered). The final result of the upgrade IODF function is always a work IODF.
2. If you plan to add or change many devices in the configuration, ensure that you increase the space allocation when creating the IODF to allow for these changes.

If you select to upgrade in place, the accessed IODF is formatted to a V5 IODF. While the allocated and used space values do not change, the space utilization of the used blocks will decrease depending on the number of devices that can be grouped.

If you select to upgrade to a new IODF, the accessed IODF will not be changed. HCD displays the Create Work I/O Definition File dialog (see Figure 20). The space allocation default is set as follows:

- If you select Condense IODF = Yes, the default allocation value is set to double the size required for a corresponding production IODF. This may result in much lower space requirements than without condensing.
- If you select Condense IODF = No, the default allocation value is set to the number of used blocks of the IODF to be upgraded.

---

Create Work I/O Definition File

The current IODF was created by an earlier HCD release, and you have requested upgrade to a new data set. To create this data set, specify the following values.

Source IODF .... : 'DOCU.IODF00.TREXDOCU.HCMUG'
Allocated space . : 1024
Used space .... : 27
IODF name ...... _____________________________________
Volume serial number . ______ +
Space allocation . . 64 (Number of 4K blocks)

F1=Help F2=Split F3=Exit F4=Prompt F9=Swap F12=Cancel

Figure 20. Create Work I/O Definition File
After the upgrade completed successfully, HCD issues a message of success and accesses the upgraded IODF with which you can start to work now.

You can also invoke the Upgrade I/O definition file to new format task in batch mode. For details refer to "Upgrade IODF" on page 310.

Note: An IODF can also be upgraded using the Copy IODF task (see "Copy an IODF" on page 43).

Work with configuration packages

Configuration packages define subset IODFs which are extracted from the accessed IODF (centrally administered master IODF) and distributed for activation at selected target systems, while keeping the processor token in sync.

When invoked with option Work with Configuration Packages from the panel shown in Figure 14 on page 43 this task displays the Configuration Package list. Configuration packages can be added and edited, transmitted from a production IODF, and they can be deleted.

The target user and node as well as the description of a configuration package can be edited by overwriting the information on the panel. When scrolling to the right you can also overwrite the target IODF name and attended/unattended information.

Define a configuration package

To define a new configuration package perform the following steps:

1. On the Configuration Package List panel use F11=Add. If you want to use an existing configuration package as a model, select the package and the Add like action from the context menu (or use action code /). The Add Configuration Package panel is displayed.

Figure 21. Sample Configuration Package List — left panel
2. Enter a package name and other entry data as appropriate. The user ID and node specify the destination of the corresponding subset IODF. The name of the accessed IODF is used as default for the name of the IODF at the target system and for the Descriptor field 1 and Descriptor field 2 parameters during the Build Production IODF step, when the configuration package is transmitted (see step 5 on page 54 in section "Transmit a configuration package" on page 53). If no value is entered for the Operating system status, Attended is assumed which means that a user has to import the subset IODF into HCD at the target system. This is done automatically if 2 (Unattended) is specified.

3. After you press ENTER, HCD displays the updated Configuration Package list.

**Edit a configuration package**

The target user and node, the attended/unattended information, the target IODF name, and the description of a configuration package can be edited on the Configuration Package List panel (see Figure 21 on page 50).

To edit a configuration package perform the following steps:

1. Display the Configuration Package List panel.
2. Make changes to the configuration package by overwriting the entries on the list panel.
   - To change the IODF name and attended/unattended information scroll to the right and overwrite.
3. Press ENTER.

**Delete a configuration package**

To delete a configuration package perform the following steps:

1. On the Configuration Package List panel select the package and the Delete action from the context menu (or use action code 4). The Confirm Delete Configuration Package panel is displayed.
2. Press the Enter key to confirm the deletion. HCD displays the updated Configuration Package list.

**Work with configuration package objects**

Configuration package objects are operating systems or processors. The objects in a configuration package determine the scope of the corresponding subset IODF. To
change the content of a configuration package select the package from the
Configuration Package List panel (see “Work with configuration packages” on page 50 and the Work with Configuration Package Objects action from the context menu (or use action code \SF 580000\SF 590000). The Configuration Package Objects List panel is displayed.

Add a configuration package object: To add a configuration package object to a configuration package perform the following steps:
1. Use F11=Add or select an object from the Configuration Package Object List panel and the Add like action from the context menu (or use action code \SF 580000\SF 590000).
   The Add Configuration Package Object panel is displayed.
2. Select the Configuration type, processor (PR) or OS configuration (OS), and specify the Configuration ID of the object.
3. Press the Enter key. The updated Configuration Package Object List panel is displayed.

Merging changes into a master work IODF: If a system has been supplied with a subset IODF and configuration changes have been made using the subset IODF, you may want to update the master IODF with these changes. This can be done by merging the changed configuration package objects from the subset IODF back into the master work IODF.

First the updated subset IODF is exported on the corresponding system and imported at the system administering the master IODF. You can update the master work IODF by merging or by replacing configuration package objects.

The Merge action updates new/changed data in the master work IODF.

The Replace action first deletes the object configuration present in the master work IODF and then replaces it with the new one.

To merge a configuration package object into a master IODF perform the following steps:
1. Access the master work IODF the configuration package belongs to.
2. Display the Configuration Package Object List panel and select one or more objects. Select Merge (action code \SF 580000\SF 590000) or Replace (action code \SF 580000\SF 590000) from the context menu.
3. Specify the source IODF the configuration package refers to and press the Enter key.
4. The master work IODF is updated and a new master production IODF can be built.

Notes:
1. Switch configurations are not repeated by Merge and Replace. If necessary, switch configuration changes should be made in the master IODF before Merge or Replace are performed.
2. The Merge and Replace actions utilize the Repeat action for entering the new configuration data into the master work IODF. Refer to “Repeating (copying) processors” on page 96 and “Repeating (copying) operating system configurations” on page 83 for details on the Repeat action.

Delete configuration package objects: To delete a configuration package object from a configuration package perform the following steps:
1. Select an object from the Configuration Package Object List panel and the **Delete** action from the context menu (or use action code 4).

2. Press the Enter key. The updated Configuration Package Object List panel is displayed.

**Transmit a configuration package**

Transmitting a configuration package means building a subset IODF and transmitting it to the target system specified in the configuration package. This action can only be carried out from a production IODF. To transmit a configuration package perform the following steps:

1. Select the package from the Configuration Package List panel and the **Transmit configuration package** action from the context menu (or use action code X).

2. If the following conditions are fulfilled, the Transmit Configuration Package panel is displayed:
   - The accessed IODF is a production IODF.
   - The selected configuration package contains at least one configuration object (OS or processor).
   - In the selected configuration package a destination user and node are specified.
   - In the selected configuration package a target IODF is specified.

   **Descriptor field 1** is defaulted to the HLQ of the target IODF name in the configuration package; **Descriptor field 2** is defaulted to the 2nd qualifier of the target IODF name in the configuration package.

   ```plaintext
   Transmit Configuration Package
   Package name .... . : PACK1 Package for BOEHCD1
The master production IODF is updated with the last sent date and time when the job stream is built. The JCL member provided, CBDJXMIT, consists of the following steps:

1. A temporary work and a temporary production IODF, according to the values entered in the Transmit Configuration Package panel are initialized.
2. The processor configurations included in the selected configuration package are written to a data set and then migrated to the temporary work IODF.
   To generate coupling facility connections, HCD needs both the CF sender and CF receiver channel paths, or peer channel definitions, within the same IODF. Therefore, if a processor of the configuration package contains a connected sending CF channel path, the processor containing the coupling facility partition will be included (with the coupling facility partition only) even if it is not part of the configuration package. (The receiving CF channel paths of the CF partition are indicated as occupied if they have connections to processors outside of the package.)
   The switches and ports which contain connections to a processor of the configuration package are distributed as well. Ports that contain connections to a processor, switch or control unit outside of the scope of the configuration package are indicated as occupied.
3. The OS configurations related to the selected configuration package are written to a data set and then migrated to the temporary work IODF.
4. The switch configurations related to the selected configuration package are written to a data set and then migrated to the temporary work IODF.
5. A temporary production IODF is built from the temporary work IODF.
   The processor token is not changed by this action.
6. The production IODF is exported to the specified user ID and node, attended or unattended, as selected.
7. The temporary work and production IODFs are deleted.

The transmit action can also be carried out using a batch utility. A sample job, CBDSXMIT, has been provided in SYS1.SAMPLIB.

**Activity logging and change logging**

When you end an HCD session or access a different IODF after modifying some configuration data, and if an activity log is enabled for the currently accessed IODF ("Create or specify an IODE" on page 36 explains how to specify an activity log), then HCD displays an activity log panel, showing the information which will be added to the activity log.

```
Date & Time ..........: 2005-11-07 10:54:46
User ...............: BOKA
I/O definition file ..: 'BOKA.IODF00.ACT'
Change reference number : 000002

****** *********************** Top of Data ***********************
000001 Type your log entries here ...
000002 ...
000003 ...
000004 ...

****** *********************** Bottom of Data ***********************
```

Figure 24. Example of an Activity Log without automatic logging
This information comprises the date and time, the user ID of the user who modified the IODF, the name of that IODF, and a change reference number. You can add your own comments to the log, for example, describing what you have done.

You can enable **automatic activity logging** by entering the following keyword into the HCD profile:

```
CHANGE_LOG = YES
```

With this setting, HCD generates automatic entries into the activity log panel, describing the updates on HCD objects, like for example, add, delete, or connect, disconnect. You can see examples of such entries in Figure 25 on page 56. You can edit the entries before you exit this panel.

The activity log panel is an ISPF/PDF panel, so the normal ISPF/PDF rules apply to it. Activity log editing requires the profile option AUTOSAVE ON. When calling the ISPF editor, macro CBDCACTL is used. You can tailor this macro to your installation needs (for example, for setting specific profile options). Use the F3=Exit key to continue.

HCD appends the qualifier ACTLOG to the IODF data set name to create the data set name for the activity log. If the ACTLOG data set does not yet exist, HCD dynamically allocates one using ESOTERIC system defaults (see “SMS-related considerations” on page 344). If you want to use a specific volume, you can specify the volume serial number to allocate a new activity log in the HCD profile (see “Defining an HCD profile” on page 23).

You can view or print the activity log associated with the currently accessed IODF during an HCD session by selecting the option **Print or compare configuration data** from the Primary Task Selection panel and then **View the activity log** or **Print the activity log**.

You can also use the ISPF/PDF facilities to browse or print an activity log.

HCD generates its proposed activity log entries from the change log file. Setting the `CHANGE_LOG = YES` option in the HCD profile causes HCD to create the change log file and store in it all update operations on the related IODF in a wrap-around manner.

The change log file is a VSAM data set. If it is not accessible for any reason, HCD cannot create any automatic activity log entries.

The name of the change log file is built from the related IODF data set name plus the suffix CHLOG. It is allocated in the same size as the related IODF.

You can write the contents of the change log file to the HCD trace data set using the `TRACE` command with ID=CLOG:

```
trace on, id=clog, level=8
```

With the profile options `CHLOG_VOL` and `ACTLOG_VOL`, in non SMS-managed environments, you can specify the volume serial numbers where to allocate a new change log or activity log.
**Rules for automatic activity logging**

HCD applies the following rules during automatic activity logging:

- When creating an object with immediate updates without leaving HCD, HCD only creates an 'add object' log entry in the activity log.
- When updating the same object multiple times during one HCD session, HCD creates only one update object entry.
- Consecutive IDs, for example for channel subsystems, channel paths, control units or devices, are marked with a ‘quantity’ number in parenthesis behind the starting ID.
- When repeating an object into a new IODF, no entry is created in the target IODF since this IODF is not in access by the user.

**Actions performed on IODFs and related activity and change log files**

- When you delete an IODF, HCD also deletes the related activity and change log files.
- When you copy an IODF, HCD also copies the activity log file with all its contents. However, HCD does not copy the change log file. Instead, a new change log file is created for the new IODF.
- When you build a production IODF, HCD copies the activity log file, but not the change log file.
- When you enable or disable an IODF for multi-user access, the same action is applied to the change log file. Before an update request is performed on an IODF, the change log file is refreshed to have the latest updates available. The activities of multiple users are logged in chronological order.
- An activity log data set is not enabled for multi-user access. If multiple users simultaneously access and update the related IODF, the activity log file is
accessed sequentially in the order the users end the HCD session or access a different IODF. The users’ activity log entries are written to the data set grouped by the user ID.

**IODF release level compatibility**

If you plan to share an IODF among multiple z/OS or OS/390 systems that are at different release levels, you have to consider several restrictions concerning IPL, IODF usage, and dynamic reconfiguration.

**Table 5** shows possible OS/390 and z/OS HCD levels that could be installed on your system. For each HCD level, the table shows whether you can take the following actions:

<table>
<thead>
<tr>
<th>HCD installed with:</th>
<th>Action</th>
<th>IODF Level</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>V4 IODF</td>
</tr>
<tr>
<td>OS/390 R9 HCD (1)</td>
<td>IPL</td>
<td>Yes</td>
</tr>
<tr>
<td>(HCS6091)</td>
<td>HCD</td>
<td>Yes (8)</td>
</tr>
<tr>
<td></td>
<td>Dyn</td>
<td>Yes (7,8)</td>
</tr>
<tr>
<td>z/OS 1.4 HCD (2)</td>
<td>IPL</td>
<td>Yes</td>
</tr>
<tr>
<td>(HCS7708)</td>
<td>HCD</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Dyn</td>
<td>Yes (7)</td>
</tr>
<tr>
<td>z/OS 1.7 HCD (HCS7720)</td>
<td>IPL</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>HCD</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Dyn</td>
<td>Yes</td>
</tr>
<tr>
<td>z/OS 1.9 HCD (HCS7740)</td>
<td>IPL</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>HCD</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Dyn</td>
<td>Yes</td>
</tr>
<tr>
<td>z/OS 1.10 HCD (HCS7750)</td>
<td>IPL</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>HCD</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Dyn</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Notes:**

1. OS/390 R9 HCD runs on OS/390 R10, z/OS 1.1, z/OS 1.2, z/OS 1.3 and z/OS 1.4.
2. z/OS V1.4 HCD runs on z/OS 1.4 Feature 0, z/OS 1.5 and z/OS 1.6. (As z990 Compatibility Support it has also be provided back to OS/390 R10.)
3. The Allocation Compatibility PTF for the corresponding z/OS release (APAR OA08197) is required.
4. The HCD R7 Compatibility PTF has to be installed (APAR OA07875). It provides limited read-only access to the V5 IODF.
5. Alternate subchannel sets are ignored.
6. Read-only access is provided. Upgrade of V4 IODF to V5 IODF is required for updates.
7. If the IODF contains processor configurations with hardware definitions that are not known to the corresponding OS/390 or z/OS release, you must use a z/OS release that has the corresponding support installed.
8. If the IODF contains processors supporting multiple channel subsystems, z/OS 1.4 HCD or higher must be used.

9. With HCD releases earlier than z/OS 1.10, you cannot directly access a multi-user access enabled V5 IODF. First you must remove the multi-user access capability from the IODF with z/OS 1.10 HCD or higher. In addition, you need to install the coexistence PTF for the corresponding z/OS release to prevent earlier HCD releases from accessing an IODF prepared for multi-user access (APAR OA22842).
Chapter 5. How to use the dialog

Overview
This information unit explains the following topics:
• Panel layout
• Working with lists
• Promptable fields
• Commands and function keys
• Getting help
• Navigating through the dialog
• Filtering
• Job statement information used in panels

Panel layout

Figure 26 explains the areas of an HCD panel. These areas appear in the same position on every panel unless you use ISPF to change the position of the command line. Not all areas are included on all panels.

Figure 26. Example of a Panel Layout

1. Action bar.

HCD provides an action bar-driven interface. You can select any of the action bar choices and display pull-down choices. Use F10=Actions to move the cursor to the action bar.
Title Line.

- Shows the panel title.
- Displays panel identifiers (on the left-hand side, if you have asked for them).
- Displays positional information for the work area:
  Row 1 of 3302 specifies that the first row of data that is displayed in the work area is the first row of 3302 available rows.
  More: > specifies that more data can be seen by moving the work area to the right using F20=Right. This can also be More: <, which means moving is possible to the left, or More: > <, which means moving is possible to the left and right.
  Instead of Row 1 of 3302, Filter Mode would be shown if the action list is filtered. For additional information, refer to “Filtering” on page 73.

Instruction Area.

Tells you how to proceed on the panel. On action list panels (see “Action lists” on page 64) you can get more instruction information by pressing the F13=Instruct key or by using the INSTRUCT command.

Work Area.

You can use the following facilities to control the work area:
- The Filter action reduces the number of objects in an object list; this is described further in “Filtering” on page 73.
- The F7=Backward and F8=Forward keys scroll the work area backward and forward. The end of the information is indicated by a line containing - end - or BOTTOM OF DATA.
- The F19=Left and F20=Right keys move the work area of an action list panel left and right (or RIGHT and LEFT command).
- The LOCATE command scrolls an object list so that a specific object (or the nearest lower match) is at the top of the work area. Search criteria is the object identifier shown in the leftmost data column.
- The TOP command scrolls backward to the top of the work area.
- The BOTTOM command scrolls forward to the bottom of the work area.

The following Figure 27 on page 61 shows the panel areas of a data entry panel.
This panel has the following characteristics:

- The position and length of empty entry fields are shown by underscore characters (_) and highlighting. If an entry field contains an entry, the underscore characters are not shown.
- Fields that you cannot change are preceded by a colon (:).
- Fields that you must complete may (depending on the terminal) be highlighted by a different color.
- Input fields with a plus sign (+) indicate that you can prompt information for this field using F4=Prompt. For more information on the prompt facility, refer to “Promptable fields” on page 67.

**Working with lists**

HCD presents most of the information in form of lists. Depending on the list, there are different methods to select an item from the list or to perform actions. The different types of lists are:

- Numbered selection lists
- Unnumbered single selection lists
- Unnumbered multiple selection lists
- Action lists
- Message lists

This section describes the different types of lists and the respective selection methods.

**Numbered selection lists**

To select an item from a numbered selection list, type the number you want to select in the input field (left of the first list item) and press the Enter key. An example of a numbered list is the HCD Primary Task Selection panel (see Figure 28 on page 62), displayed when you start an HCD session.
Unnumbered single selection lists

To select a list item from an unnumbered list from which you can select only one item, you can:

- Place the cursor in front of a list item and press the Enter key
- Select a list item with an / or a \ (slash) and press the Enter key.

Figure 29 on page 63 is an example of a list of IODFs from which you can select one IODF.

You can request a list like the one in Figure 29 on page 63 by pressing F4=Prompt while your cursor is on the I/O definition file entry field (Figure 28). For more details concerning prompting, refer to "Promptable fields" on page 67.
Note: For migrated IODFs (with indication MIGRAT in column Volume), HCD cannot determine the MUA-status and the size and therefore, columns MUA and Size remain blank in such cases.

Unnumbered multiple selection lists

To select a list item from an unnumbered list, from which you can select one or more list items, place a slash (/) in front of one or several list items and press the Enter key. In some cases, list items in unnumbered multiple selection lists are already preselected. You can overwrite this preselection.

To select a range of objects, you can place a left parenthesis ( in front of the first item in the range and a right parenthesis ) in front of the last item in the range. See also "Using the context menu" on page 64.

Figure 30 is an example of an unnumbered multiple selection list.

---

Chapter 5. How to use the dialog
## Action lists

Figure 32 on page 65 is an example of an action list panel. You see a panel like this one when you want to define, modify, or view I/O devices.

HCD uses the concept of object-to-action processing. That is, to work with an object, you first have to select the object, and then the action. For some actions, you do not have to explicitly select an object; for example, to add an object to an action list.

In an action list, you have several possibilities to perform an action on a list object:
- Using the context menu
- Using the action code
- Typing over data in the panel

### Using the context menu

To perform an action on one or several objects in an action list, you can use the context menu:

1. Select an object by placing a slash / in front of a list item.
   
   To select a range of objects, you can place a left parenthesis / in front of the first item in the range and a right parenthesis / in front of the last item in the range.

2. Press the Enter key. HCD displays a context menu showing all valid actions for the selected objects.

3. Select an action by entering the number or action code (letter in parentheses) in the entry field. After pressing the Enter key, the context menu is closed and the action is performed for the selected rows.

   If only one action is possible on an action list panel, the context menu is not shown, but the action is performed immediately after entering a / (slash) or an S.
In the example in Figure 31 on page 64, the following objects are changed:

- Objects 0001 through 01D4, and
- Object 01D6

A # marker in the selection column indicates that the row is disabled and not available for processing. For example, coupling facility devices are marked in such a way.

**Using the action code**

As you get familiar with the dialog, you might find it easier to select a list item and an action in one operation by means of an action code. To do this, enter the action code in the entry field to the left (the action column) of the associated list item; then press the Enter key. Figure 32 shows an example (for delete and for change).

You can select more than one list element and, if necessary, you can specify different action codes for each of the objects. You are first asked to confirm the deletions. Then the other actions (in the example above a change) are performed.

To get a list of action codes, place the cursor in the action column field and then press the F4=Prompt key. The same context menu is shown when using the in front of a list item and pressing the Enter key. For an example, see Figure 31 on page 64 or Appendix A. How to navigate through the dialog on page 371. If you press the F1=Help key instead of the F4=Prompt key, you get a list from which you can request explanation of the action codes.
**Action codes and selection markers:** You can also specify individual action codes within a marked range. If you do so, the following rules apply:

- When you press the Enter key, the single action codes are processed. The row selected with selection markers are not processed, they are still shown after processing the action codes.
- When you press the Enter key again, HCD shows the context menu for the rows selected with selection markers.

After successful operation, selection markers and action codes are removed from the list. If the operation is interrupted due to an error, selection markers and action codes not yet processed are still shown. You can remove them by pressing the F5=Reset key.

**Typing over existing data**

You can also make changes to editable fields by typing in new data or typing over existing data directly on the displayed panel.

You have to press the Enter key to process the changes. HCD then validates the data and displays the panel again. When you exit action list panels, all changes to the list since you last pressed the Enter key, are cancelled.

You can use the F5=Reset key to reset the values of all fields that you have changed since you last pressed the Enter key.

**Message lists**

One single operation can produce multiple messages. In this case, HCD displays a message list. You can then:

- Use the *Explain message* action from the context menu (or action code 4 ) to get an explanation of the message.
Use the **Delete message** action from the context menu (or action code \d\) to delete a message from the message list.

- Select **Save messages** from the **Save** action bar to save the displayed messages in the corresponding message log file.

**Promptable fields**

The HCD prompt facility reduces what you have to remember, what you have to type, and, possibly, what you have to correct due to typing errors. You can use the prompt facility if there is a plus sign (+) to the right of an entry field (or its column heading on an action list panel). Just place the cursor on the entry field and press the F4=Prompt key.

HCD then displays a prompt selection menu that lists all the values that are currently valid for the field. For long lists (of I/O devices, for example), HCD first displays a menu to let you limit the values listed (to only DASD devices, for example).

Prompt is also available for the action column. For this purpose you must place the cursor on the action column. HCD displays the same context menu as when using
the \slash (slash) in the action column and pressing the Enter key. For an example, see Figure 31 on page 64. Note that there is no ‘+’ sign shown for the action column heading.

To select a value, place the cursor to the left of (or on) that value or select the value with \slash or \slash (slash), then press the Enter key. HCD inserts that value into the entry field or, in case of selecting an action, performs that action immediately.

The prompt list is built dynamically; if a value can be used only once in a configuration then, after it has been selected, either it does not appear in the list again, or it is marked as nonselectable - with the hash sign (#).

Note: It is possible that HCD initially accepts a selection, but rejects it later after further validation when the entered context information is completed.

**Commands and function keys**

All HCD list panels have a command line on which you can enter the usual ISPF commands, TSO commands, or specific HCD commands.

You can use F22=Command to move the cursor to the command line. When the cursor is in the command line, F22=Command lets you step back through the commands that you have previously entered (including ISPF commands).

The HCD dialog uses 24 function keys, if your ISPF session allows the use of 24 function keys. The function keys assignments can be shown or hidden by entering PFSHOW in the command line.

To perform a function, just press that function key.

**Getting help**

HCD offers an extensive help facility. From any panel, you can get context-sensitive help by pressing the F1=Help key. Table 6 shortly explains the available types of help and how to obtain it.

**Table 6. Online Help Information**

<table>
<thead>
<tr>
<th>Type of Help</th>
<th>Description</th>
<th>How to Get Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>An explanation of what you can enter in a specific entry field on the current panel.</td>
<td>Place the cursor in the entry field and press the F1=Help key.</td>
</tr>
<tr>
<td>Extended</td>
<td>Gives information on the content and task of a function panel.</td>
<td>Move the cursor to a non-interactive field in the function panel and press F1, or select Extended help in the action bar, or press F2=Ex_help in field help panels.</td>
</tr>
<tr>
<td>Instruction</td>
<td>Specific instructions on what you can do on the current action list panel.</td>
<td>Press the F13=Instruct key. Or select Help from the action bar.</td>
</tr>
</tbody>
</table>
Table 6. Online Help Information (continued)

<table>
<thead>
<tr>
<th>Type of Help</th>
<th>Description</th>
<th>How to Get Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command</td>
<td>An explanation of the HCD commands that you can enter in the command line.</td>
<td>Place the cursor in the command line and press the F1=Help key. For detailed help on a specific command, type the name of the command and press the F1=Help key.</td>
</tr>
<tr>
<td>Keys</td>
<td>An explanation of the function keys.</td>
<td>Press the F1=Help key, then press the F9=Keyshelp key from the help panel. Or select Help from the action bar.</td>
</tr>
<tr>
<td>Reference Phrase</td>
<td>An explanation of any of the highlighted words or phrases that appear on a help panel.</td>
<td>Place the cursor on the word or phrase and press the Enter key.</td>
</tr>
<tr>
<td>Message</td>
<td>An explanation of a message that is displayed on the current panel.</td>
<td>Press the F1=Help key when a message is displayed, regardless of the cursor position.</td>
</tr>
<tr>
<td>Help for Help</td>
<td>A general explanation of how to use the help facility.</td>
<td>Press the F1=Help key, then press the F1=Help key again from the help panel. Or select Help from the action bar.</td>
</tr>
<tr>
<td>Action Bar</td>
<td>An explanation of the action bar of a panel.</td>
<td>Request extended help, then place the cursor on the reference phrase of the action you want help for, and press Enter.</td>
</tr>
</tbody>
</table>

On a help panel, you can use F5=Window to change the size of the window, to reduce the scrolling required, or to see more of the underlying panel.

Navigating through the dialog

Most tasks you do with HCD are done in one step. However, some tasks need more than one step (multi-step tasks). For example, when you define an I/O device, you usually want to define the processor and operating system data for that device at the same time. HCD lets you define this data together, in one task, by displaying a sequence of panels in a predefined order.

Moving forward and backward within a task

The following function keys are available to move forward and backward in a multi-step task.

Enter                  Displays the next panel in the predefined order. HCD validates your data, but does not save it in the IODF until you complete the task. At the end of the task, HCD redisplays the action list panel that precedes the task.

F3=Exit                Completes the task. HCD saves in the IODF all valid data that you have entered so far in the task (plus any data in the following steps that has been inherited by, for example, an Add like action). It then redisplays the action list.

F6=Previous            Available on the second and subsequent steps. It cancels the
current step and redisplay the previous panel. Any data that you have entered on the current panel is lost.

F12=Cancel Cancels the current task. Data entered in the current step is lost but, if you have entered data in any of the previous steps, HCD gives you the option to save that data (and any inherited data in the following steps) in the IODF.

Fast path

HCD offers you a fast path to jump directly from an action list panel to another panel without having to navigate through the dialog.

HCD offers you two possibilities to use this fast path:

- Using the **Goto** action bar choice
- Using the **GOTO** command

Using the action bar choice

Select the **Goto** action bar choice and on the resulting pull-down menu the target object you want to navigate to.
Objects marked with ... are associated with another object (for example, an EDT list is associated with an operating system). If you select such an object, HCD displays a list on which you can select the associated object:

Using the GOTO command

Specify GOTO (or just GO) and the target you want to jump to in the HCD command line. If you type GOTO without a target, a panel is displayed showing the same list of target objects as when using the Goto pull-down menu.

You can also jump to list panels that are associated with another object (for example, to the EDT list associated with an operating system). In this case, specify both the target and the object with which it is associated. If you omit this object, a selection list is displayed (for example, the Available Operating System list appears when going to the EDT list).

Optionally, you can also specify the object that will be shown as the first element of a list if you jump to a list panel.
The following figure shows you the syntax of the command. The parameters can be separated by either blank, comma, or dot.

**GOTO command:**

```
CHPID   processor ID
CONS    OS ID
CU       first list element
DEV      first list element
EDT      OS ID
EDT      first list element
ESO      OS ID
EDT      first list element
GEN      OS ID
EDT      first list element
MIG      OS
PART     processor ID
PORT     switch ID
PR       first list element
SW       first list element
SWCON    switch ID
MATRIX   switch ID
X         first list element
```

**CHPID**  Channel path list  **OS**  OS configuration list
**CONS**  Console list  **PART**  Partition list
**CU**  Control unit list  **PORT**  Port list
**DEV**  Device list  **PR**  Processor list
**EDT**  EDT list  **SW**  Switch list
**ESO**  Esoteric list  **SWCON**  Switch configuration
**GEN**  Generic list  **MATRIX**  Port matrix
**MIG**  Migrate configuration data  **X**  Leave HCD

**First list element**

The object that will be shown as the first element of a list if you jump to a list panel (same as if using the LOCATE command).
Examples: To go to the channel path list of processor SYSA and display CHPID 27 as the first element of the list, enter:
GO CHPID SYSA 27

To go to the esoterics list of EDT A1 in operating system MVSPROD and display the esoteric group ESO12 as the first element in the list, enter:
GO ESO MVSPROD A1 ESO12

To specify the processor ID and channel subsystem ID for XMP processors in GOTO PART or GOTO CHPID commands, you use the dot to concatenate both parts:
GO PART P2084.3

Filtering

If a displayed object list is too long, and scrolling back and forth is getting cumbersome, you may use the Filter function that lets you limit the displayed information. The Filter function is available when the action bar shows a Filter action bar choice. This is the case in a:
• Processor list
• Channel path list
• CTC connection list
• CF channel path connectivity list
• Switch list
• Port list
• Control unit list
• Device list
• Esoteric list
• I/O Path list

For example, on the device list, you can limit the displayed information by the following filter criteria:
• Device type
• Device group
• Whether the devices are defined to an operating system or not
• Serial number and description
• Volume serial number
• Device parameters and features (if you navigated to the I/O Device List via the operating system configuration list)
• Whether the devices are connected to a control unit or not
• Up to four control units, to which the devices may be connected
• Subchannel set ID

Rather than seeing all devices defined in the IODF, you see only those devices that are, for example, of a certain type connected to a certain control unit.

How to request filtering is illustrated in the following scenario:
1. Navigate to a processor’s I/O Device List. Note that for XMP processors you first need to navigate to its Channel Subsystem List and then to its I/O Device List. Now tab to the action bar and select Filter from the action bar. The pull-down menu shown in Figure 39 on page 74 is displayed:
2. Select **Set Filter**. This displays the Filter I/O Device List panel (Figure 40), tailored for the underlying I/O Device List, where you can specify your filter criteria. Note that for example, the ‘Subchannel set ID’ filter criteria is only available for the I/O Device List if invoked from the Channel Subsystem List of a z9 EC processor or later model, or from the Operating System Configuration List (action code /SF580000u/SF590000 in both cases). Similar panels are displayed for the other lists.

An alternative way to select filtering is to tab to the command line and type in FILTER SET.

Figure 39. Filter example

2. Select **Set Filter**. This displays the Filter I/O Device List panel (Figure 40), tailored for the underlying I/O Device List, where you can specify your filter criteria. Note that for example, the ‘Subchannel set ID’ filter criteria is only available for the I/O Device List if invoked from the Channel Subsystem List of a z9 EC processor or later model, or from the Operating System Configuration List (action code Y in both cases). Similar panels are displayed for the other lists.

An alternative way to select filtering is to tab to the command line and type in FILTER SET.

Figure 40. Filter I/O Device List

3. On this filter panel you can specify one or more filter criteria. All specified filter criteria must match to display the item. For example, if you specify a device type and a subchannel set where the devices of this type should be defined, and then press the Enter key, the I/O Device List is displayed again, now showing only those devices that match these filter criteria. Figure 41 on...
shows an example of a filtered list. If Filter Mode is displayed in the right top corner of the panel, it indicates that the filter mode is active.

Most of the entry fields support wildcards, that means that an asterisk (*) can be specified in front and/or after the specified term. The following table illustrates the wildcard processing using the device type field as an example.

<table>
<thead>
<tr>
<th>Specified Term</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>3380</td>
<td>Displays all devices of type 3380</td>
</tr>
<tr>
<td>3380*</td>
<td>Displays all devices of type 3380 regardless of its model</td>
</tr>
<tr>
<td>338*</td>
<td>Displays all devices whose type starts with '338'</td>
</tr>
<tr>
<td>*80</td>
<td>Displays all devices whose type ends with '80'</td>
</tr>
<tr>
<td><em>42</em></td>
<td>Displays all devices containing the string '42' within the type, for example 3420.</td>
</tr>
</tbody>
</table>

Use Field Help on the Filter panel to get information on whether wildcards are supported or not.

To remove the filter, use Clear Filter from the Filter action bar, or enter FILTER CLEAR in the command line.

To count the rows on a filtered list, use Count rows on (filtered) list from the Filter action bar choice, or enter FILTER NUM in the command line. An informational message displays the number of rows in the current list. If this list is filtered, only the rows matching the underlying filter criteria are counted.

Job statement information used in panels

Some of the HCD tasks, invoked from the dialog, generate batch jobs. These batch jobs use EXEC procedures as shown in Table 3 on page 34.

The first time you use a task that generates an HCD batch job, you must specify a JOB control statement for the job. This statement can contain user-specific information, such as an account number and SYSOUT class. For example:

```
//WASR   JOB (3259,BOX10),'SMITH',NOTIFY=WAS,
//CLASS=A,REGION=4M,MSGCLASS=F,MSGLEVEL=(1,1)
```
You need to specify the statement only once; it is saved in your user profile and reused for all subsequent HCD batch jobs that you request (in this and following HCD sessions) until you change it. HCD does not validate the JOB statement.

With JCL overwrite statements you can modify the EXEC procedures that are invoked by the job.

Notes:
1. A batch job requires a region size large enough to contain the HCD code (4M bytes) as well as the IODF (or two IODFs when you compare two IODFs).
2. A batch job to build an IOCDS must run on the processor on which the IOCDS is to be updated, except for processors configured in an S/390 microprocessor cluster.
3. A batch job generated by HCD cannot run when the HCD dialog has exclusive access to the same IODF that the batch job uses. You can release the IODF by changing to another IODF or by ending the HCD session.
4. To use the HCD profile options (e.g. UIM_LIBNAME=*) for a batch job that is started from the dialog, you need to allocate the HCD profile data set to xx.HCDPROF, where xx is the job step name.
5. In a sysplex environment, use the JOBPARM parameter to specify in which system of the sysplex you want the batch job to run. For additional information, see Chapter 12, “How to invoke HCD batch utility functions,” on page 307.

Table 3 on page 34 lists the HCD tasks that use batch jobs. It also lists their job step names, and EXEC procedure names.
Chapter 6. How to define, modify, or view a configuration

Overview

This information unit describes:

- The possibilities for creating new objects
- The navigation methods through HCD
- A suggested sequence to define a configuration
- Step-by-Step instructions on how to work with (for example, define, change, prime, delete):
  - Operating system configurations
  - EDTs
  - Generics
  - Esoteric groups
  - Processors
  - Channel subsystems (for XMP processors only)
  - Partitions
  - Channel paths
  - Control units
  - Devices
  - Consoles
- The possibilities to view information about objects

The information on how to work with switches is described in Chapter 7, “How to work with switches,” on page 169.

Before using the dialog of HCD to define a hardware configuration, you should have a plan of what your configuration should look like, and what you have to do to accomplish that. Preferably, the requirements of your configuration should be established in a configuration plan. Refer to z/OS HCD Planning (for a z/OS, OS/390, or MVS configuration) and z/VM CP Planning and Administration (for a z/VM configuration) for a description of what needs to be considered when this plan is prepared.

Creating new objects

You have three possibilities to create new objects: add, add like, and repeat (copy).

Add

Use the F11=Add key to define a new object. Initially, the entry fields contain (where applicable) default values supplied by HCD.

Add like

Use the Add like action from the context menu to define a new object that is based on the definition of an existing object. You just have to enter the fields that are different. A field that needs a unique value, such as the object’s identifier, is not copied.
Repeat (copy)
The action Repeat (copy) from the context menu is similar to Add like, but the definitions of all related objects are also copied. For example, if you repeat an operating system configuration, HCD also copies the definitions for all EDTs and consoles, and the connections to all I/O devices attached to that operating system. You can repeat parts of a configuration within the same or to another IODF.

This function is useful when you want to consolidate configuration data from several IODFs into one single IODF or to repeat configuration data (for example, esoterics) that is used several times in an IODF.

When copying parts of a configuration, the source data and the target data are merged.

- If the source object does not exist in the target IODF
  The new object is defined using the attributes you specified while copying the object. The objects and connections that are related to the object you want to copy are created with the same attributes as the source objects and connections.

- If the source object already exists in the target IODF
  The attributes of the target object are updated according to the attributes of the source object. If related objects or connections do not yet exist, they are created. If they already exist, their attributes are updated according to the attributes of the source.

Navigating through HCD

HCD offers three methods to navigate to objects:
- Centralized navigation
- Hierarchical navigation
- Graphical navigation

Centralized navigation

Centralized navigation means that you always navigate to objects starting from the Define, Modify, or View Configuration Data panel.

The tasks described in this section use this navigation method.

Hierarchical navigation

Hierarchical navigation means that you navigate to objects from top to bottom.

In Figure 42 on page 80 you start, for example, with option 3 to open the Processor List. From the Processor List, you can navigate to channel paths, from channel paths to control units, and finally from the control unit list to the device list. This device list - called Device List (attached) in Figure 42 on page 80 - is different from the device list you reach with option 5 from the Define, Modify, or View Configuration panel:

- The devices on this list are limited to the control unit you selected on the preceding control unit list.
- The actions offered on this list differ from the actions available on the device list reached with option 5 from the Define, Modify, or View Configuration panel.
Hierarchical navigation is useful, for example, when you want to use the **Attribute group change** action necessary to change the DYNAMIC parameter for a group of devices. In this case, you have to navigate to the device list via the operating system configuration list.

**Graphical navigation**

Graphical navigation means that you navigate to objects by viewing a graphical configuration report and jumping to the object lists using the F4=Jump function. Refer to “Create or view graphical configuration reports” on page 238 for information on how to use the graphical configuration report.

The graphical navigation is useful when you prefer a graphical representation of your configuration to navigate from object to object.

**Navigation map**

**Figure 42 on page 80** illustrates how you can navigate from object to object. You can either navigate to objects using the **Work with object** actions from the context menu or using the appropriate action code, for example `S`. Note that you can reach the Generic List with two action codes: `G` (ordered by name) or `P` (ordered by preference value). For information on how to use action codes refer to “Using the action code” on page 65.

**Control Unit list and Device list**

You reach these lists with option 4 and 5 from the Define, Modify, or View Configuration Data panel. These lists show all devices defined in the IODF.

The Control Unit list (attached) and the Device list (attached) can only be reached from an object higher in the hierarchy. These lists are limited to the object higher in the hierarchy and the actions available on these lists differ from the actions available on the device and control unit lists you reach with option 4 and 5 from the Define, Modify, or View Configuration panel (refer to “Hierarchical navigation” on page 78). You can, for example, limit a control unit list to the control units attached to one specific channel path by opening the list from the channel path list.
Suggested sequence to define a configuration

A hardware configuration consists of:

- Information needed by z/OS, OS/390, or MVS/ESA, and by z/VM. This is known as the *operating system configuration* or *OS configuration*.
- Information needed by the channel subsystem (CSS). This defines all the hardware resources (such as control units, channel paths, and I/O devices) and how they are connected.
- In some cases, your configuration contains information needed by the switch. For definitions and modifications of switches, refer to Chapter 7, “How to work with switches,” on page 169.

You can define the objects of a configuration in almost any order but at one point you have to connect objects together. You can only connect objects that are already...
defined; therefore it is useful to define the objects in a logical order. For example, when defining I/O devices during the hardware definition, you are prompted to add devices to existing operating system definitions. Therefore, it is useful to define the operating system before the devices.

The suggested sequence to define a configuration is:

1. Operating systems
2. EDTs (MVS-type only)
3. Esoterics (MVS-type only)
4. Switches (explained in Chapter 7, “How to work with switches,” on page 169)
5. Processors
6. Channel subsystems (for XMP processors)
7. Partitions (if processor in LPAR mode)
8. Channel paths
9. Control units
10. Devices
11. Consoles

Working with operating system configurations

An operating system (OS) configuration defines the data that is used by z/OS, OS/390, MVS/ESA, or z/VM to build its control blocks. An IODF can contain more than one OS configuration; MVS/ESA, z/OS, or OS/390 is told which one to use at IPL time.

Defining operating system configurations

It is recommended to define the operating system configuration before you define anything else. Proceed as follows to define an operating system configuration:

1. On the Primary Task Selection panel, select Define, modify, or view configuration data and on the resulting panel the object Operating system configurations. HCD displays the Operating System Configuration List of all operating system configurations currently defined in the IODF:

```
<table>
<thead>
<tr>
<th>Select type of config.</th>
<th>OS Conf. List</th>
<th>F11</th>
<th>Add OS</th>
<th>Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Operating systems</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. EDTs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Esoterics</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Switches</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Processors</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Channel subsystems</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Partitions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. Channel paths</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9. Control units</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10. Devices</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11. Consoles</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Select one or more operating system configurations, then press Enter. To add, use F11.

```
/ Config. ID  Type  Description
  _ OPSYS02  VM  z/VM operating system
******************************************************************************
```

Chapter 6. How to define, modify, or view a configuration  81
If there are no existing configurations in the IODF, the operating system configuration list is empty.

2. Use F11=Add to define a new configuration. The data-entry fields are shown in the following figure, with sample data:

```
Add Operating System Configuration

Specify or revise the following values.
OS configuration ID ..... OPSYS01_
Operating system type ....M V S +
Description ............ z/OS operating system
```

3. After you press the Enter key, HCD displays the updated operating system configuration list.

**Changing operating system configurations**

You can change the description of an operating system by just typing over the Description column or by using the Change action from the context menu (or action code /SF580000c/SF590000) on the Operating System Configuration List.

**Changing the operating system configuration ID**

To change the ID of an operating system, perform the following steps:
1. On the OS Configuration List select the operating system and the *Repeat (copy)* OS configurations action from the context menu (or action code R). The Identify Target IODF panel is displayed.

2. Press the Enter key to accept the default target IODF name, that is the IODF you are currently working with. The Repeat Operating System Configuration panel is displayed.

3. Specify the new identifier for the operating system and press the Enter key. HCD displays the OS Configuration List now showing the new operating system.

4. Delete the old operating system by selecting the operating system and the *Delete* action from the context menu (or action code D). HCD displays a confirmation panel before showing the updated OS Configuration List.

**Repeating (copying) operating system configurations**

You can copy operating systems within the same or to another IODF. When copying an operating system, the following related objects and connections are also copied:

- Devices defined for the operating system
- List of consoles (NIPCONs for MVS)
- EDTs including their esoterics and generic groups

In the following example, you copy an operating system to another IODF that already contains an operating system with the same ID that you specify in the repeat panel.

1. Make sure that the operating system in the target IODF has the same operating system type as the one in the source IODF.

2. On the Operating System Configuration List, copy the operating system using the *Repeat (copy)* OS configurations action from the context menu (or action code R). The Identify Target IODF panel is displayed.

3. Specify the IODF to which the selected operating system is to be copied. The default IODF is the IODF you are currently working with.

4. On the following Repeat Operating System Configuration panel, specify the required values and press the Enter key.
If a device defined to the operating system already exists in the target IODF, HCD tries to map the device. If more than one device with the same type and number exist, HCD maps the device to the first device found. To avoid this sometimes erroneous mapping, specify a processor and partition on the Repeat Operating System Configuration panel in which the operating system is to be run. Refer to “Migrating additional MVSCP or HCPRIO input data sets” on page 267 for detailed rules when a device is mapped.

Console devices (NIPCONs for MVS) from the source operating system are copied to the beginning of the target’s console chain.

5. Because you copy an operating system that already exists in the target IODF, HCD displays a panel to confirm the merging of configuration data.

Deleting operating system configurations

You can delete the complete definition of an operating system using the Delete action from the context menu (or action code $S$F580000d/SF590000$) on the Operating System Configuration List. This also deletes all EDTs, esoterics, consoles, and connections to devices defined for this operating system.

Working with EDTs

For an MVS-type operating system, you have to define at least one eligible device table (EDT). An EDT can consist of one or more esoteric device groups and names of the generic device types. Esoteric device groups are installation-defined groupings of I/O devices.

An OS configuration can contain more than one EDT; z/OS or OS/390 is told which one to use at IPL time. For background information about I/O device allocation in MVS that you need to understand before defining EDTs and esoteric groups, refer to z/OS HCD Planning.

Defining EDTs

Before you can define EDTs, you must have defined an operating system. You define an EDT as follows:

1. On the Primary Task Selection panel, select Define, modify, or view configuration data and on the resulting panel the object Operating system
configurations. HCD displays the Operating System Configuration List of all operating system configurations currently defined in the IODF.

2. On the Operating System Configuration List, select the OS configuration and the Work with EDTs action from the context menu (or action code 5). HCD displays the EDT List.

<table>
<thead>
<tr>
<th>Command</th>
<th>Scroll</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goto Backup Query Help</td>
<td>PAGE</td>
</tr>
</tbody>
</table>

Select one or more EDTs, then press Enter. To add, use F11.

- Configuration ID : OPSYS01 MVS or z/OS operating system
- EDT Last Update By AS 1994-10-04
- Description basic

If there are no EDTs defined in the IODF, the EDT list is empty.

3. Use F11=Add to add a new EDT. The data-entry fields are shown in the following figure, with sample data:

| Configuration ID : OPSYS01 MVS or z/OS operating system |
| EDT identifier . . . A1 |
| Description . . . special |

4. After you press the Enter key HCD displays the updated EDT List.

**Changing EDTs**

You can change the description of an EDT by just typing over the Description column or using the Change action from the context menu (or action code C) on the EDT List.
Changing the EDT ID

To change the ID of an EDT, perform the following steps:

1. On the EDT List select the EDT and the Repeat (copy) EDTs action from the context menu (or action code /SF580000r/SF590000). The Identify Target IODF panel is displayed.

2. Press the Enter key to accept the default target IODF name, that is the IODF you are currently working with. The Repeat EDT panel is displayed.

3. Specify the new identifier for the EDT and press the Enter key. HCD displays the EDT List now showing the new EDT.

4. Delete the old EDT by selecting the EDT and the Delete action from the context menu (or action code /SF580000d/SF590000). HCD displays a confirmation panel before showing the updated EDT List.

Repeating (copying) EDTs

You can copy EDTs within the same or to another IODF. When copying an EDT, the esoteric groups and the VIO eligible parameter are also copied. Perform the following steps to repeat an EDT:

1. Make sure that the devices are already defined to the target operating system.

2. On the EDT List, select an EDT and the Repeat (copy) EDTs action from the context menu (or action code /SF580000r/SF590000). The Identify Target IODF panel is displayed.

3. Specify the IODF to which the selected operating system is to be copied. The default IODF is the IODF you are currently working with.
4. On the following Repeat EDT panel, specify the required values and press the Enter key.

If the EDT already exists in the target IODF, the esoteric groups and their devices are merged. In this case, HCD displays a panel to confirm the merging of data.

Deleting EDTs

You can delete the definition of an EDT using the Delete action from the context menu (or action code \d\ ) on the EDT List. This also deletes the esoterics.

Working with generics

Device types with similar characteristics are logically grouped together and assigned a name to by the system. Such a group is called a generic device type. Reference to a generic device type is made by its name. To request a device allocation, a user can specify a generic device type rather than a specific device number. z/OS or OS/390 then allocates a device from the specified generic device type.

1. On the Primary Task Selection panel, select Define, modify, or view configuration data and on the resulting panel the object Operating system configurations. HCD displays the OS Configuration List of all operating system configurations currently defined in the IODF.

2. On the Operating System Configuration List, select the OS configuration and the Work with EDTs action from the context menu (or action code \s\ ). HCD displays the EDT List.

3. To change the generics, select either the action Work with generics by name from the context menu (or action code \g\ ), or the action Work with generics by preference value (or action code \p\ ) on the EDT List. On the resulting panels you can then change the VIO indicator, the preference value for a generic, and you can display a subsequent panel that lists the devices belonging to the specific generic.
Working with esoteric groups

An esoteric device group identifies the I/O devices that are included in that group. The name you assign to an esoteric device group is called the esoteric name. To request allocation of a device from an esoteric device group, specify the esoteric name on the UNIT parameter of a JCL DD statement. The name esoteric device group is often shortened to esoteric group or simply esoteric.

Defining esoteric groups

You can define which esoteric device groups are in each EDT after you have defined the OS configuration. But you cannot assign I/O devices to an esoteric device group until the devices have been defined.

1. On the Primary Task Selection panel, select Define, modify, or view configuration data and on the resulting panel the object Operating system configurations. HCD displays the OS Configuration List of all operating system configurations currently defined in the IODF.

2. On the Operating System Configuration List, select the OS configuration and the Work with EDTs action from the context menu (or action code /SF580000s/SF590000). HCD displays the EDT List.

3. On the EDT List, select the EDT and the Work with esoterics action from the context menu (or action code /SF580000s/SF590000). HCD displays the Esoteric List.

4. Use F11=Add to add a new esoteric group. The data-entry fields are shown below, with sample data:
5. After you press the Enter key HCD displays the updated Esoteric List.

**Assigning devices to esoterics**
You must define the I/O devices before you can assign them to an esoteric group. This is described in “Defining devices” on page 138. The State column on the Esoteric List indicates the esoteric groups that have no devices defined; so you can check later that your groups are properly defined.

If I/O devices are already defined, you can assign them to esoteric groups as described in “Adding devices to esoterics.”

**Changing esoteric groups**

You can change the following data of an esoteric group by just typing over the corresponding columns or by using the **Change** action from the context menu (or action code `SF580000c/SF590000`) on the Esoteric List:

- Esoteric name
- VI0 eligible parameter
- Token

**Adding devices to esoterics**
You have two possibilities to add devices to esoterics:

**While defining devices**
When you define a device, HCD automatically prompts you to define the device to an operating system, then to an EDT and esoterics. Refer to “Defining devices” on page 138 for a step-by-step instruction on how to do that.

**While modifying esoterics**

You can add existing devices to esoterics at any time from the Esoteric List as follows:

1. On the Esoteric List, select the esoteric and the **Assign devices** action from the context menu (or action code `SF580000s/SF590000`). HCD displays the Assign/Unassign Devices to Esoterics panel.
2. On the Assign/Unassign Devices to Esoterics panel, overwrite the values in the Assigned column to assign (YES) or unassign (NO) devices to the esoterics of the selected row.

If you do not want to assign a complete group of devices, you can limit the range by specifying a starting number and the number of devices. If you omit the number of devices, 1 is assumed.

```
<table>
<thead>
<tr>
<th>Devices</th>
<th>Device Type</th>
<th>Generic Name</th>
<th>Assigned</th>
<th>Starting Number</th>
<th>Number of Devices</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001,1</td>
<td>3278-3</td>
<td>3277-2</td>
<td>Yes</td>
<td>____</td>
<td>____</td>
</tr>
<tr>
<td>0006,1</td>
<td>3278-3</td>
<td>3277-2</td>
<td>No</td>
<td>____</td>
<td>____</td>
</tr>
<tr>
<td>00C1,1</td>
<td>3480</td>
<td>3480</td>
<td>No</td>
<td>____</td>
<td>____</td>
</tr>
<tr>
<td>01D1,8</td>
<td>3390</td>
<td>3390</td>
<td>No</td>
<td>____</td>
<td>____</td>
</tr>
</tbody>
</table>
```

3. Press the Enter key to process the changes. Then press the F3=Exit key to return to the Esoteric List.

**Repeating (copying) esoteric groups**

You can copy esoterics within the same or to another IODF. When copying an esoteric, the list of assigned devices is also copied.

Perform the following steps to repeat esoterics:

1. Make sure that the devices are already defined to the target operating system.
2. On the Esoteric List select an esoteric and the Repeat (copy) esoterics action from the context menu (or action code R). The Identify Target IODF panel is displayed.
3. Specify the IODF to which the selected operating system is to be copied. The default IODF is the IODF you are currently working with.
4. On the following Repeat Esoteric panel, specify the required values and press the Enter key.
   
   If the esoteric already exists in the target IODF, the devices defined for the esoteric are merged. In this case, HCD displays a panel to confirm the merging.
Deleting esoteric groups

You can delete the definition of an esoteric using the Delete action from the context menu (or action code d) on the Esoteric List.

Working with processors

A note on terminology:

Throughout this document, the following terms are used:

**XMP processor and SMP processor**

The term XMP processor designates processors that support multiple logical channel subsystems (LCSS). It is used in contrast to the term SMP processor, which designates processors of previous generations that support only one channel subsystem.

For XMP processors, the HCD dialog offers methods to explicitly define multiple logical channel subsystems (MCSS). For SMP processors, the single channel subsystem is implicitly defined with the processor.

You can define more than one processor in an IODF and for each defined processor you can configure processor-related data for further use by the CSS.

HCD allows you to define and control I/O configurations for a local as well as for all other processors that are part of an S/390 microprocessor cluster.

For processors that are physically partitioned, you must define each physical partition as an individual processor.

Defining processors

Define a processor as follows:
1. On the Primary Task Selection panel, select Define, modify, or view configuration data and on the resulting panel the object Processors. HCD displays the Processor List of all processors currently defined in the IODF.
2. Use F11=Add to add a new processor. The data-entry fields are shown in the following figure, with sample data:

```
Figure 43. Add a processor

On the Add Processor panel, you can specify the network name and the CPC name, when the processor is configured in an S/390 microprocessor cluster. If you specify a SNA address, refer for specific access authority to "Security-related considerations" on page 339. For XMP processors, you may specify a local system name. If you do not enter a name, and a CPC name is given, the local system name defaults to the CPC name. For SMP processors you must not specify a local system name.

Use Prompt on the Add Processor panel for the SNA addresses for those CPCs that are currently configured in the S/390 microprocessor cluster.

3. Depending on the processor type/model, there may be more than one support level for the processor type. The support level defines the supported channel path types, and the features such as CF duplexing and cascaded FICON switches. If the processor has several support levels, HCD displays another panel showing a list of available support levels for the processor.
Select the appropriate support level. HCD uses this level when validating the configuration for this processor. It relates to the installed microcode.

**Note:** On the Available Support Levels panel you can retrieve an explanation of the processor support level for zSeries processors. Position the cursor on the support level description and press PF1 to get an enumeration of functions provided by this support level.

4. After you press the Enter key HCD displays the updated Processor List.
   You can now use the F20=Right key to scroll to the right to see the SNA address, if you have defined one.

### Changing processors

You can change the following data of the processor:
- Type
- Model
- Configuration Mode
- Serial Number
- Description
- Network name
- CPC name

1. On the Processor List, select the processor and apply the *Change* action from the context menu (or action code `[c]`). The Change Processor Definition panel is displayed.
2. To change the processor type, or model, overtype the old processor type, or model values, and press the Enter key. To change the support level for the same processor type, press the Enter key. If you have installed a new processor type or model, or a new processor support level, you need to upgrade the processor definition within HCD. No dynamic change is possible. HCD selects the proper configuration rules that are dependent on the processor type, and support level, and generates the correct input for the IOCDS download process.

Note: If the processor change leads to error message CBDA102I, make sure that the new processor type/model supports the same configuration rules as the old processor type, for example, that the same channel path types are supported. Subsequent messages indicate an invalid support level. You first have to update your configuration according to the new processor type/model before you can change the processor.

3. If more than one support level is available for the processor type, the Available Support Levels panel is displayed where you can select the correct support level for your processor.

Note: On the Available Support Levels panel you can retrieve an explanation of the processor support level for zSeries processors: Position the cursor on the processor support level description and press PF1 to get an enumeration of functions provided by this support level.

4. The Update Channel Path Identifiers panel is displayed. This panel shows the old channel path definitions, which you can change according to your new processor configuration.
The column New CHPID shows the channel path IDs to which the values of column CHPID are mapped by default.

- You can overtype the values of the column New CHPID. Leave the column Until CHPID blank. Press the Enter key to move the old channel path IDs to the new channel path IDs.

- If you enter a value in the column Until CHPID, you move all defined CHPIDs in the range defined by the channel paths specified in the CHPID and Until CHPID column to the new range that starts with the value in the New CHPID column. If the values for the CHPID column have a ‘gap’, the ‘gap’ is also reflected in the New CHPID column range. When you entered a value in the Until CHPID column, press the Enter key. The Update Channel Path Identifiers panel is redisplayed where the new range is resolved in the New CHPID. The Until CHPID column is shown as blank. To process the updates, press the Enter key again.

**Note:** If the processor upgrade changed the SYSTEM value of the corresponding IOCPS configuration, it may not be possible to do an IOCDS download. It is only possible to perform an IOCDS download for specific processor types because IOCPS validates the generated IOCPS statements according to the configuration rules of the executing processor. See “[Supported Hardware Report](#)” on page 401 for the processor type which allows you to download an IOCDS in preparation for a processor upgrade.

If a processor type change leads to change of the SYSTEM value of the corresponding IOCPS configuration see “[Build an IOCDS](#)” on page 201 and “[Build S/390 microprocessor IOCDSs](#)” on page 203.
Changing the processor ID

To change the ID of a processor, perform the following steps:

1. On the Processor List, select the processor and the Repeat (copy) processor configurations action from the context menu (or action code /SF580000r/SF590000). The Identify Target IODF panel is displayed.

2. Press the Enter key to accept the default target IODF name, that is the IODF you are currently working with. The Repeat Processor panel is displayed.

3. Specify the new identifier for the processor and press the Enter key. HCD creates a new processor with the same characteristics and connections as the old one.

4. On the resulting Processor List, delete the old processor by selecting the processor and selecting the Delete action from the context menu (or action code /SF580000d/SF590000). HCD displays a confirmation panel before showing the updated Processor List.

Repeating (copying) processors

You can copy processors within the same or to another IODF. When copying a processor, the following related objects and connections are also copied:
- Channel subsystems (for XMP processors only)
- Partitions
- Channel paths
- Control units
- Devices

When you copy a processor within the same IODF, you create a new processor. When you copy a processor into a different IODF, you can either create a new
processor with the Repeat (copy) processor configuration action, or, with the same action, merge the configuration of the source processor into an existing configuration of the target processor.

Before copying the processor to an existing processor in another IODF (merge the processor configurations), check the following:

1. Make sure that source and target processor have the same type-model and support level.
2. Check the partition usage type if the processor has several partitions.
   You cannot replace a partition that has a different usage type (except if there are no channel paths defined for this partition):

<table>
<thead>
<tr>
<th>Usage Type in Source IODF</th>
<th>Matching</th>
<th>Usage Type in Target IODF</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS</td>
<td></td>
<td>OS</td>
</tr>
<tr>
<td>CF</td>
<td></td>
<td>CF</td>
</tr>
<tr>
<td>CF/OS</td>
<td></td>
<td>CF, OS, or CF/OS</td>
</tr>
</tbody>
</table>

If the partition usage types do not match, change them in the target IODF.

3. If a channel path already exists in the target IODF, make sure that it has the same type.

4. If the target channel path connects to another dynamic switch than the source channel path, disconnect the channel in the target IODF.

5. To replace a shared channel path mode by a dedicated or reconfigurable channel path mode, change the mode or delete the incompatible channel path in the target IODF.

Copy a processor as follows:

1. Copy the processor using the Repeat (copy) processor configurations action from the context menu (or action code \[\text{f}^7\]) on the Processor List. The Identify Target IODF panel is displayed.

2. Specify the IODF to which the selected processor configuration is to be copied. The default IODF is the IODF you are currently working with.

3. On the following Repeat Processor panel, specify the required values and press the Enter key.
   Definitions for source partitions are merged with the definitions of the target partitions. If a target partition has another partition number than the source partition, HCD keeps the target partition number.
   If control units, devices, or CHPIDs already exist in the target IODF (same number and type), HCD tries to map them. Migrating additional IOCP input data sets on page 265 explains in detail when a device or control unit is mapped. The general rule is, that target definitions are updated from the source definitions.
   See Explicit device candidate lists on page 110 on how to handle explicit device candidate lists.
Copying an SMP processor to an XMP channel subsystem

You can copy the configuration of an SMP processor into an XMP channel subsystem within the same or to another IODF. When copying an SMP processor to a CSS, the following related objects and connections are also copied:

- Partitions
- Channel paths
- Control units
- Devices

You can use this function for the following alternative tasks:

1. **alternative 1** merge the source processor with an existing CSS in the target processor
2. **alternative 2** copy the source processor to a new CSS in the target processor

Copy an SMP processor to a channel subsystem as follows:

- On the Processor List, select an SMP processor and the action **Copy to channel subsystem** \( \ldots \) (SMP) from the context menu (or action code \( \text{V} \)).
- Specify the IODF to which the selected operating system is to be copied. The default IODF is the IODF you are currently working with.
- On the following panel, specify your target processor ID and channel subsystem ID. Make sure that the target processor supports multiple logical channel subsystems.

After pressing the Enter key:
- **for alternative 1** you are prompted to confirm or cancel the merging of the source processor into the existing target CSS. A message will inform you about the success of the operation.
- **for alternative 2** on the Add Channel Subsystem panel, you are prompted to define the ID for the new CSS and the maximum number of allowed devices.
Add Channel Subsystem

Specify or revise the following values.

Processor ID . . . . . . : TSPROC1    Proc supporting multiple SSs

Channel Subsystem ID . . . 3 +
Description . . . . . . . CSS 3 of TSROC1____________

Maximum number of devices
  in subchannel set 0 . . 64512 +
  in subchannel set 1 . . 10000 +

If, by chance, partition names from the source processor already exist in the target processor, you are prompted to specify new names for those partitions.

**Priming processor data**

You can prime your I/O configuration in a work IODF with the processor serial number for the active processor.

To prime, select the action *Prime serial number* from the context menu (or action code [SF580000i/SF590000]) on the Processor List.

The Confirm Priming Processor List shows the selected processors with the sensed data for the processor type and serial number of the active processor and their corresponding definitions in the IODF. If the processor type of the active processor and the defined processor match, they are shown in the Confirm Priming Processor List.

The sensed data for the processor serial numbers can be confirmed before being taken into the IODF. If a value is blanked out, the defined IODF value is not changed. If you use the F12=Cancel key, none of the sensed values is taken.
Deleting processors

You can delete the definition of a processor using the Delete action from the context menu (or action code /SF580000d/SF590000) on the Processor List. If you delete a processor, all channel paths, partitions, and connections to control units and devices for that processor are also deleted; the control units and devices are not deleted.

Working with channel subsystems

**Note:**
All tasks described in this unit’s subsections are only available for XMP processors.

For XMP processors, you may define multiple logical channel subsystems (up to n), identified by a unique 1-digit hexadecimal number (range 0 through n-1). For each channel subsystem, then, you may define its own set of partitions and CHPIDs.

Defining channel subsystems

Define a channel subsystem as follows:
- On the Processor List, select an XMP processor and the action *Work with channel subsystems . . (XMP)* from the context menu (or action code 5). HCD displays the Channel Subsystem List.
Select one or more channel subsystems, then press Enter. To add, use F11.

Processor ID . . . . : TSPROC1 Proc supporting multiple SSs

<table>
<thead>
<tr>
<th>CSS Devices in SS0</th>
<th>Devices in SS1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Maximum + Actual</td>
</tr>
<tr>
<td>0</td>
<td>64512</td>
</tr>
<tr>
<td>1</td>
<td>64512</td>
</tr>
<tr>
<td>2</td>
<td>64512</td>
</tr>
</tbody>
</table>

******************************* Bottom of data *******************************

Use F11=Add to define a new channel subsystem. The data-entry fields are shown in the following figure, with sample data:

Add Channel Subsystem

Specify or revise the following values.

Processor ID . . . . . : TSPROC1 Proc supporting multiple SSs

Channel Subsystem ID . . . 3 +

Description . . . . . CSS 3 of TSPROC1

Maximum number of devices

in subchannel set 0 . . 64512 +

in subchannel set 1 . . 10000 +

F1=Help F2=Split F3=Exit F4=Prompt F5=Reset

F9=Swap F12=Cancel

Define the ID for the new CSS and the maximum number of allowed devices. For z9 EC processors or later, you can specify the maximum number of devices for two subchannel sets.

Pressing the Enter key brings you back to the Channel Subsystem List.

Repeating (copying) channel subsystems

You can copy channel subsystems within the same or to another IODF:

1. merge the source CSS with an existing target CSS in the same processor
2. copy a source CSS to a new target CSS in the same processor
3. copy a source CSS to another XMP processor

Copy or repeat a channel subsystem as follows:

• On the Processor List, select an XMP processor and the action Work with channel subsystems . . (XMP) from the context menu (or action code 5).
- On the Channel Subsystem List, for the source CSS, select action Repeat (Copy) channel subsystem from the context menu (or action code X).
- Specify the IODF to which the selected operating system is to be copied. The default IODF is the IODF you are currently working with.
- On the following panel, specify your target processor and CSS:

  Repeat Channel Subsystem

  Specify or revise the following values.
  Processor ID ...... XMPP01 +
  Channel subsystem ID .. 2 +

- Depending on what you want to do, continue as follows:
  - To merge source and target CSS, you are prompted for confirmation for merging the source CSS data into the target CSS data. If you confirm, you are prompted to specify new partition names, because the partitions within an XMP processor must be unique:

    Specify New Partition Names
    Command ===> ____________________________________________
    The partitions listed have already been defined in the target processor. Specify new names and press ENTER.
    Partition New Name
    LPAR01 ________
    LPAR02 ________
    **************************************** Bottom of data ****************************************

    A message indicates the successful processing.
  - To copy the source CSS into a new target CSS, you can specify or revise certain characteristics of the target CSS:

    Add Channel Subsystem

    Specify or revise the following values.
    Processor ID ........ TSPROC1 Proc supporting multiple SSs
    Channel subsystem ID .... 3 +
    Description ......... CSS 3 of TSPROC1
    Maximum number of devices
    in subchannel set 0 .. 64512 +
    in subchannel set 1 .. 10000 +

    You are prompted to specify new partition names. Also, a message indicates the successful processing.
  - To copy a source CSS to another XMP processor, the processing is identical as copying into a new target CSS. However, specifying new partition names is only necessary if the partition names of the source processor already exist in the target processor.

  **Note:** PCHID values are not copied.
Copying/repeating channel subsystems with CTC connections

When copying channel subsystems or partitions within the same IODF, valid CTC connections in the source are not automatically copied and changed in the target configuration. Instead, HCD displays the *CTC Connection Update List*. This list shows all valid CTC connections of the source configuration that need an update before being copied to the target. From this list, you can select those connections that you want to copy. For each selected CTC connection, HCD updates the CUADD definition such that the valid CTC connection is moved from the source to the target. For FCTC control units, HCD generates the full-byte CUADD value for target XMP processors, consisting of the concatenation of CSS and MIFID.

![CTC Connection Update List](image)

**Figure 45. CTC Connection Update List**

### Copying a channel subsystem to an SMP processor

HCD offers actions to copy a channel subsystem to an SMP processor or merge the CSS to an existing SMP processor configuration. When copying a channel subsystem to an SMP processor, the following related objects and connections are also copied:

- Partitions
- Channel paths
- Control units
- Devices

1. Use action *Work with channel subsystems . . (XMP)* (or action code `s`) for an XMP processor which brings you to its Channel Subsystem List.
2. Now you select action **Copy to processor** (or action code \( \checkmark \)) for the CSS that you want to copy.

3. The Identify Target IODF panel is displayed. Specify the IODF where the target processor is defined. The default IODF is the IODF you are currently working with. Press the Enter key. The Copy to Processor panel is displayed.

![Copy to Processor panel](image)

4. Specify your target processor and press the Enter key.

   **Note:** If the target processor does not yet exist, HCD invokes the Add Processor dialog. If the target processor already exists, you must confirm that you want to merge the configuration data. Anyway, make sure that the target processor does not support multiple logical channel subsystems.

5. Before returning to the Channel Subsystem List, a message will inform you about the success of the action.

### Changing channel subsystems

You can change the following characteristics of a channel subsystem:

- Description
- Maximum number of devices per subchannel set

To perform this task, proceed as follows:

1. On the Processor List, select an XMP processor and the action **Work with channel subsystems . . (XMP)** from the context menu (or action code \( \text{a} \)).

2. On the Channel Subsystem List, for the CSS you want to change, select action **Change** from the context menu (or action code \( \text{c} \)).

3. The Change Channel Subsystem panel is displayed, where you can specify your changes. Then press the Enter key. This returns you to the Channel Subsystem List.
### Change Channel Subsystem

Specify or revise the following values.

- **Processor ID** : TSPROC1 Proc supporting multiple SSs
- **Channel subsystem ID** : 0
- **Description** : CSS 0 of TSPROC1

**Maximum number of devices**
- in subchannel set 0 : 64512 +
- in subchannel set 1 : 15000 +

F1=Help F2=Split F3=Exit F4=Prompt F5=Reset F9=Swap F12=Cancel

---

**Changing the channel subsystem ID**

To change a CSS ID is only possible via the deviation of repeating (copying) the channel subsystem with a new ID and then delete the source CSS. If you cannot repeat the channel subsystem in the same processor, because all available IDs are occupied, and you want to exchange the IDs of two existing CSSs, then you need to copy both CSSs into a different target processor, delete them in the source processor and copy them back to the source processor with the exchanged IDs.

It may be necessary to restore certain definitions afterwards, for example, coupling facility connections get lost during the copy process.

**Deleting channel subsystems**

You can delete the definition of a channel subsystem using the *Delete* action from the context menu (or action code /SF) on the Channel Subsystem List. If you delete a channel subsystem, all channel paths, partitions, and connections to control units and devices for that CSS are also deleted; the control units and devices are not deleted.

**Working with partitions**

The following section describes how to work with partitions. Note that you can define partitions for a processor regardless of whether it is defined with configuration mode BASIC or LPAR.
Defining partitions

Define partitions as follows:

1. On the Primary Task Selection panel, select Define, modify, or view configuration data and on the resulting panel the object Processors. HCD displays the Processor List of processors currently defined in the IODF.

2. On the Processor List:
   - for SMP processors, select the processor and the Work with partitions action from the context menu (or action code /SF580000p/SF590000).
   - for XMP processors, select the processor and the Work with channel subsystems (XMP) action from the context menu (or action code /SF580000s/SF590000) to display the Channel Subsystem List. From this list, select the appropriate channel subsystem and the Work with partitions action from the context menu (or action code /SF580000p/SF590000).

   HCD displays the Partition List showing the currently defined partitions for the designated processor.

   ┌───────────────────────────── Partition List ─────────────────────────────┐
   │ Goto Backup Query Help │
   │ ----------------------------------------------------------------------- │
   │ Row1 of 2                │
   │ Select one or more partitions, then press Enter. To add, use F11. │
   │ Processor ID ....: PROC1 This is the main processor │
   │ Configuration mode .: LPAR │
   │ Channel Subsystem ID : │
   │ / Partition Name Number Usage + Description │
   │ _ PROD1 1 OS First production partition │
   │ _ PROD2 2 OS Second production partition │
   │ **************************** Bottom of data **************************** │
   └──────────────────────────────────────────────────────────────────────────┘

3. Use F11=Add to add the partitions. The data-entry fields are shown in the following figure, with sample data:

   ┌─────────────────────── Add Partition ───────────────────────┐
   │ Specify the following values. │
   │ Partition name . . TEST3 │
   │ Partition number . . 5 │
   │ Partition usage . . CF + │
   │ Description . . CF partition │
   └─────────────────────────────────────────────────────────────┘
The partition usage field marks a partition to be used for coupling facility support or for operating system usage. The type of partition usage can be either: CF, OS, or CF/OS.

Specify CF/OS if the partition usage will be determined at partition activation. You can then include this partition into the access list of all channel path types. At partition activation those definitions are ignored that are not valid for the actual usage.

4. Press the Enter key. HCD displays the updated Partition List, if you have not yet defined any channel paths.

If you have already defined channel paths, HCD displays the Update CHPID Access and Candidate Lists panel, where you can include the partition in the access or candidate list of a channel path. For an explanation of access and candidate list, refer to “Defining channel paths” on page 113.

After pressing the Enter key HCD displays the Update Device Candidate Lists panel, if the new partition is given access to a channel path that attaches devices with an explicit device candidate list. Use this panel to add the new partition to the device candidate list of the listed devices.

**Defining reserved partitions**

For XMP processors, HCD provides the capability to add or remove logical partitions via dynamic I/O configuration. In an IODF used to create your initial IOCDS for power-on reset (POR), you can define reserved partitions, which you plan to add dynamically at a later point in time. In the **Add Partition** dialog (see step 3 on page 106 from the previous list), you specify an ‘*’ as the placeholder partition name for reserved partitions. Reserved partitions will appear with this ‘*’ at the end of the Partition List. Furthermore, you specify a partition number, a usage type and optionally a description.

Reserved partitions do not appear in the access or candidate lists of channel paths or devices.

To activate a partition dynamically, you need to change the ‘*’ name to a valid partition name and to define the appropriate partition configuration before building a new production IODF.

**Note:** You cannot change the partition number dynamically.

**Changing partitions**

You can change the following data of a partition using the **Change** action from the context menu (or action code **c**) on the Partition List. On the resulting Change Partition panel, you can change:

- Name
- Number
- Usage
- Description
If there are already channel paths attached to the partition, HCD displays the channel path access and candidate lists after pressing the Enter key on the Change Partition panel. On these lists, you can update the channel path access of the partition.

**Note:** You can also change these partition definitions (except the name) by simply typing over the appropriate columns on the Partition List.

### Changing partition names dynamically

To dynamically change a partition name, you have to perform two steps:

1. Disconnect all channel paths and devices from the partition and change the partition name to * (see “Defining reserved partitions” on page 107). Activate this intermediate IODF. (This is the only required step if you want to keep this partition as a reserved partition).
2. Now you can change the * partition name to a new valid name, reconnect the wanted channel paths and devices and activate the new configuration.

### Repeating (copying) partitions

You can copy partitions within the same or to another IODF. When copying a partition, the following related objects and connections are also copied:

- Channel paths having the partition in their access list
- Control units reached by the partition
- Devices reached by the partition

**Note:** PCHID values are not copied.

Before copying the partition, perform the same checks as when repeating a processor (see “Repeating (copying) processors” on page 96). Omit step 1 that applies to a processor repetition only.

Then copy a partition as follows:

1. Copy the partition using the **Repeat (copy) partitions** action from the context menu (or action code X) on the Partition List. The Identify Target IODF panel is displayed.
2. Specify the IODF to which the selected partition is to be copied. The default IODF is the IODF you are currently working with.
3. On the following Repeat Partition panel, specify the required values and press the Enter key.

Definitions for the source partition are merged with the definitions of the target partition. If the target partition has another partition number than the source partition, HCD keeps the target partition number.
If control units or devices already exist in the target IODF (same number and type), HCD tries to map them. “Migrating additional IOCP input data sets” on page 265 explains in detail when a device or control unit is mapped. If they are mapped, the attributes of the target control unit or device are kept.

See “Explicit device candidate lists” on page 110 how to handle device candidate lists.

Note: HCD provides special processing when copying/repeating partitions with CTC connections. For more information refer to “Copying/repeating channel subsystems with CTC connections” on page 103.

**Transferring partition configurations**

Use this function to transfer control units and devices attached to a channel path from one partition to another within the same IODF.

In contrast to the *Repeat (copy)* function, you do not copy the partition and channel paths, but move the attached control units and devices to another partition, possibly in another processor.

Before transferring the data, you must define the target channel path with its partition access and candidate list, dynamic switch ID, entry switch ID and entry port.

The new channel path may have a different type than the source channel path.

1. On the Partition List, select the *Transfer (move) partition configs* action from the context menu (or action code X).

2. On the *Identify Target Partition* panel, specify the target processor and partition.

3. The *Transfer Partition Configuration* panel is shown. To transfer all control units and devices reached by the source partition, specify a new CHPID value for every source CHPID. The new CHPID of the target partition must exist. The data-entry fields with sample data are shown in the following figure.
Specify the new CHPID values.

<table>
<thead>
<tr>
<th>CHPID Type</th>
<th>Mode</th>
<th>New CHPID</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>BL</td>
<td>DED</td>
</tr>
<tr>
<td>02</td>
<td>BL</td>
<td>DED</td>
</tr>
<tr>
<td>03</td>
<td>BL</td>
<td>DED</td>
</tr>
<tr>
<td>04</td>
<td>BL</td>
<td>DED</td>
</tr>
<tr>
<td>20</td>
<td>CNC</td>
<td>SHR</td>
</tr>
<tr>
<td>21</td>
<td>CNC</td>
<td>SHR</td>
</tr>
<tr>
<td>25</td>
<td>CNC</td>
<td>DED</td>
</tr>
<tr>
<td>26</td>
<td>CNC</td>
<td>DED</td>
</tr>
</tbody>
</table>

4. After you press the Enter key the Partition List is displayed again.
   The attribute values of the transferred control units and devices remain the
   same for unit address/range, destination link address, time-out, and STADET.
   The logical address, protocol, and I/O concurrency level of a control unit
   remain the same if they are compatible with the target processor and channel
   path. If they are not compatible, default values are used.

**Explicit device candidate lists**

If the devices that are affected by the Transfer (move) partition configs action from
the context menu (or action code X) have an explicit device candidate list, the
result of the transfer action depends on whether or not the device was already
connected to the target processor. See Figure 46 for the different combinations.
These combinations also apply to the Repeat Partition and Repeat Processor
actions.

<table>
<thead>
<tr>
<th>Transfer source partition</th>
<th>Transfer target partition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Device already connected to target partition</td>
</tr>
<tr>
<td></td>
<td>no cand</td>
</tr>
<tr>
<td>no cand</td>
<td>=</td>
</tr>
<tr>
<td>cand + (partition included)</td>
<td>=</td>
</tr>
<tr>
<td>cand - (partition not included)</td>
<td>cand - (*)</td>
</tr>
</tbody>
</table>

**Note:** Result of the Transfer Action, Relation of Device to Partition:
- = no action, target partition remains unchanged
- no cand no explicit device candidate list exists for partition
- cand + partition included in explicit device candidate list
- cand – partition not included in explicit device candidate list

*Figure 46. Result of the Transfer (Move) Partition Configs Action*

**Note:** (*) The source partition is not included in the explicit device candidate list.
During the transfer, HCD checks whether all partitions of the source
processor in the candidate list have the same name on the target processor.
If partitions with the same name on the target processor are identified they are added to the device candidate list for the target processor. If no partition with the same name is found for the target processor, no explicit device candidate list is built. Therefore, a partition transfer can result in a loss of candidate lists, if all partition names between source and target processor are different. It is recommended to run a device compare report after the partition has been transferred.

**CF channel paths**
Connections of CF sender and CF receiver channel paths will not be transferred. You have to connect them again after having transferred the partition.

### Deleting partitions

You can delete the definition of a partition using the *Delete* action from the context menu (or action code `[d]`) on the Partition List.

To enhance deleting of partitions, which have a CHPID assigned to only this partition in either the access or candidate list, the **Confirm Delete Partition** indicates all such CHPIDs that are exclusively assigned to this partition by flagging them with an ‘*’ (CHPIDs B8 and BA in Figure 47). Thus, users can remove the flagged CHPIDs in one step and then delete the partition more efficiently.

---

**Figure 47. Confirm Delete Partition**

---
Working with channel paths

Channel paths can be dedicated, reconfigurable, shared, or spanned. The following list explains when to use which channel path operation mode.

**DED**  Dedicated; if you want only one logical partition to access a channel path, specify that channel path as dedicated. You cannot reconfigure a dedicated channel path. This is the default mode.

**REC**  Reconfigurable; if you want only one logical partition at a time to access a channel path and you want to be able to reconfigure the channel path from one partition to another, specify that channel path as reconfigurable.

**SHR**  Shared; if you want more than one logical partition to access a channel path simultaneously, specify that channel path as shared.

**SPAN**  Spanned; if in XMP processors for certain channel types, you want to have a shared channel accessed by partitions from multiple logical channel subsystems, specify that channel path as spanned.

On the Add Channel Path panel, enter a channel path type and use F4=Prompt for the operation mode to find out the allowed operation modes for the specified type.

Channel paths can also be categorized as static or managed. For more information, see “Defining managed channel paths” on page 117.

Using Multiple Image Facility

If a processor complex has Multiple Image Facility (MIF) capability, and is running in LPAR mode, multiple logical partitions can access the same shared channel paths, thereby reducing the number of required physical connections. In contrast, if a processor complex does not have MIF capability, all logical partitions must use separate channel paths to share I/O devices. For more information about LPAR mode and MIF, see the PR/SM Planning Guide.

More about spanned channel paths in multiple LCSSs

Depending on the processor type, in the HCD dialog you may define certain channel paths with operation mode SPAN. A spanned CHPID will have partitions belonging to more than one channel subsystem in its access and candidate list.

A spanned channel path will be created with the same CHPID number in all channel subsystems that are using it. For example, you have a processor MCSSPRO1 with channel subsystems 0 through 3, and you create CHPID 1A (type IQD, SPAN) and let it access partitions from CSS 0, 2, and 3. Then CHPID 1A is the same CHPID in CSSs 0, 2, and 3. In CSS 1, you can use CHPID 1A for a different channel path.

Generally speaking, a channel subsystem that is not using a spanned channel can use the CHPID of that spanned channel for a separate channel path definition.

If you define a channel as SPAN, but connect it to partitions from a single channel subsystem only, then HCD displays its operation mode as SHR. The other way round, if a shared channel path is eligible for being spanned, and you enlarge its access or candidate list with partitions from multiple logical channel subsystems, then HCD displays this channel’s operation mode as SPAN.
Note:
It is dependent on the processor support level which channel path types can be defined as spanned. Managed channels cannot be defined as spanned.

Defining channel paths

At first, you define a channel path together with its access to logical partitions. Then you may define special channel path characteristics. These possibilities are described in "Defining special channel path characteristics" on page 116.

1. On the HCD entry panel, select the task Define, modify, or view configuration data and from the resulting panel, select Processors. HCD displays the Processor List of defined processors.

2. On the Processor List:
   - for SMP processors, select the processor and the Work with attached channel paths (SMP) action from the context menu (or action code $S$).
   - for XMP processors, select the processor and the Work with channel subsystems . (XMP) action from the context menu (or action code $S$) to display the Channel Subsystem List. From this list, select the appropriate channel subsystem and the Work with attached channel paths action from the context menu (or action code $S$).

HCD displays the Channel Path List showing all channel paths defined for the selected processor/channel subsystem.
If the Type contains three asterisks (**), the IODF channel path type is unknown to the currently used HCD.

3. Use F11=Add to add channel paths. The data-entry fields are shown in the following panel, with sample data:

![Add Channel Path Panel]

```
Specify or revise the following values.
Processor ID . . . . : PROC03  Processor type 2097
Configuration mode . : LPAR
Channel subsystem ID : 0  H05LP01 - H05LP15
Channel path ID . . . . 02 +  PCHID . . ___
Number of CHPIDs . . . . 1
Channel path type . . FCV +
Operation mode . . . . SHR +
Managed . . . . No (Yes or No)  I/O Cluster ______ +
Description . . . . ________________________________ +
```

Specify the following values only if connected to a switch:
- Dynamic switch ID . . . . 01 + (00 - FF)
- Entry switch ID . . . . 01 +
- Entry port . . . . . . . . . 84 +

Figure 48. Add Channel Path

For physical channels on an XMP processor, you have to specify the physical channel identifier (PCHID) belonging to the channel path identifier (CHPID). The CHPID Mapping Tool (CMT) can be used to make the mapping between CHPIDs and PCHIDs easier (see “How to interact with the CHPID Mapping Tool” on page 213).

4. For each static channel path you can specify which logical partitions can access that channel path. After you press the Enter key on the Add Channel Path panel, HCD displays the Define Access List.

![Define Access List Panel]

```
Select one or more partitions for inclusion in the access list.
Channel subsystem ID : 0  H05LP01 - H05LP15
Channel path ID . . . . 02  Channel path type . . FCV
Operation mode . . . . SHR  Number of CHPIDs . . . . 1
/ CSS ID Partition Name Number Usage Description
/ 0 H05LP01 1  OS
  0 H05LP02 2  OS
  0 H05LP03 3  OS
  0 H05LP04 4  OS
  0 H05LP05 5  OS
  0 H05LP06 6  OS
/ 0 H05LP07 7  OS  Mini-OS
  0 H05LP08 8  OS  Automation-OS
  0 H05LP09 9  OS  TEST - OS
  0 H05LP10 A  OS
```

If you are working on spanned channel paths of an XMP processor, the Define Access List also shows the partitions defined for other channel subsystems:
If you want a logical partition to access a dedicated, reconfigurable, or shared channel path when you initially activate the logical partition, place that logical partition in the channel path’s access list. For shared channel paths and spanned channel paths, you can place more than one partition in the access list.

5. If you do not include all partitions in the access list, you are prompted for the candidate list (for reconfigurable and shared channel paths) after pressing the Enter key.

From the IOCP point of view, the channel path candidate list includes the channel path access list. From the HCD point of view, the channel path candidate list does not include the channel path access list. The partitions already in the access list do not appear in the candidate list.

If you want to be able to configure a reconfigurable or shared channel path online to a logical partition, place that logical partition in the channel path’s candidate list.
6. After pressing the Enter key, you return to the Channel Path List. Scroll to the right to get an overview of the access and candidate list of a channel path. The following matrix is displayed:

![Diagram of Channel Path List](image)

The legend above the partition matrix shows how the partition names are associated with the columns of the partition matrix. The headings **Partitions 0x, Partitions 1x, Partitions 2x** and so on, if scrolling to the right, indicate that the partitions for the related channel subsystems (0, 1, 2, ...) are shown. The column numbers correspond to the partition numbers in the pertaining channel subsystem. In the example above, column 1 under Partitions 0x shows the definitions for the partition with partition number 1 in CSS 0.

The following entries may appear in the partition matrix:

- **a** indicates that the partition is in the channel path’s access list.
- **c** indicates that the partition is in the candidate list.
- ***** is shown for a managed channel path in all logical partitions that potentially can access that channel path.
- **#** indicates that the channel path (which is defined to the channel subsystem named in the Channel Subsystem ID field) cannot be attached to the partitions of another channel subsystem. Either a channel path with the same identifier is already defined for the other channel subsystem, or the channel path cannot be spanned or it can be spanned, but the channel path mode is not SPAN or SHR. For information on how to change a CHPID’s operation mode to SPAN, if applicable, refer to “Changing the operation mode of a channel path” on page 124.

**Defining special channel path characteristics**

This section handles the following topics:

- “Defining managed channel paths” on page 117
- “Defining multiple channel paths in one step” on page 117
- “Connecting a channel path to a switch” on page 117
- “Defining the maximum frame size” on page 118
Defining managed channel paths: You can define a channel path as being managed by Dynamic Channel Path Management (DCM). DCM will use such a channel path to dynamically assign the logical paths to control units in order to optimize I/O activity. A managed channel path must connect to a dynamic switch and may be used for control units that connect to the same switch. If a channel path is defined as managed in an LPAR mode processor, it must be defined as shared. It cannot be connected to logical partitions but must specify an I/O cluster name. An I/O cluster is a sysplex that owns the managed channel path. All systems of the sysplex on the given processor are allowed to share the managed channel path. A managed channel path cannot be connected to a control unit by HCD.

Defining multiple channel paths in one step: You can define, in one step, a group of channel paths of the same type and mode and with consecutive identifiers. It is recommended to define only a group of channel paths that have the same partitions in their access and candidate lists. Otherwise, you have to change the channel paths that have different partitions in their access and candidate list in a further step.

1. Define the group by specifying the first channel path identifier (CHPID) and the number of channel paths in the group. Define a channel path type, mode, and description. HCD applies the definition to all channel paths in the group.
2. Type over the fields that are different, for example description, in the Channel Path List.

Connecting a channel path to a switch: If you have already defined a switch, you can connect the channel path to the switch on the Add Channel Path panel. Specify the dynamic switch ID, the entry switch ID, and the entry port to connect the channel path to a switch.

The values are only valid for the first channel path if you have defined a group of channel paths in one step. To define values for the other channel paths of the group, HCD displays an additional panel. This panel allows you to define the entry ports for all subsequent channel paths of the group. For information on dynamic switch ID and entry switch ID, refer to "Possibilities of switch connections" on page 169.
Defining the maximum frame size: For an IQD channel path type, HCD allows you to specify the maximum frame size to be used for iQDIO requests on that channel path. If you define or update an IQD channel path, HCD will display a dialog panel that allows you to specify a maximum frame size for the channel path.

**Specify Maximum Frame Size**

Specify or revise the value below.

Maximum frame size
in KB ......... 16 +

F1=Help  F2=Split  F3=Exit  F4=Prompt
F5=Reset  F9=Swap  F12=Cancel

Defining more than 160 TCP/IP stacks: When defining or changing channel paths of type OSD for XMP processors with the corresponding support level, HCD prompts you with a dialog whether you want to allow for more than 160 TCP/IP stacks with this channel. This is done by disabling priority queuing. If priority queuing is disabled, the channel can support four times as many queues (4 * 480 = 1920 subchannels) corresponding to four times as many TCP/IP stacks (4 * 160 = 640) as with enabled queue prioritization.

**Allow for more than 160 TCP/IP stacks**

Specify Yes to allow more than 160 TCP/IP stacks, otherwise specify No. Specifying Yes will cause priority queuing to be disabled.

Will greater than 160 TCP/IP stacks be required for this channel? ... No

F1=Help  F2=Split  F3=Exit  F5=Reset  F9=Swap
F12=Cancel

Defining or editing a CIB channel: When defining or changing a CIB channel path, HCD prompts you with a dialog which asks for the specification of the Adapter ID of the HCA and the Port on the HCA of that channel path.

**Specify HCA Attributes**

Specify or revise the values below.

Adapter ID of the HCA .. _ +
Port on the HCA .......... _ +

F1=Help  F2=Split  F3=Exit  F4=Prompt  F5=Reset
F9=Swap  F12=Cancel
Defining spanned channel paths: You can define a suitable channel path as *spanned* directly when creating it. On the Add Channel Path panel from Figure 48 on page 114, specify Operation mode . . . . SPAN

After pressing the Enter key, HCD displays the combination of the Define Access List and the Define Candidate List, offering partitions from multiple logical channel subsystems. Note that if you, nevertheless, in both lists select only partitions from the current CSS, then the operation mode of the channel path is set back to SHR.

For information on how to change a CHPID’s operation mode to SPAN, if applicable, refer to “Changing the operation mode of a channel path” on page 124.

Over-defining channel paths on an XMP processor: For an XMP processor, you can define a channel path that is not physically installed on the machine. This may be useful if you want to migrate from a machine which had more channels defined than the target XMP processor has currently installed, or if you want to prepare a configuration for future upgrades of the channel cards.

To distinguish an over-defined CHPID from a physically installed CHPID, use character ‘*’ for the PCHID value when creating the CHPID. An over-defined CHPID must adhere to all validation rules, but are not taken into account by an IOCDS download. Also they are not included in the IOCP statements or in a CONFIGxx member or during a dynamic activation.

If a control unit contains only CHPIDs with a PCHID value ‘*’, the whole control unit (including any attached devices) is omitted from the configuration to be activated.

When installing the channel path later, you must edit the CHPID and replace the ‘*’ by its valid PCHID.

Establishing coupling facility channel path connections

Before you start to establish a Coupling Facility (CF) channel path connection, you must have defined a processor that supports coupling facilities, a coupling facility partition, a coupling facility receiver (CF receiver) channel path and coupling facility sender (CF sender) channel path, or peer channel paths.
1. On the Channel Path List (Figure 49 on page 116) select a channel path and the Connect CF channel path action from the context menu (or type action code f). HCD displays the CF Channel Path Connectivity List showing all CF channel paths defined for a processor.

To show that a CHPID is already connected in another IODF you can set the indicator in the Occ (occupied) column to Y (yes). You cannot connect a CHPID labeled Y. However, you can change the occupied status by overwriting.

Notes:

a. The "Source partition name" field indicates a name only when the Filter function is employed.

b. Column CU type indicates the type of the connected control unit(s).

2. Select the source channel path for a coupling facility connection and the Connect to CF channel path action from the context menu (or action code p). HCD displays the Connect to CF Channel Path panel. The data-entry fields are shown in the following figure, with sample data:

```
Specify the following values.
Source processor ID . . . . . : SMPPROC
Source channel subsystem ID . . :
Source channel path ID . . . . . : 19
Source channel path type . . . . : CFS
Destination processor ID . . . . . : XMPPROC1 +
Destination channel subsystem ID . . +
Destination channel path ID . . . . . +
Timing-only link . . . . . . . : No
```

Figure 51. Connect to CF Channel Path
3. To establish the CF channel path connection, specify the destination processor ID, destination channel subsystem ID and destination channel path ID.

If the CF control unit definition does not yet exist, HCD automatically generates a CF control unit and CF devices for a sending channel path when CF channel paths are connected. HCD uses type CFS for a (legacy) CF sender channel path and type CFP for a CF peer channel path. The sending CF channel paths that connect one processor to a CF partition are assigned to the same CF control unit. For each CF sender channel path connection, HCD generates two CF devices. For each sending CF peer channel path, HCD generates seven CF devices. HCD proposes the highest unused control unit number and highest unused consecutive device numbers in the IODF. If a CF peer channel path does not connect to a target CF partition (i.e., the sending function is not used), HCD does not connect the channel path to CF control unit and CF devices.

If a CF control unit definition is already used for another CF connection from the processor of the sending CF channel path to the same target CF partition, HCD proposes the same control unit number. This control unit number may be overwritten by an unused control unit number, provided the partition lists of the channel paths on the existing control unit do not overlap with the partition list of the sending CF channel path for the new CF connection. Thus, it is possible to establish more than 8 CF links between a single CPC and a specific target CF partition.

Note: HCD checks the partition access lists of the channel paths for an overlap. That means, you can define an overlap in the partition candidate lists. In such cases however, you must ensure by operational means that at any one time, the CHPIDs of only one of the control units with overlapping partitions are configured online.

The following panel is displayed, where you have to confirm or revise the values for the CF control unit and CF devices. The data-entry fields are shown in the following figure, with sample data:

```
Add CF Control Unit and Devices

Confirm or revise the CF control unit number and device numbers for the CF control unit and devices to be defined.

Processor ID ........: XMPPROC1
Channel subsystem ID ....:
Channel path ID ........: 19       Operation mode ..: SHR
Channel path type .......: CFS
Control unit number ..: FFFE +
Device number ........: FFFE
Number of devices ..: 2
```

Figure 52. Add CF Control Unit and Devices

Notes:

a. The CF control unit and device definitions are displayed on the Control Unit List and on the I/O Device List, but in a disabled state where they cannot be modified or deleted.

b. If you specified a timing-only link in the dialog from Figure 51 on page 120 then the field **Number of devices** is set to 0 and cannot be changed, as no
Defining Server Time Protocol (STP) links

HCD supports Server Time Protocol (STP) links (timing-only links) between two zSeries (z890, z990 or later) processors. Timing-only links are only needed in case coupling links are not desired, not possible or not needed. If you want to define a coupling facility connection which will be used as a timing-only link, you must set the Timing-only link entry in the Connect to CF Channel Path panel (Figure 51 on page 120) to Yes. Both source and destination processors must be timing-capable in this case and the used channel paths must either be CFP or CBP.

Establishing a timing-only link between two processors does not require a CF partition, but can be established between two OS partitions.

For an STP only link, HCD generates a control unit of type ‘STP’ on both sides of the connection. No devices are defined. ‘STP’ is used as control unit type in the CF Channel Path Connectivity List in column CU type, which indicates the type of the connecting control unit(s) for non-STP links (see Figure 50 on page 120).

For changing a CF connection to an STP only connection and vice versa, you must break the existing connection and establish a new one.

Disconnecting coupling facility channel path connections

Perform the following steps to break a coupling facility channel path connection:

1. On the Channel Path List select any channel path and the Connect CF channel path action from the context menu (or action code F1). HCD displays the CF Channel Path Connectivity List showing all CF channel paths defined for a processor.

2. Select the source channel path for a coupling facility connection and the Disconnect action from the context menu (or action code F3).

Note: The appropriate CF control unit definition is removed implicitly with the last broken connection to the coupling facility to which the control unit belongs. The appropriate CF device definitions are removed implicitly, when the coupling facility connection to which they belong is broken.
Changing channel paths

To change channel path data you have to follow the same panel flow as for defining channel path data:
- Changing channel path characteristics
- Changing channel path access and candidate list

The following steps describe the panel flow and where you can change the data.

1. On the Channel Path List, select a channel path and the Change action from the context menu (or action code /SF580000c/SF590000).

2. On the following Change Channel Path Definition panel you can change channel path definitions such as:
   - Channel path ID (see also "Changing processors" on page 93 for an example of the Update Channel Path Identifiers panel, and how to change the CHPID values)
   - Channel path type (see "Changing the type of a channel path" on page 124)
   - Operation mode (see "Changing the operation mode of a channel path" on page 124)
   - PCHID
   - Description
   - If connected to a switch
     - Dynamic switch ID
     - Entry switch ID
     - Entry port

3. After pressing the Enter key, the Define Access List is displayed. Select one or more partitions to be included in the access list.

4. After pressing the Enter key again, the Define Candidate List is displayed (if applicable). Select one or more partitions to be included in the candidate list.

You can also change channel path definitions (except the channel path ID) and the channel path’s access and candidate list by simply typing over the appropriate values on the Channel Path List. To change the access and candidate list definitions, scroll to the right to see the channel path/partition matrix (refer to Figure 49 on page 116). Overwrite the values in the channel path/partition matrix with either a for access list and c for candidate list.

Changing the ID of a channel path
Changing the ID of a channel path may first require the disconnection of the entry switch and entry port on the channel path. If channel paths of multiple processors or channel subsystems (e.g. spanned CHPIDs) connect to the same entry switch and entry port, proceed as follows:

1. Remove the entry switch and entry port from the channel path definitions.
2. Change the channel path ID of the corresponding channel paths.
3. Once again add the entry switch and entry port to the channel paths.

**Changing the type of a channel path**

Changing the type of a channel path from parallel to serial (or vice versa) will result in changing the type of all other channel paths that are attached to the affected logical control units. When changing the type of a channel path:

- The new/changed channel path type must not conflict with the already existing channel path IDs, control unit and device parameters. Adjust the values of the affected control units and devices according to the rules of parallel or serial channel path type. (For information on how to change control unit processor attachment and device parameters, see: “Changing control units” on page 134.)
- When changing from serial to parallel, you have to disconnect the entry switch and entry port first, if the channel path is connected to a switch.
- When changing a channel path of type BL or BY that is connected to more than one control unit, the channel path has to be defined to a corresponding CVC (converter channel path) first and then be changed to serial.

**Note:** A channel path type change cannot be performed in one step while changing the channel path ID.

**Changing the type of a coupling facility channel path:** To change the type of CF channel paths, disconnect the channel path you want to change before performing the type change. Any coupling facility devices associated with the changed channel path are removed by HCD. The associated control unit is removed only when the last connection to the coupling facility to which the control unit belongs is broken. (For details on that task, see: “Establishing coupling facility channel path connections” on page 119.)

**Changing the operation mode of a channel path**

Changing the operation mode of a channel path is dependent on its type. For example, BL, BY, CVC, CBY, and CF receiver channel paths cannot be shared.

Before you can change the operation mode of a channel path, the rules for partition access and candidate lists of those channel paths that are attached to the affected logical control units must conform to the rules for the new operation mode. You have to check which partitions have access to these channel paths. When changing the channel path operation mode from SHR to REC or DED, you first have to remove partitions in the appropriate access and candidate lists. The partition lists for the affected logical control units have to be changed when the mode change has been done.

**Changing the operation mode of a channel path to SPAN:** If you want to change the operation mode to SPAN for applicable channel path types, you must ensure that the CHPID is unused in those channel subsystems into which it should be spanned. This means that the CHPID whose operation mode you want to change, must be unique throughout the processor complex. So you need to distinguish the following scenarios:

- For an existing shared CHPID that is uniquely defined throughout all LCSSs of the processor:
  Enlarge its access and candidate lists with partitions from other channel subsystems. For a shared CHPID, in the Channel Path List, scroll right once for each CSS of the current processor to see the available partitions that you can specify for access or candidates.
Select one or more channel paths, then press Enter. To add, use F11.

1=LPAR01 2=LPAR02

- For an existing dedicated or reconfigurable CHPID that is uniquely defined throughout all LCSSs of the processor:
  You can change its operation mode to SPAN using the Change action from the context menu (or action code /SF580000c/SF590000) on the Channel Path List. In the subsequent Define Access List and Define Candidate List panels, you must select at least two partitions from different channel subsystems (0x, 1x, ...).

- For any existing CHPID that is multiply defined throughout the LCSSs in the processor complex:
  You must delete the CHPID from all but one LCSS, before you can change its operation mode to SPAN, using the Change action from the context menu (or action code /SF580000c/SF590000) on the Channel Path List and selecting appropriate partitions from the Define Access List and Define Candidate List panels.

When spanning a channel path, that has control unit(s) (and devices) attached, to a new CSS, HCD invokes a dialog asking whether these control unit(s) (and devices) should also be reachable from the new CSS.
Specify Yes, if you want all existing CU and device connections of the designated channel path to be copied to all channel subsystems the channel path gets newly spanned to.

**Changing the operation mode of CF channel paths:** CF channel paths that connect a processor to the same coupling facility partition via the same control unit must be either all shared or all nonshared. Hence, if you want to change the operation mode for one channel path (from non-SHR to SHR), you have to change it for all. To do this proceed as follows:

1. Disconnect all receiving CF channel paths that connect a processor to the same coupling facility partition (see “Disconnecting coupling facility channel path connections” on page 122).
2. Change the operation mode of all sending CF channel paths.
3. Re-establish the connections for all channel paths that you disconnected in step 1.

The associated coupling facility control unit and coupling facility devices are removed and generated again by HCD.

**Changing a coupling facility connection**

To change a coupling facility connection:

1. Disconnect a connection (see “Disconnecting coupling facility channel path connections” on page 122).
2. Establish the new connection (see “Establishing coupling facility channel path connections” on page 119).

**Note:** A Y (yes) in column Occ (occupied) of the channel path list indicates that this CHPID is occupied. This is useful if you have another IODF where the CF connection is already defined. If you wish to connect this path, you must first change the occupied status by overwriting the Y with N.

It is not possible to include the first CF or CF/OS LP in, or remove the last CF or CF/OS LP from, the combined access and candidate list of a connected CF peer channel. Instead, the CF connection first has to be removed. Then, the CF LP can be connected to or disconnected from the CF peer channel path; afterwards, the CF peer connection can be reestablished. This is to avoid implicit generations of deletions of CF control units and devices.

**Aggregating channel paths**

When selecting *Aggregate channel paths* from the Channel Path List against a channel path, HCD offers the possibility of moving all control units from a source
channel path to the selected target channel path of the same processor. This is useful for combining several under-utilized channel paths to a single one.

You can aggregate channel paths using the **Aggregate channel paths** action. The following steps describe the procedure:

1. On the Channel Path List select a channel path and then the **Aggregate channel paths** action from the context menu or type action code \( g \) next to the selected CHPID.

2. On the following Aggregate CHPID definition panel you can enter the target channel path ID for the aggregate action.

   ![Aggregate CHPID panel]

HCD displays a list of control units that are currently attached to the source CHPID. If possible, each control unit shows the switch port to which it is connected. Also, the target switch port and the target link address after the aggregate is shown if HCD can determine these. You can select all or a subset of control units to be aggregated to the target CHPID. The selected control units and their attached I/O devices are disconnected from the source CHPID and connected to the target CHPID. The target CHPID may now be connected to a different switch than the source CHPID. Panel **Select Control Units to be Aggregated** allows you to change the control unit port and link address for the move to the target CHPID.

![Select Control Units to be Aggregated panel]

After successful aggregation, a message will be displayed and the Aggregate CHPID panel will remain to allow you to aggregate additional channel paths. If aggregation fails because of validation errors, the validation errors are displayed. If prompting for channel paths for aggregation, HCD will only show the channel paths that allow aggregation without validation errors. Prompting is thus a useful planning aid.

The **Aggregate Channel Paths** action is only possible if the following prerequisites are fulfilled:
• Source and target channel paths must be different.
• All selected control units connected to the source channel path must be connectable to the target channel path.
• The source channel path must not be connected to a control unit which is already connected to the target channel path. In addition, a link address - unit address - CUADD combination used by a control unit connected to the source channel path must not also be used by a control unit connected to the target channel path.
• Either the source channel path must have the same channel path mode as the target channel path, or all devices accessible by the source channel path must be connected to only one channel path.
• Source and target channel paths must have defined a dynamic switch.
• The user must not lose connectivity by a channel path aggregate action. The source channel path access and candidate list must be the same as or a subset of the target channel path access and candidate list.
• By connecting control units of the source channel path to the target channel path, the defined maximum value for the target channel path type (e.g. maximum number of unit address ranges) must not be exceeded.

As a result of an aggregation action, HCD will:
• Change the preferred channel path of a device to the target channel path if the source channel path was the preferred channel path of the device initially.
• Leave the reachability of devices by logical partitions unchanged.
• Move the CTC control units of the source channel path port to the entry port to which the target channel is connected.

Deleting channel paths

You can delete the definition of a channel path using the Delete action from the context menu (or action code d) on the Channel Path List. If you delete a spanned channel path, it is removed from all channel subsystems which had access to it.

Working with control units

The following section describes how to work with control units.
Defining control units

You need two steps to define a control unit:
• Define the control unit characteristics
• Define how the control unit is attached to processors.

Before you define a control unit, you should have defined the processors and channel paths to which the control unit is to be attached.

Defining the control unit characteristics
1. On the Primary Task Selection panel, select Define, modify, or view configuration data and on the resulting panel the object Control units. HCD displays the Control Unit List showing all control units currently defined in the IODF.

Column CUADD shows the CUADD value defined for the control unit, where available. If the CUADD is inconsistently set for the control unit among processors, an asterisk “*” is displayed.

Column #CSS shows the number of channel subsystems to which a control unit is connected. This column contains a value only if a connection exists.

Column #MC shows the greater of the number of managed channel paths defined for the connected processors or the number of managed channel paths defined for the selected processor when coming down from the processor. This column contains a value only if managed channel paths are defined for the control unit.
Note: The CF control units generated when connecting CF channel paths are listed but are disabled for any action. It is not possible to add a new such control unit via this dialog.

2. Use F11=Add to define a new control unit. The data-entry fields are shown below, with sample data:

<table>
<thead>
<tr>
<th>Add Control Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify or revise the following values.</td>
</tr>
<tr>
<td>Control unit number .... 00D1 +</td>
</tr>
<tr>
<td>Control unit type .... 3990-3_______ +</td>
</tr>
<tr>
<td>Serial number ....... __________</td>
</tr>
<tr>
<td>Description ........ DASD control unit ________________</td>
</tr>
<tr>
<td>Connected to switches ..._ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ +</td>
</tr>
<tr>
<td>Ports ..........._ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ +</td>
</tr>
<tr>
<td>If connected to a switch:</td>
</tr>
<tr>
<td>Define more than eight ports . 1. Yes</td>
</tr>
<tr>
<td>2. No</td>
</tr>
<tr>
<td>Propose CHPID/link addresses and unit addresses . 1. Yes</td>
</tr>
<tr>
<td>2. No</td>
</tr>
</tbody>
</table>

**Defining switch connections**

The *Add Control Unit* panel can also be used to specify the switches and ports the control unit is attached to.

If you specify Yes for Define more than eight ports, the *Define Control Unit Ports* dialog will be displayed to allow you to specify up to 64 control unit/switch port connections. To connect a unit to a maximum of 128 switch ports, in this dialog, you can invoke another panel to define an additional 64 switch port connections.

If you specify Yes for Propose CHPID/link addresses and unit addresses and the control unit is connected to at least one switch, HCD suggests control unit to processor attachment parameters (channel path/link addresses and the unit address range) based on the switch/ports the control unit is connected to. HCD will propose up to eight channel path/link address pairs, starting with the channel path that has the lowest number of devices attached to it.

If you add a new control unit (via Add or Add-like), HCD automatically assigns as many logical paths as possible for all processors defined.

The following prerequisites must be fulfilled for this function:
- The control unit must support ESCON or FICON attachments and not be used for channel-to-channel (CTC) connections.
- The control unit must have physical switch / port connections (switch entry ports) defined.
- Channel paths that use the connected switch as a dynamic switch must exist.

HCD then automatically selects the channel paths and link addresses according to the following rules for each processor that has been defined.
All channel paths that use a switch that connects to the control unit as a dynamic switch are candidates for assignment.

The channel paths are sorted ascending by the value of the reversed channel path ID. The resulting sequence is again sorted ascending by the number of connected devices.

The connected control unit ports are ordered ascending by the numbers of already connected control units and path connections, respectively.

For each connected switch port in the resulting sequence, the channel paths are tested in sequence. If the switch port can be used as a link address, the CHPID/link address is taken.

A maximum number (up to 8) of possible CHPID/link address combinations is assigned.

On the following Select Processor/Control Unit panel you can type over the fields that are different from the suggested attachment values.

**Defining processor attachment data**

1. After pressing the Enter key on the Add Control Unit panel HCD displays a list that shows all the defined processors. You can then define how the control unit is to be attached to one or more processors.

   A Y for Yes in the Att column indicates that the control unit is attached to the processor.

2. Select a processor and the Select (connect/change) action from the context menu (or action code 5). When a control unit is attached to multiple processors, you can use the Group connect action from the context menu (or action code 6). This group action is particularly useful when performing symmetric changes, for example, on CPCs...
defined in an S/390 microprocessor cluster. The changes are applied to all
selected processors, when you issued the change action against a group of
processors.

When you issue a change or group connect action, the following panel for
processor-dependent control unit information is displayed:

```
Add Control Unit

Specify or revise the following values.

Control unit number . : 0099  Type . . . . . . : 2105
Processor ID . . . . : FR3BLPAR  Raised floor production
Channel Subsystem ID . :

Channel path IDs . . . . 07  08  *  *  *  _  _  _  _  +
Link address . . . . . . 80  81  _  _  _  _  _  _  _  _  _  _  +
Unit address . . . . . . _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
Number of units . . . . _  _  _  _  _  _  _  _  _  _  _  _  _
Logical address . . . _  _  _  _  _  (same as CUADD)
Protocol . . . . . . . _  _  _  _  _  (D,S or S4)
I/O concurrency level : 2  _  _  (1, 2 or 3)
```

3. On the Add Control Unit panel specify the channel paths that connect the
control unit to the processor.

If the control unit is attached to a switch, you have to define a link address for
each channel path. The link address is the port to which the control unit
attaches. If the control unit attaches only to one port, the link address is the
same for each channel. For addressing the target control unit in a fabric
containing cascade switching, a two-byte link address is used, which specifies
as first byte the switch address and as second byte the port address to which
the control unit is attached.

For a description what the link address is, see Figure 68 on page 170, Figure 69
on page 170 and Figure 71 on page 171

**Note:** For managed control units, i.e., control units that can have managed
channel paths assigned by DCM, you must indicate how many managed
channel paths can be connected to the control unit. Enter at least one
static channel path and the corresponding link address, and, in addition,
an asterisk (instead of the channel path ID and link address) for each
managed channel path.

You must also specify the unit address and the number of units, that is the unit
address range of I/O devices that the control unit recognizes. Serial control
units may have specified only one unit address range starting with 00.

If the path to the control unit is not unique, and more than one serial control
unit connects to the same channel path via the same link address, you have to
specify a logical address (CUADD parameter). For more information refer to
the explanation of the CUADD in the IOCP User’s Guide for your processor.

4. Press the Enter key. HCD displays the updated Select Processor/Control Unit
panel. There you may scroll to the right (using F20=Right) to see the data that
you have entered on the previous panel.

5. Repeat defining processor attachment data for all processors the control unit
should be attached to.

6. Press the Enter key to return to the Control Unit List.
Upgrading to two-byte link addresses
In a FICON fabric, all one-byte link addresses on a channel path may need to be migrated to a two-byte link address. HCD supports this definition change via the Change Channel Path Link Addresses dialog. If you specify a two-byte link address on a control unit for a specific channel path, and there are already one-byte link addresses specified on that path, the panel shown in Figure 55 appears.

This panel shows all link addresses specified for the specific channel path that must be changed. If there has been an entry switch defined for the channel, its ID and switch address (if defined) are displayed; else the displayed information is taken from the dynamic switch.

You can change the switch address. HCD then uses the modified value to preset the two-byte link addresses in the displayed New column. Its value is stored, if the entry switch is defined. Alternatively, the new two-byte link addresses can be entered.

Pressing Enter will change the control unit link addresses on the corresponding control units.

---

### Figure 55. Change Channel Path Link Addresses

You may want to change link addresses from two byte to one byte. This is possible, as long as all affected control units are attached to the channel path switch. If a defined two-byte link address is changed to a one-byte link address on a given channel path, all other two-byte link addresses defined for control units attached to that channel have to be changed to a one-byte link address also.
Changing control units

To change control unit data you have to follow the same panel flow as for defining control units.

- Changing Control Unit Characteristics
- Changing Processor Attachment Data

The following steps describe the panel flow and where you can change the data.

1. On the Control Unit List select a control unit and the *Change* action from the context menu (or action code /SF580000c/SF590000).

2. On the following Change Control Unit Definition panel you can change the following data:
   - Control unit number
   - Type-model
   - Serial number
   - Description
   - Connections to switches/ports

   **Note:** You can also change these control unit definitions (except the control unit number and the connections to switches/ports) by simply typing over the appropriate columns on the Control Unit List.

3. After pressing the Enter key you see the Select Processor/Control Unit panel. Select a processor and the *Select (connect/change)* action from the context menu (or action code /SF).

4. On the following Change Control Unit Definition panel you can change the processor attachment data:
   - Channel paths / Link addresses
   - Unit addresses / Number of units
   - Logical address
   - Protocol
   - I/O concurrency level

When changing control unit data of control units that affect other control unit or device data (like unit address/ranges), a list is displayed that shows all affected control units and proposed new address ranges for those control units. A panel like the following one is displayed:
After you modified control unit data (like protocol, I/O concurrency level, or unit address range), and pressed the Enter key, the Modify Device Parameters panel is shown with the devices attached to the affected control units. The devices are grouped by ranges:

HCD proposes starting unit addresses for the listed device groups.

Use the F20=Right key to scroll to the right to see the attached control units. Accept or change the definitions for unit address (UA New), Time-Out, STADET, and preferred CHPID.

**Changing control unit attachment parameters for multiple processors**

You can change control unit (CU) attachment parameters or attach a control unit for a group of processors. If all parameters to be changed are identical, use the following group action.
1. On the Control Unit List, select a control unit that is attached to the group of processors and use the **Change** action from the context menu (or action code `c`).

2. On the Change Control Unit panel press the Enter key. HCD then displays a Select Processor/Control Unit panel with a list of processors already defined (see Figure 53 on page 131).

3. Select the processors for which you want to change the control unit-processor definitions and use the **Group change** action from the context menu (or action code `g`).

4. The Change Control Unit Definition panel is displayed showing the values/attributes for the first processor in the group. An asterisk (*) in the Processor ID field indicates that you are using the **Group connect** action from the context menu and the changes will be applied to more than one processor.

---

### Disconnecting control units from a processor

1. On the Control Unit List, select a control unit and the **Change** action from the context menu (or action code `c`). HCD displays the Change Control Unit Definition panel.

2. On the Change Control Unit Definition panel, press the Enter key. HCD displays the Select Processor/Control Unit panel.

3. On the Select Processor/Control Unit panel select a processor and the **Disconnect** action from the context menu (or action code `n`).

---

### Disconnecting multiple control units from a processor:

If you want to disconnect multiple control units from one processor in one step, open the Control Unit list via the Channel Path List. On the Control Unit List, select one or multiple control units and use the **Disconnect** action from the context menu (or action code `n`).

---

### Priming control unit data

You can prime your I/O configuration in a work IODF with the control unit serial number for the active processor. For the prerequisites for this function refer to "Prerequisites" on page 9.
To prime the control unit serial number, select the action **Prime serial number** from the context menu (or action code `J`) on the Control Unit List. The Confirm Priming Control Unit Data List shows the selected control units with the sensed data for the control unit types and serial numbers, and their corresponding definitions in the IODF.

```
--- Confirm Priming Control Unit Data List ---
Command ==> ___________________________ Scroll ==> CSR
Press Enter to confirm priming, or Cancel to leave the list. A blank value will not change the IODF definition.

CU  Type  Serial Number ---
Number actual defined sensed defined
90C1 3990-L03 3990 33333
9100 3990-L03 3990 30984
9101 3990-L03 3990 30984
9140 3990-L03 3990 37160
9141 3990-L03 3990 37160
9180 3990-L03 3990 37201
9181 3990-L03 3990 37201
91C0 3990-L03 3990 37163
91C1 3990-L03 3990 37163

********************* Bottom of data **********************
F1=Help  F2=Split  F3=Exit  F5=Reset
F7=Backward  F8=Forward  F9=Swap  F12=Cancel
```

1. The control unit serial number is defined in the IODF, but no sensed data is available on the active system.

2. No control unit serial number is defined in the IODF, but the sensed data of the active system is available. To confirm the sensed data, and to define them in the IODF, press Enter.

3. The control unit serial numbers that are defined in the IODF, and that are sensed are different. Press Enter, to overwrite the defined data by the sensed data.

**Note:** The sensed values can only be blanked out or left unchanged. Blank out the sensed values, if you don’t want to change the defined IODF values. To confirm priming, press Enter. Use the F12=Cancel key, if you don’t want to use the sensed values, and to leave the list.

### Deleting control units

You can delete the definition of a control unit using the **Delete** action from the context menu (or action code `d`). Deleting a control unit means that all connections to channel paths, switches, and I/O devices are also deleted; these objects are not deleted.
Working with devices

Operating systems need I/O device data to address the devices. The CSS also needs the data to provide the required information for performing I/O operations to a specific device.

Four-Digit Device Numbers: Since MVS/ESA SP Version 5, HCD has supported the definition of four digits (numbers higher than ‘0FFF’) for device numbers for the MVS operating system. The four-digit device numbers make it easier for large installations to use unique device numbers across their installation. The device numbers for MVS/ESA SP 4.3 and lower versions are still restricted to three digits. If you use four-digit definitions, these are ignored when you IPL or dynamically activate an MVS/ESA SP 4.2 or 4.3 system. The software products installed need to support four-digit device numbers as well.

Defining devices

You need three steps to define an I/O device:
- Define device characteristics and control unit connection
- Define CSS-related definitions for a device
- Define OS-related definitions for a device (including EDT and esoteric group assignment - MVS-type only).

Before you define a device that should be defined to an operating system and to the channel subsystem (CSS), you must have defined the operating system configuration, processor, channel path, and control unit. HCD omits some steps if data is missing. For example:
- You cannot define the processor data for the device if the device is not attached to a control unit or the control unit is not attached to a processor.
- You cannot define the EDT/esoteric group data for the device until you have defined an EDT for the OS.

Defining device data

1. On the Primary Task Selection panel, select Define, modify, or view configuration data and on the resulting panel, select I/O devices. HCD displays an initial I/O Device List where devices with consecutive device numbers having the same definitions are automatically grouped together. Figure 56 on page 139.
A device group is shown as device number, range. A range value of one (1) is not explicitly shown. For example, the entry '0002,4 3390A' indicates a device group of four devices of type 3390A with consecutive device numbers from 0002 through 0005. Using action **Work with single I/O devices** from the context menu (or action code **SF**) displays the I/O Device list showing all single devices defined in the IODF, with all device groups resolved [Figure 57 on page 140].

Figure 56. I/O Device List with device groups

The # sign in front of a row indicates that this row is disabled. You cannot modify or delete it. In the example from Figure 56, you can see four devices of type CFS that are used for coupling facility connections.

If you scroll to the right in the I/O Device list, you would see additional columns **Serial-#**, **Description** and **VOLSER**.
Columns CSS and OS state the number of channel subsystems and operating systems accessing the device. If the I/O Device List is called from the Processor List or Channel Subsystem List, the number in the IM column states how many partitions (images) of the selected processor or channel subsystem are accessing the device. For basic processors this value is one.

If the I/O Device List is called from either

- the Operating System Configuration List
- the Processor List for SMP processors
- the Channel Subsystem List for XMP processors,

using action Work with attached devices from the context menu (or action code U), then the list contains an additional column SS which indicates, if applicable, in which subchannel set the device should be placed.

2. Use F11=Add to add I/O devices. The data-entry fields are shown in the following figure, with sample data:
In the Device number field, you can use the F4=Prompt key to have a list displayed containing unused device number ranges. If you select a proposal from this list, HCD fills Device number and Number of devices with the selected values.

The Add Device panel can also be used to specify the control units the devices are connected to.

Since VM dummy devices are definable with an arbitrary device type, a device with an unknown device type is accepted by HCD. It is treated like an unsupported device with the device type DUMMY. For MVS-type systems, you have to explicitly define the device as DUMMY.

Defining multiple devices in one step: You can define, in one operation, a group of I/O devices of the same type and with consecutive device numbers. You define the group by specifying the first device number and the number of devices in the group. Then HCD applies the definition to all devices in the group. On the I/O Device List, you can type over the values that should be different.

Use and definition of serial number of device: HCD allows you to assign the same device number to more than one I/O device; that is, device numbers alone do not uniquely identify a device in an IODF. To clearly identify devices, HCD keeps track of each occurrence of the same device number by appending an internal suffix to the device number.

When activating a configuration dynamically, HCD might be unable to determine whether certain I/O devices in the currently active IODF and the IODF to be activated are physically the same. This may happen, if the new IODF was not created by copying or updating the current IODF but was newly created by migrating with IOCP or using the HCD dialog. In this case HCD is unable to determine which of the devices are physically identical.

To avoid problems when activating a configuration dynamically, you should check if more than one device uses the same device number attached to the same control units in the current IODF and in the newly created (not copied) IODF. If so, specify the same serial number for the devices that HCD should treat as physically the same.
Defining CSS-related definitions for a device

If you have defined a connection to a control unit on the Add Device panel, and the control unit is connected to a processor, then HCD displays the Device / Processor Definition panel (Figure 59) that shows the processors to which the control units are attached.

On the Device / Processor Definition panel you can proceed in two ways:

- You can specify the CSS-related definitions directly by typing over the fields in each column. If you want to specify an explicit device candidate list for a device, type ‘yes’ into column Device Candidate List - Explicit. This leads you to panel Define Device Candidate List (Figure 61 on page 144).
- You can select a processor and press the Enter key. The Define Device / Processor panel is displayed (Figure 60). From this panel you can edit the same values as shown in the Device / Processor Definition panel.

Defining the subchannel set for a device: Starting with z9 EC processors, each channel subsystem contains more than one subchannel set (SS 0, SS 1), where you can place the devices. Starting with z/OS V1R7 HCD, you can place PAV alias devices (types 3380A and 3390A) into an alternative subchannel set. In SS 0, you can place 63.75K devices, and in SS 1 you can place 64K-1 PAV alias devices.

You can specify the subchannel set ID for a device either in column SS of Figure 59 or in field Subchannel set ID of Figure 60.
**Note:**

HCD messages that refer to a device in a subchannel set with a subchannel set ID > 0 will display the device number in the format \( n-devnumber \) where \( n \) is the subchannel set ID. For example, the device 1234 located in subchannel set 1 will show up as 1-1234. A device 4567 in subchannel set 0 will further on be shown as 4567.

**Rules for placing devices into subchannel sets:**

Observe the following rules and recommendations when working with different subchannel sets:

- There is no required correspondence between device numbers in the subchannel sets. For example,
  - devices in the range 8000-807F in SS0
  - devices in the range 8000-807F in SS1 (PAV alias devices)

  may relate to completely separate devices. However, you can use this feature to have PAV base and aliases in different subchannel sets, but with the same device numbers.

- Unit addresses of base and alias devices on a single control unit must be unique. These cannot be duplicated across subchannel sets. So if you want to define the PAV base and alias devices in the range 8000-807F in different subchannel sets, but on the same control unit, you can define them like follows:
  - base devices, range 8000-807F in SS0, unit address 00-7F (CU number 8000)
  - alias devices, range 8000-807F in SS1, unit address 80-FF (CU number 8000)

- You can use dynamic reconfiguration to move PAV alias devices to SS 1.

**Restricting Partition Access for Devices:** You can restrict logical partition access to an I/O device on a shared channel path by using the explicit device candidate list to select which logical partitions can access that I/O device. On the Define Device / Processor panel enter Yes or No in the Explicit device candidate list field to specify whether you want to restrict logical partition access to an I/O device:

- A No specifies that all logical partitions can access this I/O device. No is the default; all logical partitions are in this I/O device's candidate list.
- A Yes specifies that only your selected logical partitions can access this I/O device. Note that the partition must also be in the channel path access or candidate list to access the device. On the Define Device Candidate List, place a slash (/) character to the left of each selected Partition Name.

If you specify Yes in the Explicit device candidate list field, the following panel is displayed, showing possible candidate partitions:
A Yes in the Reachable column indicates that the device can be reached from the respective partition, through at least one physical channel. You can only include reachable partitions into the explicit device candidate list by typing a slash (/) into the action column. Deleting the slash means to remove the respective partition from the device candidate list.

**Null device candidate list for XMP processors:** If devices are connected to a control unit which is shared between multiple channel subsystems, some (not all) of these devices may specify an empty (or null) device candidate list for one or more CSSs. You create a null device candidate list for a device either by deselecting all candidate partitions from an existing list or by not selecting any partition for a new list.

If you define a null device candidate list of a device for a certain CSS, then no partition of this CSS may have access to the device. If you define an explicit device candidate list for a device, the **Device / Processor Definition** panel ([Figure 59 on page 142](#)) indicates whether this candidate list is a null device candidate list in column **Device Candidate List - Null**. If no partition is allowed to have access to the device, value Yes is shown, otherwise value No. This field is left blank if no explicit device candidate list exists for the selected device (which is the default when creating new devices).

**Defining OS-related definitions for a device**

1. After pressing the Enter key on the **Define Device / Processor** panel, the **Device / Processor Definition** panel is displayed again. Select another processor or press the Enter key again to display the **Define Device to Operating System Configuration** panel that shows all the defined OS configurations.
Select an operating system and the _Select (connect/change)_ action from the context menu (or action code 5).

As described in “Defining the subchannel set for a device” on page 142, starting with 2094 (z9 EC) processors, you can place PAV alias devices (types 3380A and 3390A) into SS 1.

If you define a PAV alias device, as shown in our example from Figure 58 on page 141, HCD displays the _Specify Subchannel Set ID_ panel that asks for the subchannel set where you want to place the device. The default depends on the value given for the corresponding CSS definition.

---

**Define Device Parameters / Features**

<table>
<thead>
<tr>
<th>Parameter/Feature</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WLMPAV</td>
<td>Yes</td>
<td>Device supports work load manager</td>
</tr>
</tbody>
</table>

---

Figure 62. Specify Subchannel Set ID

2. Pressing Enter on the dialog from Figure 62 brings you to the following panel where you can now define the data about device parameters and features that are required by the operating system configuration.

The Parameter/Feature fields vary depending on the I/O device type and operating system type.

A plus sign (+) in the Value column indicates that you may use F4=Prompt to get a list of possible values for the parameter/feature in the same row. Note that not all parameters are promptable.

A Y in the R column indicates that a value for the parameter/feature in the same row is required.

You accomplish the change by accepting the default values or by changing the Value entries and pressing the Enter key. The default values are set in the UIM for the device type. For parameters you can specify different default values via the OS_PARM_DEFAULT keyword in the HCD profile.

3. For eligible devices, after you have defined the device parameter and feature data and pressed the Enter key, HCD displays the _Assign/Unassign Device to_
4. On the Assign/Unassign Devices to Esoterics panel, overwrite the values in the Assigned column to assign (Yes) or unassign (No) devices to the selected esoterics.

If you do not want to assign a complete group of devices, you can limit the range by specifying a starting number and the number of devices. If you omit the number of devices, 1 is assumed.

**Changing devices**

To change device data, you have to follow the same panel flow as for defining a device:
1. Changing device and control unit definitions
2. Changing CSS-related definitions
3. Changing OS-related definitions

The following steps describe the panel flow and where you can change which data.
1. On the I/O Device List, select a device or a group of devices and the **Change** action from the context menu (or action code [c]). HCD shows the following panel:
Depending on whether you invoke this action for a single device or a group of
devices, the line Number of devices shows how many devices are affected by
the change.

2. On the following Change Device Definition panel you can change device and
control unit definitions such as:
   • Serial number
   • Description
   • Control unit connections
   • Volume serial number

Note: You can also change these definitions as well as the device type by
simply typing over the appropriate fields on the Device List.

3. After pressing the Enter key, the Device / Processor Definition panel is
displayed. Select a processor and press the Enter key to change the following
CSS-related definitions:
   • Subchannel set ID
   • Unit address
   • Time-Out
   • STADET
   • Preferred CHPID
   • Explicit device candidate list

4. After pressing the Enter key twice, the Define Device to Operating System
Configuration panel is displayed. Select an operating system and the Select
(connect/change) action from the context menu (or action code 5 ) if you want
to change the following OS-related definitions:
   • Parameters/Features
   • Assignments to esoterics

5. After pressing the Enter key again, the Assign/Unassign Device to Esoteric
panel is displayed. If you want to change the assignment of devices to
esoterics, type over the values in the Assigned column by either Yes or No.

6. Press the Enter key twice to return to the I/O Device List.
Changing CSS-related definitions of a group of devices

You can change CSS-related definitions of a group of devices using the CSS group change action. This helps you, for example, to attach a group of DASDs to another control unit. To do this, the devices to be changed must be in the same device group, that is, they must all be of type, for example, DASD or TAPE.

1. On the I/O Device List select one or more devices and use the CSS group change action from the context menu (or action code /SF580000g/SF590000). The Change Device Group panel is displayed.

HCD displays the definition of the first device in the group. You can modify this definition and HCD applies the definition to all devices in the group.

2. After pressing the Enter key, HCD displays the Change Device Group / Processor Definition panel, where you can select the processors for which you want to change the CSS-related definitions. For an example of this panel, see page [142] that shows the Device / Processor Definition panel which is similar to the one you see here.

Changing esoterics for a group of devices

For a description of how to change esoterics for multiple devices, refer to “Adding devices to esoterics” on page 89.

Changing OS-related definitions of a group of devices

You can change OS-related definitions for a group of devices using the OS group change action (or action code /SF580000o/SF590000). This helps you, for example, to attach a group of devices to another operating system. The device parameter/features will be the same for all devices in the group.
If you want to change OS-related definitions for PAV devices, HCD displays a similar dialog as shown in Figure 62 on page 145 which lets you change or specify the subchannel set ID where to place the device or the device group.

1. On the I/O Device List for single devices select one or more devices and use the **OS group change** action from the context menu (or action code \(\text{SF580000o/SF590000}\)). HCD displays the Change Device Group / Operating System Configuration panel.

   ┌─────────── Change Device Group / Operating System Configuration ────────────┐
   │ Row 1 of 1 │
   │ Command ===> ___________________________________________ Scroll ===> PAGE │
   │ Select OSs to connect or disconnect devices, then press Enter. │
   │ / Config. ID OS Type Description Defined │
   │ _ OPSYS01 MVS MVS or z/OS operating system Yes │
   │ ***************************** Bottom of data ****************************** │
   │ F1=Help F2=Split F3=Exit F4=Prompt F5=Reset │
   │ F7=Backward F8=Forward F9=Swap F12=Cancel F22=Command │

2. Select the operating system to which you want to attach the group of devices and the **Select (connect/change)** action from the context menu (or action code \(\text{SF580000s/SF590000}\)). A panel like the following one is displayed.

   ┌───────────────── Define Device Group Parameters / Features ─────────────────┐
   │ Row 1 of 4 │
   │ Command ===> ___________________________________________ Scroll ===> PAGE │
   │ Specify or revise the values below. │
   │ Configuration ID . : OPSYS01 MVS or z/OS operating System │
   │ Parameter/ Feature Value P Req. Description │
   │ ADAPTER TYPE7 + Yes Channel adapter type │
   │ OFFLINE No Device considered online or offline at IPL │
   │ DYNAMIC Yes Device supports dynamic configuration │
   │ OWNER VTAM + Subsystem or access method using the device │
   │ ***************************** Bottom of data ****************************** │
   │ F1=Help F2=Split F3=Exit F4=Prompt F5=Reset │
   │ F7=Backward F8=Forward F9=Swap F12=Cancel F22=Command │

3. You accomplish the change by accepting the default values or by changing the **Value** entries and pressing the Enter key.

   The specified device parameters/features are applied to all devices of the group.

**Changing the DYNAMIC, LOCANY or OFFLINE parameter of a group of devices**

```
Select type of op
1. Operating work connected sys-attached
equipments
2. Switches powerswitch control mode
3. Processes channel path partitions channel
4. Control units
5. I/O devices
```

Chapter 6. How to define, modify, or view a configuration 149
You can change the DYNAMIC, LOCANY or OFFLINE parameter of a group of devices using the **Attribute group change** action. This function helps you to change parameters for a group of devices without having to use the **Change** action for each device individually.

This function can only be invoked from the I/O Device List accessible from the Operating System Configuration List.

1. On the Primary Task Selection panel, select **Define, modify, or view configuration data** and on the resulting panel the object **Operating system configurations**.
2. Select an operating system and select the **Work with attached Devices** action from the context menu (or action code /SF580000u/SF590000). HCD displays the I/O Device List.
3. Select one or more devices on the I/O Device List and the **Attribute Group change** action from the context menu (or action code /SF580000e/SF590000). HCD displays the Attribute Group Change panel:

```
<table>
<thead>
<tr>
<th>For all devices in the selected group, choose whether ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Allow dynamic configuration ............ DYNAMIC=YES</td>
</tr>
<tr>
<td>2. Do not allow dynamic configuration .... DYNAMIC=NO</td>
</tr>
<tr>
<td>3. UCB can reside in 31 bit storage ...... LOCANY=YES</td>
</tr>
<tr>
<td>4. UCB can not reside in 31 bit storage .. LOCANY=NO</td>
</tr>
<tr>
<td>5. Device is set offline at IPL .......... OFFLINE=YES</td>
</tr>
<tr>
<td>6. Device is set online at IPL ........... OFFLINE=NO</td>
</tr>
</tbody>
</table>
```

Select the appropriate parameter.

HCD only changes the single parameter for all devices of the group, leaving the other parameters/features of the group unchanged.

### Changing type/model of a group of devices

You can change the type or model for a group of devices using the **Device type group change** action. However, you have to make sure that all devices to be changed in one step have the same device type and model. The control units the devices are attached to, have to support the attachment of the new device type as well, and required parameters have to be identical. The new device type has to be supported by the same operating system type.

1. Select one or more devices on the I/O Device List.
2. Use the **Device type group change** action from the context menu (or action code /SF580000t/SF590000). HCD displays the Device Type Group Change panel.
Specify a new device type-model.

Changing the subchannel set placement for a group of devices

You can change the placement of PAV alias devices any time, for example, if you want to migrate PAV alias devices into a subchannel set of a new processor. From the I/O Device List showing device groups or single devices, use action Subchannel Set ID group change from the context menu (or action code $n$). HCD displays the following dialog where you can specify the new ID of the subchannel set.

Note: When defining or changing the subchannel set placement for devices, you need to observe certain rules. For more information, read “Defining CSS-related definitions for a device” on page 142 and refer to the $z/OS HCD$ Planning.

If at least one of the selected devices has a connection defined to a processor supporting multiple subchannel sets, HCD displays a dialog where you can select from the eligible channel subsystems where to move the devices.
Also, if at least one of the selected devices has a connection defined to an operating system configuration, HCD displays a dialog listing all OS configurations that have connections to any of the selected devices. You can select all OS configurations for which you want to change the subchannel set ID for the selected devices.

```
Figure 64. Eligible Channel Subsystems

Select all channel subsystems for which the subchannel set ID has to be changed for all selected devices that have a connection to them.

/ Proc.CSSID Description
  - MSSPROC1.0 CSS0 of MSSPROC1
  - TSPROC1.0 CSS0 of TSPROC1
  - TSPROC1.1 CSS1 of TSPROC1

*************** Bottom of data ***************

F1=Help    F2=Split    F3=Exit
F7=Backward F8=Forward F9=Swap
F12=Cancel  F22=Command
```

```
Figure 65. Eligible Operating System Configurations

Select all operating system configurations for which the subchannel set ID has to be changed for all selected devices that have a connection to them.

/ Config.ID Type Description
  - ZOS17    MVS first z/OS 1.7 operating system
  - Z17SCND  MVS second z/OS 1.7 operating system

*************** Bottom of data ***************

F1=Help    F2=Split    F3=Exit    F7=Backward    F8=Forward
F9=Swap    F12=Cancel  F22=Command
```
Changing the device number

To change the number of a device:

1. Remove the connections to the control units for the devices to be changed as follows:
   a. On the I/O Device List, select the devices to be changed and the CSS group change action from the context menu (or action code /SF580000g/SF590000). The Change Device Group panel is displayed.
   b. Remove the control unit numbers from the panel and press the Enter key.

2. On the I/O Device List, select the device and the Add like action from the context menu (or action code /SF580000a/SF590000). The Add Device panel is displayed.

3. Specify the new number for the device and the control unit numbers to which the devices are to be attached. Press the Enter key. HCD now displays a series of panels showing the settings of the previously selected device (the one to be changed). The settings are propagated to the new devices. Press the Enter key until HCD redisplays the I/O Device List now showing the new device.

4. Delete the old device by selecting the device and selecting the Delete action from the context menu (or action code /SF580000d/SF590000).

Disconnecting devices from an operating system

Perform the following steps to disconnect a device from an operating system.

1. On the I/O Device List select a device and the Change action from the context menu (or action code /SF580000c/SF590000).

2. On the following Change Device Definition panel, press the Enter key.
3. On the following Device / Processor Definition panel, press the Enter key once again. HCD displays the Define Device to Operating System Configuration panel.

```
│ Row 1 of 2 │ Define Device to Operating System Configuration │
├───────────┼───────────────────────────────────┤
│ Device number . : 01D1 Number of devices : 1 │
│ Device type . : 3390 │
│ / Config. ID Type SS Description Defined │
│ OPSYS01 MVS MVS or z/OS operating system Yes │
│ OPSYS02 VM z/VM operating system │
└───────────────────────────────────┴───────────────────────────────────┘
```

4. On the Define Device to Operating System Configuration panel select an operating system and the Disconnect from OS action from the context menu (or action code R). The Define Device to Operating System Configuration panel is displayed again without showing a Yes in the Defined column.

**Disconnecting multiple devices from an operating system:**

If you want to disconnect multiple devices from one operating system in one step, open the I/O device list via the OS configuration list. On the I/O Device List, select one or multiple devices and use Disconnect from OS action from the context menu (or action code N).

**Showing or hiding parameter/feature definitions of devices**

You can define up to five parameters/features for a device that can be shown on the I/O Device List in addition to the default information. These parameters/features will be retained across sessions.

1. On the Operating System Configuration List, use the Work with attached devices action from the context menu (or action code U).
2. On the following I/O Device List, select the Show device parameters/features pull-down choice from the Show/Hide action bar (no action code available).
3. On the following Device Parameters/Features Profile panel, you can specify up to five parameters/features that will be displayed on the I/O Device List. HCD saves your settings across sessions.
4. On the I/O Device List, use the F20=Right key to scroll to the rightmost part of the panel, where the information is displayed. Note that the DYNAMIC parameter and LOCANY parameter are default information that is also shown on the leftmost part of the I/O Device List in columns D and L.

You can filter the shown devices by device parameters and features using the Set Filter function. On the Filter I/O Device List, you can specify a value for any displayed parameter/feature you want to use for filtering.

In case you no longer need the parameters/features to be displayed, you use Hide device parameters/features pull-down choice from the Show/Hide action bar on the I/O Device List.

**Priming device data**
You can prime your I/O configuration in a work IODF with the device serial numbers and volume serial numbers (VOLSER) for the active processor. For the prerequisites for this function refer to “Prerequisites” on page 9.

To prime, select the action **Prime serial number and VOLSER** from the context menu (or action code **1**) on the I/O Device List.

The Confirm Priming Device Data List shows the selected devices with the sensed data for the device types and serial numbers, and their corresponding definitions in the IODF. For DASD devices, the sensed VOLSER is also shown on this panel.

---

1. The values that are defined in the IODF and that are sensed are different. Press Enter, to overwrite the defined data by the sensed data.

2. No values are defined in the IODF, but the sensed data of the active system is available. To confirm the sensed data, and to define them in the IODF, press Enter.

**Note:** The sensed values can only be blanked out or left unchanged.

Blank out the sensed values, if you don’t want to change the defined IODF values.

To confirm priming, press Enter.

Use the F12=Cancel key, if you don’t want to use the sensed values, and to leave the list.
Deleting devices

You can delete the definition of a device or a device group using the **Delete** action from the context menu (or action code `[d]`). If you delete a device, all connections to the operating system including esoterics and EDTs are also deleted.

Working with operating system consoles

The following procedure describes how to specify which devices MVS can use as NIP consoles and which devices VM can use as VM consoles. Before you can define consoles you must have defined these I/O devices to the operating system.

1. On the Primary Task Selection panel, select **Define, modify, or view configuration data** and on the resulting panel the object **Operating system configurations**. HCD displays the Operating System Configuration List showing all OS configurations currently defined in the IODF.

2. Select an OS configuration and the **Work with consoles** action from the context menu (or action code `[w]`). HCD displays the NIP Console List or VM Console List (depending on the type of the selected operating system).

3. Use F11=Add to define each console. The following panel is displayed:
The order number is the sequence the consoles are used by the operating system.

**Changing operating system consoles**

You can change the order number of an operating system console by just typing over the corresponding column or by using the *Change* action from the context menu (or action code /SF580000c/SF590000) on the Console List.

**Deleting operating system consoles**

You can delete the definition of an operating system console using the *Delete* action from the context menu (or action code /SF580000d/SF590000) on the Console List. The devices are not deleted.

**Working on IODFs enabled for multi-user access**

When multiple users concurrently use the same IODF, a user's changes are not immediately refreshed in the views of the other users. However, each user has a consistent view of the data either from the initial access to the IODF or after each last update that he had applied to the IODF.

In a few scenarios, this information unit demonstrates how HCD applies these rules when two users, *UserA* and *UserB* concurrently view or update an IODF:

- "Simultaneously updating and viewing an IODF" on page 159
- "Concurrently updating an IODF" on page 159
- "Immediately reflecting changes during concurrent updates" on page 160.
Simultaneously updating and viewing an IODF

Imagine that both users UserA and UserB invoke the Channel Subsystem List of the same IODF. UserA wants to delete channel subsystems CSS 1 and CSS 3, and UserB wants to work with the partitions of CSS 1.

Both users press Enter. While UserA sees the updated Channel Subsystem List, UserB sees the partitions of the meanwhile deleted CSS 1, because he still views the state of the IODF as loaded from storage. He will get a refreshed view after applying an update on the IODF.

Concurrently updating an IODF

Both users UserA and UserB start on the Channel Subsystem List of the same IODF. UserA invokes action Change on CSS 1 and UserB updates Maximum Devices in SS0 for CSS2 and CSS3 from 65280 to 64512 and additionally invokes action Delete on the same CSS 1 maybe a few seconds later.
HCD displays panel *Change Channel Subsystem* for *UserA* while *UserB* receives message CBDA340I. When *UserB* returns from the *Message List* to the *Channel Subsystem List*, the panel is not refreshed and *UserB*'s updates are kept on the screen. Thus, *UserB* can retry his update request several times until *UserA* releases the lock on the IODF.

**Immediately reflecting changes during concurrent updates**

Both users *UserA* and *UserB* start on the *Channel Subsystem List* of the same IODF. *UserA* invokes action *Delete* on CSS 1 and CSS 2, locking the IODF, and *UserB* at the same time invokes action *Change* on several CSSs.

Even after *UserA* has successfully deleted the two channel subsystems, *UserB* sees the *Channel Subsystem List* with all channel subsystems, because this is the state of the IODF as loaded from storage. Let us suppose, he wants to change channel subsystems CSS 1, CSS 2 and CSS 3. An appropriate message is displayed for each channel subsystem that is already deleted. When returning to the *Channel Subsystem List*, *UserB* sees the refreshed contents of this panel.
Viewing information

HCD offers several possibilities to view information that might be helpful when defining or maintaining your configuration data.

Viewing object definitions

To view information about objects that are defined in your configuration, select the Define, modify, or view configuration data option from the Primary Task Selection panel. From the resulting panel, select the desired object to get the appropriate list, on which you can use Work with object actions from the context menu that lead to further list panels that display other related objects. You can use these panels to define, modify, and view configuration data. See Figure 42 on page 80 on how to navigate to related list panels of different objects.

Viewing full definition of an object

You can ask HCD to display, for viewing only, the full definition of an object. This might include additional information that is not displayed on the action list panel.

To view object definitions select an object on an action list and use the View object definition action (or action code V).

This following lists offer this possibility:

- Processor List
- Channel Path List
- Control Unit List
- I/O Device List

Viewing additional object lists

Besides the action list panels shown in Figure 42 on page 80, there are list panels, on which you can only view information about objects without being able to change it. You can navigate to these lists using the View objects actions from the context menu. HCD offers View objects actions on the following panels:

- List Panel
- Channel Path List
- Partition List
- Operating System List
- Generics List

View
Connected switches
Attched channel paths
Attached control units
Attached devices
Generics
Devices
Graphical view
HCD offers you the possibility to view a graphical representation of the configuration.

Use the Create or view graphical configuration report on the Primary Task Selection panel to view the entire configuration. To view objects in context of their attached objects you can also select an object from an action list panel and use the View graphically action from the context menu (or action code ). The following object lists support this possibility:
- Channel Path List
- Control Unit List
- I/O Device List (only for devices that connect to a control unit)
- Partition List
- Switch List

For more information on how to view a graphical report, refer to “Create or view graphical configuration reports” on page 238.

Viewing logical control units
On a Control Unit List or I/O Device List, you can display the groups of logical control units that HCD has created to represent the physical control units defined in a configuration. Logical control units are used by the CSS to schedule the processing of I/O requests.

Viewing coupling facility information
HCD lets you view information for a specific CF channel path. You can view, for example, access and candidate list of selected channel paths and information on the CF control units and devices in a CF channel path connection.

On the CF Channel Path Connectivity List, select a channel path and one of the following actions:
- The View source channel path definition action displays the View Channel Path Definition panel for the source channel path of the CF connection.
- The View destination channel path def. action displays the View Channel Path Definition panel for the destination path of the CF connection.
- The View CF control unit and devices action displays the View CF Control Unit and Devices panel that shows the CF control unit number(s), the starting CF device number(s), and the range of devices defined for a CF connection.

Viewing CTC connections
HCD offers you the possibility to view and verify your CTC connections. You can view existing CTC connections including online diagnostic messages on the following lists:
- Processor List
- Partition List
- Channel Path List
- Control Unit List
- I/O Device List

The CTC Connection List lets you immediately verify whether your definitions are done correctly.
**ESCON channel-to-channel support**
An ESCON CTC connection requires a CTC channel at one end of the connection and a CNC or FCV channel at the other end of the connection. The two channels can be considered as communicating directly with each other in a peer-to-peer fashion. Each channel defines the channel at the other end of the CTC connection as an SCTC control unit. This is illustrated in Figure 66.

**FICON channel-to-channel support**
FCTC support differs from ESCON CTC support not only in the channel types used. The main differences are as follows:
- An FCTC connection is given via a FICON channel path on each side of the FCTC communication line.
- It is possible to have an FCTC connection between the LPARs of the same CEC via a single FICON channel path in a switched environment.

In order to be usable as an FCTC connection channel, a FICON channel path must be defined to an FCTC control unit which is connected to FCTC devices.

---

**Figure 66. CTC/CNC connection established using a dynamic connection**

The entry port of the channel at the other end of the CTC connection corresponds to the link address of the control unit representing the channel.

The CTC devices associated with the control units at both ends of the CTC connection may have different device numbers, but they must have the same unit address. The device type of both devices must be the same (for example, SCTC or BCTC).

**Restrictions applying to the CTC Connection List**
- HCD can only show CTC connections if the connected processors are defined in one IODF.
- For a switched SCTC connection, the CNC/FCV and CTC channel paths must be connected to the same ESCON director. CTC connections running via chained ESCON directors cannot be determined.
- CTC connections using a stand-alone CTC adapter cannot be shown.
- Point-to-point CTC connections are only recognized if the control units associated to a specific CTC connection have the same serial number defined.
**CTC connections with shared channels**

If your processor has MIF support you can share your channels among several partitions to save physical connections. The following figure shows you the case when a CNC channel is shared between two partitions. The CTC channel will need a separate control unit definition for each partition sharing the CNC channel. Each of these control unit definitions has the same destination link address but the control unit logical addresses (CUADD) must be different. The control unit logical address must correspond to the image number of the logical partition.

**Note:** If the target channel path is non-shared, either you must not specify a control unit logical address, or its value must be 0.

---

**Figure 67. CTC/CNC connection established using shared channels**

For further specification rules, refer to the IOCP User’s Guide for your processor.

**How to view CTC connections**

You can use action **View related CTC connections** (or action code 4) on the following lists:

- Processor List
- Partition List
- Channel Path List
- Control Unit List
- I/O Device List

A panel similar to the following one is displayed:
This panel shows the definitions of the CNC/FCV side in relation to the definitions of the CTC side, such as processor, partition, channel path, control unit, and device information.

**Incomplete CTC definitions:** If the CTC connection is not correctly defined, the fields on the CTC Connection List can be incomplete and an error message is shown. For example, G754 in column Msg. refers to message CBDG754I, which indicates that HCD cannot determine the connection, because no control units and devices match to the processor, partition, control unit, and device of the same row.

**Displaying more detailed information**

Scroll to the right to see more detailed information about the CTC side of the connection, such as channel path mode, switch information, detailed control unit and device information.

Scroll once again to the right to see the same detailed information for the CNC/FCV side of the connection.
### Filtering CTC definitions

To get a better overview of your CTC connections you can filter the list by specifying different filter criteria. Select action bar **Filter** and then **Set filter**. The following panel appears:

<table>
<thead>
<tr>
<th>Message ID ...</th>
<th>Device type...</th>
<th>Dynamic switch ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTC or FC side</td>
<td>CNC/FCV or FC side</td>
<td></td>
</tr>
<tr>
<td>Processor.CSSID</td>
<td>Processor.CSSID</td>
<td></td>
</tr>
<tr>
<td>Partition ...</td>
<td>Partition ...</td>
<td></td>
</tr>
<tr>
<td>CHPID ...</td>
<td>CHPID ...</td>
<td></td>
</tr>
<tr>
<td>CU number ...</td>
<td>CU number ...</td>
<td></td>
</tr>
<tr>
<td>Starting device no.</td>
<td>Starting device no.</td>
<td></td>
</tr>
<tr>
<td>Defined to OS ...</td>
<td>Defined to OS ...</td>
<td></td>
</tr>
</tbody>
</table>

### Printing CTC connection lists

You can also print the list panel using the SAVE command as described in “How to print list panels” on page 245.

### Displaying diagnostic messages

Select one or more CTC connections and press the Enter key to display diagnostic messages for the connections.

For each connection, HCD displays one message, even if the connection includes several errors. HCD displays the messages according to the following priority list:

1. **CBDG750I** Logical address (CUADD) is specified for CU @1, but CHPID @2 of processor @3 is not defined as shared.

2. **CBDG751I** Device type of device @1 connected to processor @2, CHPID @3 does not match with device type of device @4 on the other side.
3. CBDG752I Channel path type error. CHPID @1 of processor @2 is connected to a CHPID @3 of processor @4 with the same type.

4. CBDG753I Wrap around connection detected for processor @1 (partition @2) via CHPID @3 and CHPID @4.

A message list may look as follows. The messages are sorted by severity.

<table>
<thead>
<tr>
<th>Save</th>
<th>Query</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command ===&gt;</td>
<td>Message List</td>
<td>Scroll ===&gt; PAGE</td>
</tr>
<tr>
<td>Row 1 of11</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Messages are sorted by severity. Select one or more, then press Enter.

/ Sev Msg. ID Message Text
- E CBDG750I Logical address (CUADD) is specified for CU 1010, but CHPID 20 of processor PROC001A is not defined as shared.
- E CBDG752I Channel path type error. CHPID 20 of processor PROC001A is connected to CHPID 11 of processor PROC002 with the same type.
- W CBDG753I Wrap around connection detected for processor PROC002 via CHPID 11 and CHPID 13.
- I CBDG756I HCD cannot determine connection. CHPID 24 of processor PROC002 is connected via chained switches.
Chapter 7. How to work with switches

Overview

This information unit explains:

- The possibility of switch (ESCON Director or Fibre Channel switch) connections
- The advantages when you define switches with HCD
- How to work with switches (defining, changing, priming, deleting)
- How to work with connections to switches (channel paths, control units, priming switch port names and connections, actions on the Port List)
- How to work with switch configuration data (defining, changing, deleting)
- How to migrate, activate, and save switch configuration data

With the introduction of the ESCON architecture and its supporting hardware, ESCON Directors (switches) became an integral part of the configuration. With the FICON architecture, FC switches are supported.

You can define switches, switch connections and how the switches are physically cabled. A switch configuration, also called port matrix, defines how the various ports of the switch connect to each other. For example, the switch configuration defines whether a port is blocked, has a dedicated connection to another port, or whether dynamic connections to other ports are allowed or prohibited. In other words, the switch configuration defines the inside of a switch.

Possibilities of switch connections

ESCON Directors (ESCDs) enable either dynamic connections or dedicated connections. FC switches allow only the definition of dynamic connections.

HCD supports fabrics containing cascade switching using FICON switches. Such a fabric consists of two or more FICON switches.

Note: Other than for chained ESCON switches where the dynamic switch for a channel path specifies the switch containing the port address that is used as link address, in the FICON case the dynamic switch ID is always the channel path entry switch.

The following four figures illustrate the path types of switch connections for ESCON Directors or FC switches. For FC switches, only the first and the fourth configuration type is supported. Figure 68 on page 170 shows a configuration with a single switch; the entry switch is the dynamic switch. (The dynamic switch in HCD corresponds to the SWITCH keyword of the CHPID macroinstruction from the IOCP point of view.)
Figure 69 shows a configuration with two switches, where the entry switch is different from the dynamic switch. (The two switches are chained and the entry switch for the channel path has a dedicated connection.)

Figure 70 on page 171 shows a configuration with two switches, where the entry switch is the same as the dynamic switch. (The two switches are chained and the CU switch has a dedicated connection.)
Figure 71 shows a configuration with cascading switches. The fabric in this figure contains two cascading FICON switches. The link address 5904 specifies 59 as switch address and 04 as port address.

**Note:**

The switch address is unique within a fabric, but may occur also in other cascaded switch fabrics. However, as HCD has no knowledge of which switches are within the same fabric, it is highly recommended to assign unique switch addresses across fabrics, for example, by using the switch IDs as switch addresses.

Figure 71. Configuration with two cascading FICON switches
Advantages of switch definitions with HCD

You have the following advantages when you define switches with HCD:

- More rigorous validation of the configuration definition.
  If all switches between the channels and control units are defined, HCD can determine whether a valid path exists between the processor and control unit. For example, HCD can validate that the destination link address specified for a channel path is a valid port on the dynamic switch.

- The possibility to define several switch configurations for each switch.

- The possibility to define and activate switch configurations without leaving HCD (from the same workplace).
  If you have defined a switch configuration using HCD, you can activate the switch configuration without leaving HCD and do not need any knowledge about other operating system components that are involved in the activation process.

- The possibility to define a switch as CU, device, and switch as such, to:
  1. Migrate an active switch configuration from a switch, or a saved switch configuration from a switch file, or convert an ISPF table to a HCD switch configuration for later manipulation by HCD.
  2. Send switch configuration data from an IODF to a switch, making it the active port matrix, or save it in a switch file.

- Graphical configuration reports include switch connections.

Note: HCD supports a generic FC switch (type FCS) supporting port addresses 00 to FF. This switch type does not support a switch control unit and switch device and therefore can not be accessed by I/O Operations functions like migrate or activate switch configuration.

Defining switches

To define switches and their associated ports, you need to:

- define switch characteristics,
- define connections to channel paths, control units, and other switches,
- define switch configuration data (port matrix).

Working with switches

In this section, you can learn how to define, change, prime, and delete switches.

Defining switch characteristics

1. On the Primary Task Selection panel, select Define, modify, or view configuration data and on the resulting panel the object Switches. HCD displays the list of all switches currently defined in the IODF.
The Switch List (left part), Figure 72, lists one switch control unit and device. If there is more than one switch control unit and device, the list entry gets an indication (‘>’). With the F20=Right key, you can scroll to the right part of the Switch List. Up to five switch control units and devices can be shown. If there are more, an indication is given for the corresponding entry (‘Yes’ in column ‘More?’ on the right part of the Switch List). These additional switch control units and devices can be viewed, for example, on the Port List for port FE.

Figure 73. Switch List (right part)

```
<p>| | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>01 2032</td>
<td>0701 0701 0702 0703 0704 0705 0706 0707 0708 0709 0710 0711</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>AA 9032</td>
<td>001A 001A 002A 002A 003A 003A 004A 004A 005A 005A 006A 006A</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>AB 9032</td>
<td>001B 001B 002B 002B 003B 003B 004B 004B 005B 005B 006B 006B</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>AC 9032</td>
<td>001C 001C 002C 002C 003C 003C 004C 004C 005C 005C 006C 006C</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>AD 9032</td>
<td>001D 001D 002D 002D 003D 003D 004D 004D 005D 005D 006D 006D</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>AE 9032-3</td>
<td>001E 001E 002E 002E 003E 003E 004E 004E 005E 005E 006E 006E</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>
```

2. Use the F11=Add key to add a new switch.
HCD allows you to specify the port range of a switch to be set to installed, if more ports are to be used than the minimum range. Specify the first and last port of the range you want to use. If you do not specify values for the Installed port range field, the hardware status of the minimum range of supported ports is set to installed.

In order to allow consistency checks for the configuration, when adding a new switch, you can optionally define a switch address for a FICON switch. You can also specify control unit numbers and device numbers for the switch.

On the Add Switch panel you can initially define up to five switch control units and devices for the switch. To define more than five switch control units and devices, or to add additional switch control units and devices later, you must use the control unit and device definition dialogs.

If you specify switch control units that do not yet exist, they are automatically added as new objects to the IODF, and are connected to the switch through the switch control unit port. In this case, you need to specify new switch devices. The switch devices are also automatically added as new objects to the IODF and connected to the switch through the switch control units. However, to complete the configuration path, you must attach the switch control units and switch devices to a processor, and then you can assign the switch devices to an operating system.

If the switch control units already exist, they are automatically connected to the control unit port on the newly defined switch. In this case, you do not need to specify switch devices. If you do, the switch devices must already exist and be attached to the designated switch control units.

Specified serial numbers or descriptions are also copied to the switch control units and switch device definitions.

3. After you press the Enter key, HCD displays the updated Switch List.
   - Connect the switch control units to the processor (which also connects the switch devices to the processor). To specify additional parameters use the Change action on the Control Unit List. For details, see “Defining processor attachment data” on page 131.
Connect the switch devices to the operating system. Use the Change action on the I/O Device List. For details, see “Changing devices” on page 146.

Changing switch data

To change the following switch characteristics, you can type over the columns on the Switch List, or you can use the Change action from the context menu (or action code /SF580000c/SF590000) on the Switch List:

- Switch type
- Serial number and description

When you change the type, serial number, or description of the switch, the control units, and devices attached to the switch are also updated.

![Diagram of Switch List]

Figure 75. Change Switch Definition

Priming switch data

You can prime your I/O configuration in a work IODF with the switch serial number for the active processor. For the prerequisites for this function see “Prerequisites” on page 9.

To prime, select the action Prime serial number from the context menu (or action code /SF580000i/SF590000) on the Switch List.
The Confirm Priming Switch List shows the selected switches with the sensed data for the switch types, serial numbers, switch control units, and switch device numbers, and their corresponding definitions in the IODF.

**Note:** Only one defined switch control unit and one defined switch device is shown even if several have been defined. If the sensed switch control unit and device is one that has been defined, it will be displayed. Otherwise the defined switch control unit and device with the lowest control unit number will be displayed.

An update of the switch serial number also updates the serial number of the corresponding switch control units and switch devices.

The sensed data for the switch serial numbers are shown on the Confirmation panel, and can be accepted, or rejected before being incorporated into the IODF. If a value is blanked out, the defined IODF value is not changed. If you use the F12=Cancel key, none of the sensed values is used.

### Moving ports

When selecting **Move Ports**, HCD offers the possibility of moving control unit, channel path or switch port connections on the same switch or from other switches to the selected target switch.

To perform this action, select the target switch for a port move action on the Switch List (see Figure 72 on page 175). The panel Actions on selected switches appears and the action **Move Ports** can be selected. You can also reach this panel directly by typing in the action code **x** next to the appropriate switch in the Switch List menu. This brings up the following panel Move Ports to a Target.
Switch.

This panel contains data entry fields for the ports to be moved. It is also possible to move a range of ports from a switch to the target switch occupying subsequent port addresses starting with the target port address specified. The target switch field in this panel has been preset and cannot be changed.

Depending on the context, HCD performs the following as part of the Move Ports action:

- Copies the attributes of the source port to the target port (and set the target port to installed, if necessary).
- Disconnects all source ports from the connected units.
- Connects all target ports to the units previously connected to the source ports.
- Copies existing port configurations of the source port if the move is on the same switch and port configurations exist. The source port configurations will be set to default, i.e., all dynamic and dedicated connections are reset.
- Changes the dynamic switch of the connected channel path to the target switch if the source switch serves as a dynamic switch and the target switch is different from the source switch.
- Changes the link address to the target port if the source port serves as a link address to a channel path connection for a control unit and is connected to a control unit or another switch.

Note: HCD does not perform any checks on whether the user also moves implied ports as well. For example, if a channel path is moved to another switch, the control units that are connected to the channel path must also be moved. Moving a control unit may imply that connected channel paths must be moved as well. If not all implied ports are moved, the configuration may become invalid and a validation error will be shown either during the Move Ports action or later during the Build Production IODF action.
Note: If the target switch has switch configurations defined and the port move occurs between different switches, then the switch configurations must be adapted after the port move action.

The Move Ports action is especially of value when consolidating switches or installing new switches.

Deleting switches

You can delete the complete definition of a switch or switch configuration by using the Delete action from the context menu (or action code D) on the Switch List or Switch Configuration List. This also deletes the connections from the ports to channel paths, control units, and other switches. The link address and dynamic switch definitions for a channel path are not deleted.

The Confirm Delete Switch panel shows all the switch control units and devices that will be deleted with the switch. If you don’t want them to be deleted with the switch, disconnect the control units from port FE of the switch, before you delete the switch.

Working with connections to switches

In this section, you will learn about how to define connections to switches, for example channel paths, control units, other switches, how to prime switch port data, and what kind of additional actions is offered on the Port List.
Defining connections to switches

You can connect the following objects to a switch starting from the Switch List:

- Channel paths
- Control Units
- Other switches

You can also define connections from the objects to the switch when defining the object themselves. See Chapter 6, “How to define, modify, or view a configuration,” on page 177 for a description how to define the objects.

Connecting a channel path

The following procedure describes how to define a connection between a channel path and a switch starting from the Switch List.

1. On the Switch List, select the switch and the Work with ports action from the context menu (or action code /SF580000p/SF590000). The Port List is displayed.

   Ports which show value Y in column 0 indicate that they are occupied by a processor, control unit or switch that is not defined in the accessed IODF.

2. Select a port and the Connect to channel path action from the context menu (or action code /SF580000p/SF590000).

   Occupied ports cannot be connected. However, you may change the Occupied status of a port by overwriting the Occupied indicator.
On the resulting Connect to Channel Path panel specify the target processor ID and channel path ID.

```
Connect to Channel Path

Specify the following values.
Switch ID : 99 Port : C4
Processor ID . . . . . __ __ +
Channel Subsystem ID : _ _ +
Channel path ID : _ _ +
F1=Help F2=Split F3=Exit
F4=Prompt F5=Reset F9=Swap
F12=Cancel
```

**Note:** If you selected a spanned physical channel path as connection target, HCD connects the port to all of the channel's instances across all channel subsystems. You can see the result of your connection action in the Port List (Figure 79 on page 179).

### Connecting a control unit

The following procedure describes how to define a connection between a control unit and a switch starting from the Switch List.

1. On the Switch List, select the switch and the **Work with ports** from the context menu (or action code /SF580000p/SF590000). The Port List is displayed (see Figure 79 on page 179).

2. Select a port and the **Connect to control unit** action from the context menu (or action code /).

```
Connect to Control Unit

Specify the following values.
Switch ID : 99 Port : C4
Control unit numbers : 00E1 __ __ __ __ __ __ __ __ __ __ __
F1=Help F2=Split F3=Exit F4=Prompt F5=Reset F9=Swap
F12=Cancel
```

You can enter up to eight control unit numbers each time the panel is displayed.

3. Repeat defining connections for all control units connected to the switch.
Connecting another switch

The following procedure describes how to define a connection between a switch and another switch.

1. On the Switch List, select the switch and the Work with ports action from the context menu (or action code /SF580000p/SF590000). The Port List is displayed (see Figure 79 on page 179).

2. Select a port and the Connect to switch action from the context menu (or action code /SF580000w/SF590000).

3. Repeat defining connections for all other switches connected to the selected switch.

Priming switch port data

You can prime your I/O configuration in a work IODF with the switch port names, and connections for the active processor. For the prerequisites for this function refer to “Prerequisites” on page 9.

To prime, select the action Prime port name and connections from the context menu (or action code /SF580000i/SF590000) on the Port List.

The Confirm Priming Port Data List lists the selected ports with the sensed data for the port names, and for the connected control units, or switches, or the connections to channel paths of the active processor. Their corresponding definitions in the IODF are shown in the line below the sensed data on the panel.
You get sensed data for connected channel paths only if the processor definition in the IODF contains a serial number that matches the serial number of the active processor.

The sensed port names and connection data can be confirmed before being taken into the IODF. If a value is blanked out, the defined IODF value is not changed. If you use the F12=Cancel, or the F3=Exit key, none of the sensed values is used.

```
   Confirm Priming Port Data List                        Row 1 of 4
Command ===> _____________________________________________________________________ Scroll ===> CSR
Press Enter to confirm priming, or Cancel to leave the list. A blank value will not change the IODF definition.
Switch ID .. : 01
Port --- Sensed Port Name --- Sensed Connection ------- Defined Port Name --- Defined Connection -------
A1 200A-E          CU  200A          3990-6
A2 400B-CG
400B-CG          CU  520B          3990-3
A3 360A-00
360A-00          CU  360A          9343-0C4
A4 VMA(32)
VMA(32)          PR  VMBASIC CHPID 32 9672-R61
```

1. No defined data is available for the port connection on the active system. The sensed and defined port names are the same.
2. The sensed and defined port names are available. The port connection is defined in the IODF, but no sensed data is available on the active system.
3. No sensed port name data is available on the active system. The port connection shows differences of the switch type, but the sensed data is not taken in the IODF.
4. No sensed data is available for the port connection on the active system. The sensed and defined port names are the same.

**Additional actions on the port list**
Besides connecting channel paths, control units, and other switches to a switch, you can perform additional actions on the Port List. Most of these actions are also possible on the Port Matrix panel, this panel offers an alternative procedure. The following panel shows data after connecting a channel path, a control unit, and another switch as described in the previous sections.
A disabled marker # in the action entry field indicates that the field is nonselectable and the whole row is disabled for processing. This occurs if more than one object is attached to one port, for example, for spanned channels or when multiple control units are connected to the same port. Except for the first object attached to the port, all other objects are flagged with the # sign. For example, if the port is connected to more than one unit, a disconnect action specified in the selectable row of that port will lead to the display of another panel where you can select the unit(s) to be disconnected.

**Changing ports to installed or uninstalled**
The H column indicates whether the ports are installed (Y for Yes) or not (N for No). If you did not specify a range of ports to be set to installed while adding a switch, HCD automatically sets the minimum range of ports to installed at switch definition time.

You can set the port to installed or uninstalled by just typing over the Y or N value in the H column.

**Changing ports to occupied or not occupied**
The O column indicates whether a port is occupied (Y for Yes) or not (N for No) by a system external to the IODF.

You can change the port to Occupied or Not Occupied by just typing over the Y or N value in the O column.

**Establishing dedicated connections and blocking ports**
If no switch configurations are defined for a switch, the B and Ded Con columns for blocked indicator and port of dedicated connection are not shown. If switch configurations are defined, data of the first switch configuration (in alphabetical order) is displayed. To display this data for other switch configurations, use Select other switch configurations from the Option action bar choice.
You can change the blocked indicator and dedicated connections for the switch configuration displayed in the panel header by just typing over the values in the appropriate column. See also “Establishing dedicated connections” on page 185 and “Blocking ports” on page 185 for detailed explanations.

Note: You cannot establish dedicated connections for an FC switch.

Working with switch configurations

In this section you can learn about defining switch configuration data, changing the switch configuration IDs, and deleting switch configurations.

Defining switch configuration data

After defining the switch, you can define the switch configuration, that is the “inside” of the switch.

1. On the Switch List, select the switch and the **Work with switch configurations** action from the context menu (or action code $S$). HCD displays the Switch Configuration List containing all currently defined configurations for that particular switch.

<table>
<thead>
<tr>
<th>Goto Backup Query Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch Configuration List Row 1 of 1</td>
</tr>
<tr>
<td>Select one or more switch configurations, then press Enter. To add, use F11.</td>
</tr>
<tr>
<td>Switch ID ....: 99  First switch</td>
</tr>
</tbody>
</table>

   Switch Default / Config. ID Connection + Description
   - NIGHT Allow ________________________________________

2. Use F11=Add to add a new switch configuration. The data-entry fields are shown below, with sample data:

<table>
<thead>
<tr>
<th>Add or Repeat Switch Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify or revise the following values.</td>
</tr>
<tr>
<td>Switch ID ....... : 99</td>
</tr>
<tr>
<td>Switch configuration ID . BASIC___</td>
</tr>
<tr>
<td>Description ........</td>
</tr>
<tr>
<td>Default connection . . 1 . Allow</td>
</tr>
</tbody>
</table>
The Default connection field sets the default connection for all ports, either allowed or prohibited. Individual port connections can be reset on the Port Matrix panel described in the next step.

3. On the Switch Configuration List, select the switch configuration and the Work with port matrix action from the context menu (or action code 3). HCD displays the Port Matrix panel showing all ports currently installed on the switch.

Establishing dedicated connections
You can establish a dedicated connection between two ports by specifying the number of a port to which a dedicated connection is defined in the Ded Con column. After pressing the Enter key HCD completes the definition by mirroring the definition. For example, if you specify a dedicated connection in the row of port E0 to port E4, HCD establishes the same dedicated connection in the row of E4 to port E0.

A dedicated connection acts like a physically cabled connection between two ports. Establishing a dedicated connection is not the same as prohibiting all but one connection to a port. Dedicated connections are required to support communication through an ESCD with an ESCON Converter (ESCC), and to support chained ESCDs.

If you maintain switch configurations with HCD, you must define the required dedicated connections before you connect a CVC or CBY channel path to a switch port.

If you have alternate required dedicated actions in a configuration (for example, for backup purposes), you must define alternate switch configurations.

Blocking ports
You can block or unblock a port by just specifying a Y for Yes or N for No in the B column.

Dynamic connection ports
In our example the default switch connection is set to allowed. To prohibit a switch connection from, for example, EA to FA, scroll down until you see port EA and scroll right until you see port FA. Then type over the * symbol with a p below the heading Dynamic Connection Ports. After pressing the Enter key HCD
automatically mirrors the entries on the diagonal of the matrix. That means, HCD applies the same entry not only to the matrix element EA/FA but also to FA/EA.

To ensure a correct mirroring of the entries, press the Enter key each time you changed one complete row.

The following symbols can be shown below the heading Dynamic Connection Ports:
- A indicates that the dynamic connection is allowed.
- P indicates that the dynamic connection is prohibited.
- * indicates that the dynamic connection is set to the default connection attribute (shown in the instruction area on the top of the panel).
- \ indicates the intersection of a port’s column and row. (This is only shown for the matrix of an ESCON switch since the dynamic connection of an ESCD port to itself is prohibited and cannot be changed. A FICON switch, however, supports the definition of such a loopback port.)
- - indicates that one of the dynamic connection ports is not installed or supported.

To allow you a more comfortable scrolling in the matrix, use the FIND command. For example, type:
```
FIND EA, FA
```
to find the row of port EA and the column of port FA.

### Changing the switch configuration ID

To change the ID of a switch configuration, perform the following steps:
1. On the Switch List, select the switch and select the **Work with switch configuration** action from the context menu (or action code 5). HCD displays the Switch Configuration List.
2. On the Switch Configuration List select the switch configuration and the **Repeat (copy) switch configurations** action from the context menu (or action code R). The Repeat Switch Configuration panel is displayed.
3. Specify the new identifier for the switch configuration and press the Enter key. HCD displays the Switch Configuration List now showing the new switch configuration.

4. Delete the old switch configuration by selecting the switch configuration and the **Delete** action from the context menu (or action code `d`). HCD displays a confirmation panel before showing the updated Switch Configuration List.

**Generating a switch matrix**

When selecting **Generate Matrix**, HCD will define the content of the switch matrix according to the logical paths defined and the existing matrix will be replaced.

This is useful, if channel path - control unit connections have been added or changed and you want to define a switch configuration which considers all defined logical paths running through the selected switch.

To generate a matrix, perform the following steps:

1. On the Switch List select the switch and select the **Work with switch configuration** action from the context menu (or action code `w`). HCD displays the Switch Configuration List.

2. On the Switch Configuration List select a configuration (or action code `g`). HCD will issue an informational message requiring you to confirm your action.

   The successful generation of the matrix is confirmed by HCD.

When generating the switch configuration, all defined logical paths will be analyzed by HCD. In the case of chained connections, all possible paths of chaining switches will be determined by their respective switch configurations. Therefore, it is necessary, that you select a switch configuration for each chained switch, for which more than one switch matrix has been defined. In this case, the following panel
Select Active Switch Configurations

Row 1 of 2

Command ==> ___________________________________________ Scroll ==> PAGE

For each switch select one to be used as context to generate a switch matrix.

Switch ID: 01 Configuration ID: SC1

Switch Configuration Description

<table>
<thead>
<tr>
<th>ID</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>SC1</td>
</tr>
<tr>
<td>02</td>
<td>SC2</td>
</tr>
</tbody>
</table>

will be displayed allowing selection of the related switch configurations which are considered for the *Generate matrix* action.

The following rules are applied when generating a matrix:

- The default connection for the switch configuration will be set to prohibited.
- A logical definition between a channel path using the switch as a dynamic switch and a control unit will lead to a dynamically allowed connection between the channel path entry port and the port serving as the link address.
- A logical definition between a channel path without a dynamic switch and a control unit, or connections via a chaining switch, will lead to a dedicated connection.

HCD will define a dedicated connection in all the cases where there is only one connection possibility left between the control unit and the channel path entry port. For dedicated connections, HCD considers only those ports, which have not yet been used for dynamic connections. In the case of chained connections, the selected switch configurations of chaining switches are used to determine the possible paths.

- Connections between ports connected to a channel path type, which make a specific port configuration necessary, will be set accordingly. For example, connections between ports serving as entry ports for FCV channels will be defined as explicitly dynamically prohibited.
- In addition to generating the matrix, HCD issues informational messages for all cases in which a path was not completely defined or where different paths conflict with each other. If HCD is not able to determine port connections unambiguously, it will leave those definitions to the user.

### Deleting switch configurations

To delete a switch configuration, perform the following steps:

1. On the Switch List, select the switch and select the *Work with switch configuration* action from the context menu (or action code 4). HCD displays the Switch Configuration List.
2. On the Switch Configuration List select the switch configuration and the Delete action from the context menu (or action code d). The Confirm Delete Switch Configuration panel is displayed.

```
Row 1 of 1
Command ===> _________________________________ Scroll ===> PAGE

Scroll forward to view the complete list of switch configurations to be deleted. Press ENTER to confirm delete request. Press F12 to cancel delete request.

Switch ID ......... : 03
Switch Config. ID Description
SWCON1
**************************************** Bottom of data ****************************************
```

3. Press the Enter key to confirm deletion of the switch configuration, or use the F12=Cancel key to cancel the delete request.
4. The updated Switch Configuration List is displayed.

**Migrating existing switch configurations**

HCD allows to migrate a switch configuration from three sources into the IODF:
- An ISPF table containing a switch configuration as stored by I/O Operations
- An active switch
- A saved switch file

**Prerequisites**

To migrate from an ISPF table, the I/O Operations ISPF table data set name escm.SINGITBL has to be concatenated in the table library chain in your HCD start-up procedure, where “escm” is the high-level qualifier for your I/O Operations installation, or any other ISPF table data set containing ISPF tables saved by I/O Operations.

For additional information, see “Prerequisites” on page 9.

**Migration steps**

1. On the Primary Task Selection panel select Migrate configuration data.
2. On the resulting panel select Migrate switch configuration data.

```
Migrate Switch Configuration Data

Migrate switch configuration definitions to:

Switch ID ......... : 98 +
Switch configuration ID .... A98BC__ +

From one of the following:
  _ 1. ISPF table
  2. Active Director
  3. Saved Director file
```
Specify the switch ID and the switch configuration ID of the empty switch configuration in the IODF to which you want to migrate the data.

If you have not previously defined the switch and the switch configuration, a panel appears that lets you define them (see "Defining switches" on page 172). Select the source from which you want to migrate the switch configuration.

3. A panel appears on which you can define the source:

   - From ISPF tables:
     - Specify the name of the ISPF table that contains the configuration.
   - From an active director:
     - Specify the device number of the switch.
   - From a saved director file:
     - Specify the name of the switch file.
     - Specify the device number of the switch from which the switch file is to be taken.

The following panel is displayed when you are migrating from a saved switch file.

```
+-------------------+   Migrate from Saved Director File   +-------------------+
|                   | Specify the following values.          |
| Director file name | ______ |
| Director device number | ____ |
```

**Note:** The source of the switch configuration as specified in the switch device number field does not have to be the switch as specified in the Switch ID field on the Migrate Switch Configuration Data panel. It is possible to take a switch configuration from any switch and save it with HCD for another switch. However, the description of each port connection is saved with the switch configuration, and has to be updated to reflect the real channel path and control unit connections of the target switch.

**Changing port names**

In HCD, port names are saved with the ports, because the port names reflect the connections of a port. When migrating switch configurations with different port names to HCD, the IODF reflects the port names defined in the switch configuration that were migrated last.

**Changing hardware status of a port**

If the migration source contains ports set to installed and the existing IODF contains the same ports set to uninstalled, the hardware status after the migration depends on the kind of source:

<table>
<thead>
<tr>
<th>Migration Source</th>
<th>Hardware Status of Ports after Migration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active switch</td>
<td>Changed to installed</td>
</tr>
<tr>
<td>Saved director file</td>
<td>Uninstalled, migrated values ignored</td>
</tr>
<tr>
<td>ISPF table</td>
<td>Uninstalled, migrated values ignored</td>
</tr>
</tbody>
</table>
Activating switch configuration data

You can activate a switch configuration, which has been defined or changed with HCD, for a switch. Thus, change and maintenance in a configuration controlled by I/O Operations is simplified by the possibility to use the same user interface.

Prerequisites

- The IODF has to be a production IODF.
- A switch control unit must be defined for each switch.
- A switch device must be defined for each switch control unit.
- The switch control unit must have at least one channel path connected using the switch.
- For activation and saving, an ESCON Manager lock of another user must not exist.

For additional information, see “Prerequisites” on page 9.

Activation steps

1. On the Primary Task Selection panel, select *Activate or process configuration data* and from the resulting panel, select *Activate switch configuration*. The Activate Switch Configuration panel is displayed.

   ┌───────────── Activate Switch Configuration ───────────────────────────────┐
   │ │
   │ Specify the following values, and select how to handle an existing ESCON │
   │ Manager lock. │
   │
   │ IODF name ........: 'BPAN.IODF01' │
   │
   │ Switch ID ......... ____+ │
   │ Switch configuration ID . . BASIC + │
   │
   │ ESCON Manager lock . . 1 1 . Break another user's ESCON Manager lock │
   │ 2. Preserve another user's ESCON Manager lock │
   └───────────────────────────────────────────────────────────────────────────┘

2. You may choose between two different kinds of switch activation:

   **Single switch activation**
   For single switch activation, use the Activate Switch Configuration panel to:
   - Specify the switch ID and the switch configuration ID that is to be written to the switch.
   - Select how to handle an existing ESCON Manager lock if it is in use by another user.

   I/O Operations uses a locking mechanism to serialize connectivity changes across multiple users and systems. This lock allows only one user (or program) to control I/O Operations command processing at a time. If more than one user at your installation is given the ability to enter I/O Operations commands, they must synchronize their usage of I/O Operations to avoid delays or contention for I/O Operations resources.

   **Multiple switch activation**
   For multiple switch activation, use the Activate Switch Configuration panel to:
   - Specify only the switch configuration ID and no Switch ID. A panel is displayed showing all switches that have a configuration under the specified name.
Switches that are not connected to any switch device are marked with a disabled marker # in the action column, and cannot be selected for activation. The panel shows 'YES' in the 'More?' column if there are more than one control unit and one device. If there is only one control unit and one device, the 'More?' column is left blank. HCD uses the switch devices that are connected to the active system.

- Select the switches that will be activated simultaneously.

When you have finished your selection, press the Enter key. A confirm panel is displayed. On this panel you have to confirm if you want the active switch configuration of all switches shown in the list to be updated.

**Note:** In case a switch activation fails, no switch will be activated, the rule is "none or all".

For activating switch configuration data, refer to [IBM Tivoli System Automation for z/OS Planning and Installation](https://www.ibm.com).
Specify the following values.

Save switch configuration:
Switch ID .............. 65 +
Switch configuration ID ... FICON65 +

In Director file:
File name .............. CONF65
File description .......... as of 11/2000

Select how to handle any existing:
ESCON Manager lock ... 2
1. Break another user's ESCON Manager lock
2. Preserve another user's ESCON Manager lock

Director file ....... 2
1. Overwrite
2. Do not overwrite

2. Use the Save Switch Configuration Data panel to:
   • Specify the switch ID and the switch configuration ID for the configuration data that is to be saved in the switch file
   • Specify the name of the switch file used to store the switch configuration data
   • Select how to handle an existing ESCON Manager lock
   • Indicate whether to overwrite an existing switch file with the file name you specified.
Chapter 8. How to activate or process configuration data

Overview

This information unit describes how to:
- Build a production IODF
- Build an IOCDS
- Build S/390 microprocessor IOCDSs
- Manage S/390 microprocessor IPL attributes
- Build an IOCP input data set
- Create JES3 Initialization Stream Checker data
- Build an OS configuration data set
- Verify an I/O configuration
- Activate a configuration dynamically
- Activate a configuration sysplex wide
- Build a CONFIGxx member
- Process the Display M=CONFIG(xx) command
- Switch IOCDS for next POR
- Specify an IODF for IPL

Before the channel subsystem and the operating system can use the configuration that you have defined with HCD, you must build a production IODF from the work IODF.

With the production IODF, you can perform the following tasks in preparation for IPL or dynamic activation.
- Build an input/output configuration data set (IOCDS) from the production IODF for processors not configured in an S/390 microprocessor cluster. The configuration can then be used by the channel subsystem.
- Build IOCDSs of central processor complexes (CPCs) configured in an S/390 microprocessor cluster.
- Manage IPL attributes of central processor complexes (CPCs) configured in an S/390 microprocessor cluster.
- Build an input data set for the input/output configuration program (IOCP) from the production IODF.
- Create data for input to the JES3 Initialization Stream Checker. This checker program ensures that the data used by MVS is consistent with the data used by JES3. (This task can also be done with a work IODF.)
- Build an OS configuration data set from the production IODF. For VM this is an HCPRIO input data set.
- Verify the configuration described in an IODF against a system.
- Activate the configuration dynamically using the activate function (locally or sysplex wide).
- Build a CONFIGxx member for a system from the I/O definitions in an IODF.
- Compare the information in the CONFIGxx member of a system of the sysplex with the existing configuration on that system.
- Switch the IOCDS for the next POR.
Finally, you can use HCD to:

- Specify which IODF is to be used for IPL.

### Build a production IODF

Although HCD validates configuration data as it is entered, a complete validation may not be performed, because data may not be defined at this time. Therefore, a "post-validation" is performed at "Build Production IODF" time. This validation might issue messages you have to deal with, according to their severity. The production IODF is not created if any errors with a severity higher than 'warning' are produced.

During the validation HCD invokes the IOCP program to perform checking of the channel packaging rules. Therefore, note that the correct version of the IOCP program must be accessible.

Depending on what is defined in the configuration, the work IODF must contain a definition for at least one operating system, or one processor, or one switch.

- For an MVS operating system, the IODF must contain at least one EDT and one device.
- For a VM operating system, the IODF must contain at least one device as console.
- For a processor, the IODF must contain a definition for at least one channel path, one control unit, and one device. If only receiving CF channel paths are defined for a processor, the control unit and device definitions can be omitted.

**Note**

A production IODF must have a single extent. If the production IODF has multiple extents, the IPL process results in a WAIT state (wait state code '0B1', reason code '002'). HCD issues error message CBDA009I if a production IODF cannot be built in a single extent.

You can use production IODFs with multiple extents for dynamic activation only. In this case, HCD warns you with message CBDA009I that an IPL with this IODF is not possible, but dynamic activation continues.

To build a production IODF, perform the following steps:

1. On the HCD entry panel, select *Activate or process configuration data*. 
2. From the resulting panel, select **Build production I/O definition file**. HCD validates the configuration data in the work IODF. If the work IODF is valid, then a production IODF can successfully be built.

For work IODFs containing XMP processor definitions, before you can build a production IODF, the correct PCHIDs must be defined in the work IODF. You can use the CHPID Mapping Tool to achieve the task to either insert missing PCHIDs or to update PCHIDs in a work IODF. However, inserting or updating PCHIDs into an IODF using the CHPID Mapping Tool is only possible with a so-called **validated work IODF** that you can get in one of two ways:

a. Use the task **Build validated work I/O definition file**. This task validates a work IODF for correctness and completion, and may issue messages that describe incomplete or erroneous logical definitions. Missing PCHID values are not flagged as errors. If errors occur, correct them an restart this task. As soon as no more errors occur, the output from this task is a validated work IODF.

b. If you tried to build a production IODF without being aware of one or more missing PCHIDs for XMP processors, but the work IODF satisfies all other validation rules, then the output from **Build production I/O definition file**, too, is a validated work IODF. A message will show all CHPIIDs for which the required PCHIDs are missing.

With a validated work IODF, you can use the CHPID Mapping Tool to accomplish the task to update or insert required PCHIDs. Input to this tool is an IOCP input data set. To get this input, now use the task **Build IOCP input data set** from the panel shown in [Figure 80](#). This leads you to the **Build IOCP Input Data Set** panel shown in [Figure 88](#) on page 208. Because the input to the CHPID Mapping Tool must be a stand-alone IOCP, in this panel, specify the appropriate option as shown:

- **Input to Stand-alone IOCP? Yes** (Yes or No)

How to proceed using the CHPID Mapping Tool to get PCHIDs inserted or updated in the validated work IODF, see “How to interact with the CHPID Mapping Tool” on page 213. As soon as all PCHIDs are correct in the validated work IODF, the production IODF can be built.

3. If you initially requested activity logging, a panel like the one shown in “Activity logging and change logging” on page 54 is displayed. Enter the
activity logging details your installation requires. The Build Production I/O Definition File panel is displayed.

---

**Figure 81. Build Production I/O Definition File**

The selection of option Continue using as current IODF controls which IODF is in access after the production IODF has been built. In addition, if you select option 1, The work IODF in use at present, the content of the currently built production IODF is copied to the work IODF. This ensures that the work IODF contains the latest configuration tokens of the IODF, and you can continue to use the work IODF for further updates. If you select option 2, The new production IODF specified above, the content of the production IODF is not mapped into the work IODF. In that case, you should start from the newly built production IODF when performing further changes to the I/O configuration.

4. Specify the name and volume serial number (if applicable) for the production IODF. "IODF naming convention" on page 35 describes the syntax of a production IODF name. If you choose a name without complying to the prescribed syntax of a production IODF name, that IODF can not be used for the IPL and dynamic activate. Moreover, to perform a dynamic activate, the high-level qualifier of the production IODF has to be the same as the one of the IODF used for the previous IPL or dynamic activate.

If the data set name for the production IODF does not adhere to the naming convention for a production IODF, the Confirm Production IODF Name panel is displayed, and you must confirm the IODF name.
If you use the same name for the new IODF as for an existing IODF, you can replace the existing IODF. In that case, the Confirm Delete I/O Definition panel is shown. Select yes, to confirm deletion of the IODF. Be careful, not to delete the active IODF. If you have specified the name of the active IODF, another confirmation panel is shown that warns you once more about the effect of the chosen name.

5. After pressing Enter, the Define Descriptor Fields panel appears.

---

**Figure 82. Confirm Production IODF Name**

The data set name you specified does not follow the naming convention for production IODFs. You will not be able to IPL or ACTIVATE a system configuration from an IODF with this data set name.

Confirm the specified IODF name... 1. Yes 2. No

F1=Help F2=Split F3=Exit F9=Swap F12=Cancel

---

**Figure 83. Confirm Delete I/O Definition File**

To confirm delete request, select Yes. To cancel delete request, select No.

Delete request confirmed... 1. Yes 2. No

Creation date...: 1997-06-10
Last update...: 1997-09-10 14:19
Volume......: SMS012
Description......: IODF for raised floor 710
For systems D0, D2, D4 and D6
  incl. sensed data and mig. SW data

---

**Figure 84. Define Descriptor Fields**

Specify or revise the following values.

Production IODF name...: 'DOCU.IODF01'
Descriptor field 1...: DOCU
Descriptor field 2...: IODF01

F1=Help F2=Split F3=Exit F5=Reset F9=Swap F12=Cancel
Specify the descriptor field 1, 2, or leave the default values. The descriptor fields describe the IODF and will be part of the HSA token. **Attention:** If you specify asterisks (**), equals (==), pluses (++, or minuses (--)) for the IODF suffix in LOADxx, never change the default descriptor field values, because z/OS uses these values to find the current IODF during IPL. Take this relationship also into consideration, if you copy the IODF to a different data set name. For further details refer to [z/OS HCD Planning](#). After the production IODF is built, HCD displays a message.

6. If the work IODF has an activity log file defined for the work IODF, it is copied. After the production IODF has been built, HCD informs you that the production IODF has been created.

You can also create a production IODF using the HCD batch facility (for details see “Build a Production IODF” on page 314).

---

**Note:**

If the work IODF has an associated MCF, the MCF data set is copied and associated to the production IODF.
Build an IOCDS

When a production IODF has been created, you can build an IOCDS (it can be built only from a production IODF). Processors may have varying numbers of IOCDSs. A particular IOCDS is used at POR time to make the configuration data known to the CSS.

The following procedure is only recommended for processors that do not have an SNA address defined, including processors configured in an S/390 microprocessor cluster. For processors in an S/390 microprocessor cluster with an SNA address defined, use the procedure described under “Build S/390 microprocessor IOCDSs” on page 203.

While building IOCDSs HCD internally calls the IOCP program. Therefore, note that HCD must be installed in an APF-authorized library.

1. On the Primary Task Selection panel, specify the name of a production IODF and select Activate or process configuration data.
2. From the resulting panel select Build IOCDS. HCD displays the Processor List panel.
3. On the Processor List panel, select the processor and press the Enter key. HCD displays the IOCDS List panel.

The IOCDS list shows those IOCDSs that are built using the currently accessed production IODF.

4. Whenever the IOCDS list is invoked, HCD tries to get actual IOCDS data (e.g. date and time of last update) for processors with SNA addresses directly from the support element (SE) and displays it.

IODF data is shown only if the SE does not provide information or where an SNA address is not defined.

A production IODF is updated with data retrieved from the SE if discrepancies between that data and the stored IODF data are detected.

5. On the IOCDS List panel, select the IOCDSs that you want to update and select Update IOCDS from the context menu (or action code 9). HCD displays the Build IOCDS panel.
On this panel, you can:

- Into the Title1 field enter identification information you want to be written on the first header line of the IOCP input data set. The first eight characters are used as IOCDS name. This input is used as the MSG1 parameter value of the IOCP ID statement. The batch job passes the MSG1 parameter to the IOCP input data set via the HCDCNTL DD statement (see Figure 130 on page 318 and Figure 131 on page 319).

- Specify the Dualwrite option that describes whether the IOCDS is to be updated on both sides of a physically partitioned processor.

- Specify whether you want to perform a remote or local write of an IOCDS for a processor that has a SNA address defined. The Remote Write option is initialized with 'Yes' if a SNA address is defined to the selected processor. In such a case, HCD initiates a remote IOCDS build and write to the support element with the designated SNA address. If the option is changed to 'No', a local IOCDS build is performed.

- Specify the Switch IOCDS for next POR option, that means whether you want to make this IOCDS the active one for the next power-on reset (POR).

- Specify whether to Write IOCDS in preparation of upgrade. This specifies whether an IOCDS is to be written regardless of processor type. This is useful to prepare for a processor upgrade.

If Yes is specified, an IOCDS for the selected processor is written regardless of the processor type. For a list of processors that support writing an IOCDS, in preparation for a processor upgrade, or for which such an IOCDS can be written, see "Supported Hardware Report" on page 401.

**Note:** If, as a result of a processor upgrade, an IOCDS download is not possible you can, after having built the production IODF, create an IOCP input data set that can be used with the stand-alone IOCP to generate an IOCDS for use with POR.

- Change the job statement information to meet the installation needs. Note that a batch job to build an IOCDS must run on the processor on which the
IOCDS is to be updated. (See “Job statement information used in panels” on page 75 for a description of the job control information that you need to specify when you build an IOCDS.)

In a multiprocessor JES environment, be sure to specify the JES command and/or job class to ensure that the job runs on the correct processor(s).

The recommended region size is 2 MBytes more than the IOCP needs. For the region size required by IOCP, refer to the IOCP User’s Guide for your processor.

6. When an IOCDS is built, a record is written for the processor configuration. If you build a new IODF from an existing IODF, the records are copied to the new IODF. When you build IOCDSs from this new IODF, the IOCDSs from the old IODF are also shown on the IOCDS list. Because the batch job requires exclusive use of the production IODF for processors that have no SNA address specified, you have to either leave the HCD session or change the currently accessed IODF to run the submitted job.

You can also invoke the Build IOCDS task in batch mode, see “Build an IOCDS or an IOCP input data set” on page 316.

Notes:
1. Unlike writing an IOCDS using the IOCP program, the HCD process generates no IOCP report when using the Build IOCDS function.
2. When a processor has been upgraded in the IODF, the old IOCDS status data in the IODF is deleted.

Using this procedure for processors with an SNA address:
If you use this procedure for processors in an S/390 microprocessor cluster with an SNA address defined, HCD does the following:

- HCD writes the IOCDS for that processor to the support element with the designated SNA address (remote IOCDS build).
- If you run the job under MVS/ESA SP Version 5, OS/390, or z/OS, HCD writes the IOCDS for that processor to the support element with the designated SNA address.

To run the batch job, you do not have to leave the HCD session. If the processor has defined an SNA address, HCD assumes that it is part of an S/390 microprocessor cluster. In this case, the job can immediately start without the need to free the currently accessed IODF.

HCD tries to update the IOCDS record but is not able to because the IODF is still allocated by the HCD dialog. This results in an error message on the console log. To avoid this error message, HCD offers the profile option of bypassing the IODF information update (see “Bypass IODF information update for SNA processor” on page 30).

Build S/390 microprocessor IOCDSs
The following procedure describes how to build an IOCDS for processors in an S/390 microprocessor cluster with an SNA address defined.

To build IOCDSs within an S/390 microprocessor cluster:
- The SNA address has to be defined for a CPC configured in an S/390 microprocessor cluster.
• Specific RACF authority has to be attained (for details on required access authority, refer to “Security-related considerations” on page 339).
• The operating system must not be running as a guest under VM.

Perform the following steps:
1. On the Primary Task Selection panel, select **Activate or process configuration data** and from the resulting panel select **Build and manage S/390 microprocessor IOCDSs and IPL attributes**. The S/390 Microprocessor Cluster List panel is displayed:

   This panel shows all CPCs configured in an S/390 microprocessor cluster. They are identified by the SNA address of their support element and displayed together with their Type and Model as well as the Processor ID in the IODF. The SNA address has been specified in the processor definition task for the IODF processor definitions and enables the relation to the configured CPCs.

   A disabled sign (#) in the action entry field can be due to:
   • SNA address not defined in the IODF. In this case, the IODF Processor ID shows no value. Either define the SNA address for a corresponding processor in the accessed IODF or use another IODF.
   • SE (support element) of CPC did not respond. In this case, the CPC Type and Model fields show no values.

   A processor ID followed by two dots (..) indicates that this SNA address has been defined for several processors in the IODF. The first processor ID (in alphabetical order) with the SNA address is displayed. If you want to apply any of the group actions on another processor, use **Select other processor configuration** from the context menu (or action code ）。

2. On the S/390 Microprocessor Cluster List panel, select the CPCs for which you want to build and manage the IOCDSs and **Work with IOCDSs** from the context menu (or action code ）。HCD displays the IOCDS List panel (shown with sample data):
On the IOCDS List panel all applicable IOCDSs of the selected CPCs are displayed and arranged in ascending order by IOCDS names (starting, for example, with A0-IOCDSs, A1-IOCDSs). This list enables you to apply the IOCDS functions as group actions against one or several IOCDSs for all selected processors.

The data displayed is retrieved directly from the support elements. If, however, the support element does not answer, HCD displays the data saved in the IODF and issues a message accordingly.

A production IODF will be updated with data retrieved from the support element if discrepancies between that data and the stored IODF data are detected. A work IODF will remain unchanged.

The Type field contains one of the following types of power-on reset modes to be used with the I/O configuration defined in the IOCDS: S/370, ESA/390, or LPAR.

The Status field indicates the status of the IOCDS:
- **Alternate**: not to be used at the next POR
- **POR**: to be used at the next POR
- **Invalid**: IOCDS is opened for update

The Token Match-IOCDS/HSA field indicates whether the IOCDS token matches the current HSA token. If Yes is shown, it means that the IOCDS has been built by HCD, and that it matches the current I/O configuration - either because this IOCDS was used for the last POR, or the matching configuration has been activated dynamically.

The Token Match-IOCDS/Proc. field indicates whether the IOCDS token matches the processor token in the IODF, currently used in the HCD dialog. If Yes is shown, the IOCDS has been built from the IODF currently used in the HCD dialog.

3. On the IOCDS List panel, you can select the following actions from the context menu:
   - Use the Update IOCDS action (or action code [U]) to build or update the selected IOCDSs with the I/O configuration data from the currently accessed production IODF. See step [4 on page 206](#) on how to proceed.
   - Use the Switch IOCDS action (or action code [S]) to mark an IOCDS as the IOCDS that is used for the next POR. The Status field will be set accordingly.

You can only switch to an IOCDS that has an IOCDS/HSA token match or to an IOCDS of a processor that is not activated ('POR-required' status).
• Use the *Enable write protection* or *Disable write protection* action (or action codes e and X) to allow or prohibit updating the selected IOCDSs of the designated CPCs. The Write Protect field will be set accordingly.

Use the F20=Right key to move the work area to the right to see information such as date and time of the last IOCDS update and the IOCDS configuration token.

4. If you select the **Update IOCDS** action, HCD displays the Build IOCDSs panel.

   ![Build IOCDS Panel]

   On this panel you can:
   - Enter identification information you want to be written on the first header line of the IOCP input data set in the Title1 field.
   - Specify the Switch IOCDS option, if you want to make this IOCDS the active one for the next power-on reset (POR).
   - Specify whether to Write IOCDS in preparation of upgrade. This specifies whether an IOCDS is to be written regardless of processor type. This is useful to prepare for a processor upgrade.

   If Yes is specified, an IOCDS for the selected processor is written regardless of the processor type. For a list of processors that support writing an IOCDS, in preparation for a processor upgrade, or for which such an IOCDS can be written, see "Supported Hardware Report" on page 401.

   **Note:** If, as a result of a processor upgrade, an IOCDS download is not possible you can, after having built the production IODF, create an IOCP input data set that can be used with the stand-alone IOCP to generate an IOCDS for use with POR.

5. After pressing the Enter key on the Build IOCDSs panel, the Job Statement Information panel is displayed. Specify the information for the batch job that HCD generates to build the IOCDSs.

   The recommended region size is 2 MBytes more than the IOCP needs. For the region size required by IOCP, refer to the **IOCP User’s Guide** for your processor.

   ![Job Statement Information Panel]
Manage S/390 microprocessor IPL attributes

For IPL operations for CPCs configured in an S/390 microprocessor cluster, you can:
- Display the IPLADDR and IPLPARM attribute values of the last and for the next IPL.
- Modify IPLADDR and IPLPARM attribute values to be used for next IPL.

Perform the following steps:
1. On the Primary Task Selection panel, select *Activate or process configuration data* and from the resulting panel select *Build and manage S/390 microprocessor IOCDSs and IPL attributes*. The S/390 Microprocessor Cluster List panel is displayed (see Figure 86 on page 204).
2. On the S/390 Microprocessor Cluster List panel select the CPCs for which you want to view and modify IPL attributes.
3. Select the *Work with IPL attributes* action from the context menu (or action code 1). HCD displays the IPL Attribute List panel.

The IPL Attribute List panel displays the IPLADDR and IPLPARM attribute values for all selected processor definitions and their partitions (if defined in LPAR mode) that are obtained from the support element of the associated CPCs.

4. On the IPL Attribute List panel view or modify by typing over the attribute values for IPLADDR and IPLPARM.

Use the F20=Right key to move the work area to the right to view the IPL attributes used for the last IPL.

The Next IPLADDR column shows the LOAD address and specifies the number of the IPL device used for next IPL.

The Next IPLPARM column shows the LOAD parameter used for MVS and is a concatenation of the following attributes: IODF Device, LOADxx Suffix, Prompt/Message Option, and Nucleus Suffix. The Next IPLADDR and/or the Next IPLPARM value for the next IPL are taken, when ‘Use dynamically changed IPL address’ and/or ‘Use dynamically changed IPL parameter’ are selected on a LOAD profile that is used to initiate an IPL, or to activate an operating system. This enables you to change the values of IPLADDR and IPLPARM without updating the profile.
Build an IOCP input data set

Sometimes it is necessary to build an IOCP input data set:
- If you have to use the stand-alone IOCP program, which does not support direct access to the production IODF.
- If you want to create a backup on tape in case you need to recover the contents of the IOCDs in the service processor.
- If you need an IOCP input data set as input to the CHPID Mapping Tool in order map CHPIDs to PCHIDs for XMP processors.
- If you need an IOCP input data set for a processor on which there is no HCD running (e.g., a new processor). Also, if you upgrade a processor to a model that results in a new IOCP SYSTEM value and the processor does not support an IOCDS write in preparation for a processor upgrade, you have to run the stand-alone IOCP program.

How to build an IOCP data set

1. On the Primary Task Selection panel, specify the name of a production IODF and select *Activate or process configuration data*.
2. On the resulting panel select *Build IOCP input data set*. HCD displays the Available Processors panel.
3. On the Available Processors panel, select the processor for which you want to build the IOCP input data set. HCD displays the Build IOCP Input Data Set panel.

4. On this panel you can:
   - Enter the identification information you want to be written on the first header line of the IOCP input data set in the Title1 field.
   - Specify the name of the IOCP input data set. The IOCP input data set will automatically be allocated (record length 80, record format fixed block). If the data set already exists, you will be asked to confirm replacing it with the new one.
   - Specify whether to build the IOCP data set for stand-alone IOCP.
     - **Yes**: This is the default. The generated IOCP statements can be used as input to the stand-alone IOCP program or to the CHPID Mapping Tool.

Figure 88. Build IOCP Input Data Set
Note: You may not be able to use such a generated IOCP input data set for the migration function of HCD because, for example, the unit name of control units and device types can be truncated due to IOCP restrictions.

No

The IOCP input data set is built using the IOCP changes described in “IOCP enhancements” and generating the extended migration parameters and statement (if the profile statement MIGRATE_EXTENDED is set to YES) as described in “IOCP input data sets using extended migration” on page 210. Note that if you try to process these IOCP statements with the stand-alone IOCP program, you may run into problems, because the program may not accept the generated syntax.

- Change the job statement information to meet the installation needs. With JCL overwrite statements you can modify the EXEC procedure that is invoked. You can, for example, specify the HCD profile using the job step name GO. (See “Job statement information used in panels” on page 75 for a description of the job control information that you need to specify when you build an IOCP input data set.) Ensure that the batch job runs in a region with at least 4 MBytes.

TOK=value

Configuration programs use this keyword to forward information to the CPC which is required to enable the dynamic I/O configuration capability of any resulting IOCDS. This keyword is not intended for direct user input. The contents needs not relate to the target processor. It just must match the token in the IOCDS/HSA and the currently active IODF.

You can also invoke this task in batch mode. See “Build an IOCDS or an IOCP input data set” on page 316.

Note:

You should never change an IOCP input file generated by HCD and use it to write an IOCDS. If changes are necessary, use HCD to regenerate the IOCP input.

IOCP enhancements

The generated IOCP data set contains control unit and device types of 8 characters and a device model of up to 4 characters. Such an IOCP input data set can be processed by IOCP (with APAR OW13343) and remigrated to HCD without the need to correct the control unit and device types that exceed the 5 character UNIT and 2 character MODEL value limitation.

It also now contains an all-character readable token which allows the user to preserve the dynamic capability when performing a stand-alone IOCP run on a S/390 microprocessor cluster CPC using IOCP input from diskette.
Important Note:

It may not be possible to remigrate an IOCP input data set generated by HCD back into the IODF. The reasons are:

- HCD uses the High Level Assembler program for parsing the IOCP statements. The High Level Assembler earlier than V1.5 is restricted to 255 characters for any keyword value. IOCP statements, however, may contain keywords with a value greater than 255 characters. High Level Assembler V1.5 removes this restriction.

- HCD keeps additional data for a processor configuration that is not contained in an IOCP input data set. This data may be used for validation and, therefore, missing at the migrate step leading to validation errors. For example, the partition usage is defaulted to CF/OS. For a shared CF peer channel, this may lead to a validation error, because only a CF partition may be specified in the access or candidate list.

- Since the IOCP data are only a subset of the processor configuration data, you may lose this additional configuration data if you update a processor configuration from an IOCP input data set.

For updating the IODF via I/O configuration statements, it is recommended to use the extended I/O configuration statements of HCD instead of an IOCP input data set (see "IOCP input data sets using extended migration").

IOCP input data sets using extended migration

As described in "Updating parts of a configuration by migrating input data sets" on page 292, HCD introduces an extended migration to allow you to define your complete configuration without using the ISPF front end dialog.

The extended migration allows you to, for example, define a switch with its ports or define serial numbers and descriptions for devices and control units by editing your input data sets and migrating them into HCD.

Analogously, when building an IOCP input data set from an IODF, information is generated that describes the additional parameters of the configuration objects (if the prerequisites under "Prerequisites to exploit the extended migration" on page 211 are met). Within the generated IOCP input data set, the additional parameters and control statements are shown as comments with special HCD tags so that they can be processed by the IOCP program. When re-migrating such an IOCP input data set to HCD, the tagged comments are identified by HCD and migrated correspondingly.

If you want to use the input data set for both, IOCP processing and HCD migration, the new records must apply to the following rules, so that they can be processed by both programs:

- The new parameters start with the string *$HCDC$ in column 1.
- The new SWITCH statement starts with the string *$HCD$ in column 1.
- The IOCP statement does not have any comment.
- The additional HCD tagged records follow immediately the last record of the corresponding IOCP statement.
- The first keyword starts at column 16.
• The last operand is not followed by a comma.
• There is no comment to the right of the operand.

Prerequisites to exploit the extended migration
To generate the additional keywords during IOCP data set build, note the following prerequisites:
• Specify the following entry in the HCD profile:
  MIGRATE_EXTENDED = YES

When you specify MIGRATE_EXTENDED = NO (which is default), the additional keywords are not generated during IOCP build. In addition, when remigrating the IOCP input data sets, the migration function ignores the commented "$HCDC$" and "$HCD$" tags.

The HCD profile is explained in "Defining an HCD profile" on page 23.
• When building IOCP input data sets, you have to set the option Input to Stand-alone IOCP to No on the Build IOCP Input Data Set panel. See "Build an IOCP input data set" on page 208 for a description of the new option.

Example of an IOCP input data set
Figure 89 on page 212 shows you an example of a generated IOCP input data set with the new parameters. Note that each new parameter starts with "$HCDC$" in column 1. The new switch control statement starts with "$HCD$" in column 1.
Using the IOCP data set as input for the CHPID Mapping Tool

HCD will allow generating an IOCP deck that does not contain any or all necessary PCHID values. You can use this IOCP deck as input to the CHPID Mapping Tool in order to have PCHIDs inserted or updated. The CHPID Mapping Tool will create an IOCP deck with the missing PCHID values, which can then be used as input to the CHPID Mapping Tool again to update or complete the PCHIDs.

Figure 89. Example of an input data set for migration enhancements

```
ID    MSG1='IOCDSNAM', MSG2='BOKA.IODF03 - 95-07-21 16:00', *
      TOK=('TW0', '000000001900967216057040095202F00000000, 0000*000, 95-07-21', '16:00:57', 'BOKA', 'IODF03')
*$HCDC$ DESC='Cluster(099) test floor'
*$HCDC$ SERIAL='1044009672'
*$HCDC$ SNAADDR=(USIBMSC, TW0)
RESOURCES PARTITION=((CF001, 3), (MVSSMAL, 2), (PRIME, 1))
*$HCDC$ DESC=('Coupling facility', 'MVS 5.2.0 Systeml', 'Product on CF image')
CHPID PATH=(10), PARTITION=((CF001), (CF001)), TYPE=CFR
*$HCDC$ DESC='Receiver'
CHPID PATH=(13), PARTITION=((PRIME), (PRIME)), TYPE=CFS
*$HCDC$ TPATH=(TW0, 10, FFFE, FFFE)
*$HCDC$ DESC='Sender'
CHPID PATH=(20), PARTITION=((MVSSMAL), (MVSSMAL)), SWITCH=AB, *
      TYPE=CNC
*$HCDC$ SWPORT=((AA, C0))
*$HCDC$ DESC='Channel for DASD'
*$HCDC$ CHPID PATH=(21), PARTITION=((MVSSMAL), (MVSSMAL)), TYPE=CNC
*$HCDC$ DESC='Channel for DASD'
*$HCDC$ CHPID PATH=(25), PARTITION=((PRIME), (PRIME)), TYPE=CNC
*$HCDC$ SWPORT=((AB, B0))
*$HCDC$ DESC='Switch connection'
*$HCDC$ CHPID PATH=(26), PARTITION=((MVSSMAL), (MVSSMAL)), TYPE=CNC
*$HCDC$ SWPORT=((AA, C5))
*$HCDC$ DESC='Switch connection'
CNTLUNIT CUNUMBR=0005, PATH=(25), UNITADD=((00, 001)), UNIT=9032-3
*$HCDC$ SWPORT=((AB, FE))
*$HCDC$ SERIAL='1021-CUS11'
*$HCDC$ DESC='SWITCH AB'
CNTLUNIT CUNUMBR=0006, PATH=(26), UNITADD=((00, 001)), UNIT=9033
*$HCDC$ SWPORT=((AA, FE))
*$HCDC$ SERIAL='1021-CUS10'
*$HCDC$ DESC='SWITCH AA'
CNTLUNIT CUNUMBR=000F, PATH=(20, 21), UNITADD=((08, 008)), *
      LINK=(A1, A2), CUADD=3, UNIT=3995-151
*$HCDC$ SWPORT=((AA, C3), (AB, B2))
*$HCDC$ SERIAL='5512003330'
*$HCDC$ DESC='DASD on 04-83'
CNTLUNIT CUNUMBR=FFE, PATH=(13), UNIT=CFS
IODEVICE ADDRESS=(080, 004), MODEL=151, UNITADD=08, *
      CUNUMBR=(000F), STADET=Y, UNIT=3995
IODEVICE ADDRESS=110, MODEL=3, UNITADD=00, CUNUMBR=(0005), *
      STADET=Y, UNIT=9032
*$HCDC$ SERIAL='1021-CUS11'
*$HCDC$ DESC='SWITCH AB'
IODEVICE ADDRESS=120, UNITADD=00, CUNUMBR=(0006), STADET=Y, *
      UNIT=9033
*$HCDC$ SERIAL='1021-CUS10'
*$HCDC$ DESC='SWITCH AA'
IODEVICE ADDRESS=(FFE, 002), CUNUMBR=(FFE), UNIT=CFS
*$HCDC$ SWITCH SWID=AA, SERIAL='1021-CUS10', DESC='SWITCH AA', *
      PORT=((C0, CE), (FE, FE)), SWPORT=((CA, AB, C6)), UNIT=9033
*$HCDC$ SWITCH SWID=AB, SERIAL='1021-CUS11', DESC='SWITCH AB', *
      PORT=((B0, F0), (FE, FE)), SWPORT=((C6, AA, CA)), MODEL=3, *
      UNIT=9032
```

Figure 89. Example of an input data set for migration enhancements
Tool then generates a new IOCP input deck containing the assigned PCHID values. You can then migrate the updated PCHIDs into a validated work IODF (see also "How to interact with the CHPID Mapping Tool").

**How to interact with the CHPID Mapping Tool**

Correct PCHIDs are required in the configuration for XMP processors before you can successfully build a production IODF. The task of adding or updating required PCHID information for a work IODF for XMP processors is eased by an interaction between HCD and the CHPID Mapping Tool (CMT). Prerequisite for this task is a so-called validated work IODF that you can get in one of two ways described in "Build a production IODF" on page 196.

You can download the CHPID Mapping Tool from the Internet. It runs on a workstation.

**Process overview**

Input to the CMT is the hardware configuration file (CFReport ¹) of your machine and a valid IOCP input file (with missing or obsolete PCHIDs).

Output from the CMT is again an IOCP input file that now has all missing or updated PCHID values filled in. Upload this IOCP input file and re-import it into the validated work IODF using the HCD primary task Migrate configuration data.

Via this migration task, it is possible to update a validated work IODF with the PCHID values that have been written by the CMT into the IOCP input file. Other changes on the validated work IODF are not possible without losing the status of a validated work IODF. A PCHID migration is only possible to a validated work IODF. Since PCHID migration changes the IODF, the IODF status is reset to ‘not validated’. All functions that allow read-access to a work IODF are also possible for a validated work IODF. Activate functions are not possible, except for building an IOCP deck that can be used as input to the CMT. Only after all PCHIDs have been inserted into the validated work IODF, you can successfully build a production IODF.

**How to insert or update PCHIDs**

Here are the detailed steps you need to perform to insert or update PCHIDs in a validated work IODF.

1. Create a validated work IODF with one of the two methods described in "Build a production IODF" on page 196. Your validated work IODF may lack at least one PCHID that you need to insert or may contain obsolete PCHIDs that you want to update.

2. Go back to the Activate or Process Configuration Data panel shown in Figure 80 on page 197 and use task Build IOCP input data set to export the I/O configuration from the validated work IODF to an IOCP data set (with PCHIDs still missing or obsolete). The hardware configuration token is passed with the IOCP statements (TOK keyword). This token is used to assure that during the process of assigning PCHID values the contents of the IODF is not changed.

   Download this IOCP data set to the workstation where the CMT is running.

---

¹ When a machine is ordered, the output of the order process is a binary file that represents the physical description of the final machine. One of the components of that file is the type and physical location, including the Physical Channel Identifier (PCHID) value assigned to that location, of all the I/O features in the final machine. This file is called a CFReport.

Chapter 8. How to activate or process configuration data 213
3. Use the CHPID Mapping Tool with the downloaded IOCP data set. For information on how to use the CHPID Mapping Tool, refer to the online help and the related documentation.

The output of a successful CMT run is again an IOCP data set which contains the original I/O definitions together with inserted and/or updated PCHID values. The original hardware configuration token is still contained in the generated statements.

4. Upload the new IOCP data set to the host and use the HCD primary task *Migrate configuration data* to import the PCHIDs from the updated IOCP data set into the validated work IODF. During this task, you select

   *migrate option --- 3. PCHIDs*

from [Figure 120 on page 269](#) in “Migrating input data sets using the HCD dialog” on page 269

When importing these statements into the validated work IODF via the migration process for PCHID migration, HCD verifies that the token passed with the IOCP statements match the token stored in the IODF. If this is the case, and if the logical I/O definition described by the imported IOCP statements does not differ from the IODF data, HCD writes the PCHID values into the IODF. If the token does not match, for example, because the IODF has been updated in the meantime, a PCHID migration will not be performed. In this case you must start the process from the beginning.

**Note:** The IOCP input data set may contain keyword values which exceed the 255 character limitation of the assembler program used by HCD for parsing the IOCP statements. This may be the case for the PARTITION (PART) keywords on the RESOURCE statement and on the CHPID statements for spanned channel paths. The affected I/O configuration statements must be deleted in order to perform a successful PCHID migration. However, note that the High Level Assembler V1.5 removes the 255 character limitation.

5. If a PCHID migration has been successfully done, you can invoke the *Build Production IODF* task (again). HCD now builds a production IODF that contains all the data that is required to write the I/O configuration data set (IOCDS) via the IOCP program to the Support Element (SE) of the machine, ready to be used for the next IML.

**Note:** If for any reasons for an XMP processor you want to define a channel that is not physically installed on the machine yet, you can use the method of over-defining a channel path as described in “Over-defining channel paths on an XMP processor” on page 119. Thus you can avoid to let the CHPID Mapping Tool run into an error, because it cannot find the PCHID.

To support the algorithm of mapping the logical CHPID definitions to physical channels, a CMT user can specify priorities to the control units (CU priorities). It is possible to preserve these values across different invocations of the CMT. For this purpose, the CMT passes the CU priorities as special comments to HCD. HCD takes these comment lines and stores them in the IODF attached to the corresponding processor. When generating an IOCP input file for the CMT, HCD includes these comments into the generated statements. HCD does not make these comments visible in the HCD dialog or in the reports.
Create JES3 initialization stream checker data

Because JES3 does not access the IODF directly, it has to be checked whether JES3 I/O and MVS I/O definitions are the same. It is essential that these definitions are consistent. Each time you run this task, the JES3 initialization stream checker data is stored in a data set, thus allowing JES3 to check the above mentioned definitions and to detect inconsistencies among them.

You can start the task from a work IODF as well as from a production IODF. However, it is recommended to use a production IODF to ensure that the same information is used for IPL.

1. On the Primary Task Selection panel, select *Activate or process configuration data*.
2. On the resulting panel select *Create JES3 initialization stream data*. HCD displays the Create JES3 INISH Stream Checker Data panel.

![Create JES3 INISH Stream Checker Data panel]

3. Specify the required values.
   The output data set will automatically be allocated (record length 80, record format fixed block). Depending on whether you specify the data set name as sequential or partitioned, the data set will be either sequential or partitioned. It is recommended to specify a partitioned data set (PDS), because this is required by the JES3 initialization stream checker.
   If the data set already exists, you will be asked to confirm replacing it with the new one.

You can also invoke this task in batch mode. See "Build I/O configuration data" on page 320 for a description of the job control information that you need to specify when you build JES3 initialization stream checker data.

Build I/O configuration data

You can use HCD to create an I/O configuration data set containing either:
- an OS configuration
- a processor configuration
- a switch configuration
- FCP device data.

1. On the Primary Task Selection panel, select *Activate or process configuration data*.
2. From the resulting panel select *Build I/O configuration data*. HCD displays the *Build I/O Configuration Data* panel.
3. Specify one of the available configuration types:

- **Options 1 - 3**:
  Specify the identifier of the configuration to be used and a name for an output data set to contain the configuration statements according to the specified configuration type. The data set will be automatically allocated (record length 80, record format fixed block).

  Specifying an asterisk ("*") in the **Configuration ID** field will generate all configurations of the specific type to the output I/O configuration data set.

  If the data set already exists, you will be asked to confirm replacing it with the new one.

- **Option 4**:

  HCD exports the FCP device configurations for a specific processor from the currently accessed IODF as comma-separated values into a CSV output format which you can use as input to the WWPN Prediction Tool to assign world-wide port names to virtualized FCP ports. This HCD output file is also referred to as **FCP SAN configuration template file**.

  Specify the desired processor ID in the **Configuration ID** field. The data set will be automatically allocated (record length 132, record format fixed block).

  If the data set already exists, you will be asked to confirm replacing it with the new one.

You can also invoke this task in batch mode. See "Build I/O configuration data" on page 320 for a description of the job control information that you need to specify.

---

**Verify a configuration**

HCD allows you to check the definitions in your IODF against the actual configuration as sensed from the active system. See "Prerequisites" on page 9 for the prerequisites for the verify function.

The verify function results in a list of all sensed paths in comparison to the defined paths. Using a filter, this usually extensive list can be reduced to the data of interest. The list can be saved and/or printed. The verification can also be carried out as a batch job resulting in an I/O Path Report (see the example 399 in section "Print configuration reports" on page 324).
Verifying a configuration against the local system

1. To verify the I/O configuration of the local system select *Activate or process configuration data* from the primary task level. From the resulting panel select *Activate or verify configuration dynamically*. The Activate or Verify Configuration panel is displayed.

   **Note:** For the verify function on the Activate or Verify Configuration panel to be available, the processor configuration from which the active IOCDS was built must match the configuration in the IODF used for IPL (token match).

2. Select the *Verify active configuration against system* task to compare the system against the active IODF. Select the *Verify target configuration against system* task to compare the system against the accessed IODF. The Identify System I/O Configuration panel is displayed.

   ![Identify System I/O Configuration](image)

   **Figure 90. Identify System I/O Configuration**

3. Specify the processor ID and OS configuration ID. If the specified processor is in LPAR mode, you must also specify a partition name.

Verifying a configuration against a system in the sysplex

1. Select *Activate or process configuration data* from the primary task level. From the resulting panel select *Activate configuration sysplex-wide*. The Active Sysplex Member List is displayed, listing all active systems of the Sysplex as stored in the sysplex couple data set of the system.

2. After selecting the system to be verified a context menu with two verification actions is displayed. Select the *Verify active configuration against system* task (action code `/SF580000k/SF590000`) to compare the system against the active IODF. Select the *Verify target configuration against system* task (action code `/SF580000l/SF590000`) to compare the system against the accessed IODF.

   The Identify System I/O Configuration panel is displayed (see Figure 90).

3. Specify the processor ID and OS configuration ID. If the specified processor is in LPAR mode, you must also enter a partition name.

The I/O path list

The I/O Path List is the output of the verify function available on the Active Sysplex Member List and the Activate or Verify Configuration Dynamically panel (see “Verifying a configuration against the local system” and “Verifying a configuration against a system in the sysplex”).
The list compares the configuration in the accessed or the active IODF with the actual configuration as sensed from the system.

For each channel path sensed and/or defined in the accessed IODF the list contains a row showing the I/O path and the sensed and defined channel path, control unit, and device information. If a switch is included in the path, the right page shows the corresponding switch information.

Any discrepancies between the defined and the sensed data are indicated in column D on the right. For channel paths for which column D is blank, the defined and sensed data are consistent. The following values may appear:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Defined and sensed I/O paths differ</td>
</tr>
<tr>
<td>C</td>
<td>Defined only to processor but not to OS</td>
</tr>
<tr>
<td>0</td>
<td>Defined only to OS but without a path to the processor</td>
</tr>
<tr>
<td>@</td>
<td>A combination of * and C</td>
</tr>
</tbody>
</table>

On the display column D is highlighted.

Columns STAT and 0 indicate the status of the I/O path and the connected device, respectively. An empty field means that the corresponding I/O path or device is online. Offline I/O paths are marked with OFFL and offline devices with Y. If the system is unable to sense the status of an I/O path, it is marked UNKN.
Asterisks (*) in the CHT (channel type), CUTYPE (control unit type), or DEVTYPE (device type) columns indicate that I/O paths are returned but the values for the corresponding types are blank or invalid.

For certain configurations the I/O path list, although restricted to one processor or partition, can be extensive. Using the Filter action you can reduce the list to the entries of interest.

You can save the displayed list by entering SAVE in the command line on the I/O Path List panel.

**Note:** The LOCATE command is not available for the I/O path list.

## Activate a configuration dynamically

The system programmer (or other authorized persons) can use the option *Activate or verify configuration dynamically* or the ACTIVATE operator command to make changes to a running configuration. That is, the possibility is offered to change from a currently active configuration to some other configuration that is to be made active without the need to POR or IPL the system again.

When activating a configuration dynamically, HCD compares the currently active IODF with the IODF that is to be activated and then processes the difference.

For the IODF that is to be activated, HCD uses the production IODF that is currently in use with the dialog. Use the same high-level qualifier for the currently active IODF and the IODF to be activated.

[OS HCD Planning](#) gives a detailed description of how to dynamically activate a configuration. It describes the prerequisites for a dynamic activation, explains when hardware and software changes or software-only changes are allowed, and describes the actions necessary to change your I/O configuration dynamically. The following sections describe how to use the HCD dialog for this purpose.

Before activating a configuration dynamically, you may want to view information about the IODF that has been used for IPL or the last dynamic activation.

## View active configuration

HCD allows you to view the name and status of the IODF that has been used for IPL or for last dynamic activation. The operating system configuration and EDT identifier and, if applicable, the configuration token, which is currently active in the HSA (hardware system area), are shown. Use the *View active configuration* function for an overview of the actual status for dynamic activation, indicating whether hardware and software changes are allowed.

1. On the Primary Task Selection panel, select *Activate or process configuration data* and then *View active configuration*.

   The View Active Configuration panel with sample data is shown below:
2. Press the Enter key to display detailed information on limitation(s) to the activation scope. A Message List panel is displayed containing the messages about the reasons for the restrictions.

**How to activate if hardware and software changes are allowed**

The following procedure describes how to activate a configuration dynamically if both hardware and software changes are allowed. Refer to [z/OS HCD Planning](#) on information when both hardware and software configuration changes are allowed and when only software configuration changes are allowed.

1. On the Primary Task Selection panel, select *Activate or process configuration data*, and from the resulting panel select *Activate or verify configuration dynamically*. HCD displays the Activate or Verify Configuration panel.

2. Select what you want to activate. The following figure assumes that you selected task 1. *Activate new hardware and software configuration*. The panels when you select the other tasks are similar.
The panel contains information about the currently active IODF.
If your processor runs in BASIC mode, the option Allow hardware deletes and the option Delete partition access to CHPIDs unconditionally are not displayed.

3. On the Activate New Hardware and Software Configuration panel you can change the fields that relate to the IODF that is to be activated, and you can specify options as applicable to your requirements. It is recommended that you first specify to test an activation before you dynamically activate a configuration.

Allow hardware deletes option:
If logical partitions have been defined for the currently active configuration, you can specify whether you want to allow hardware deletes.
Yes means that the hardware deletes become effective for all partitions.
No (the default) means that, if the changes include requests for deleting hardware, the activation is rejected.

Note that hardware delete can also be indirectly performed as a result of other changes, for example, a change of a channel path consist of a deletion and an addition of a channel.
A configuration change is rejected if it includes a hardware delete for an I/O component that is online to the logical partition from which you are making the change, even if you have entered Yes in the Allow hardware deletes option field. Therefore, you should vary offline any affected I/O component in all logical partitions. For example, when changing a channel path from unshared to shared, you must allow hardware deletes, and you must configure the channel path offline and vary offline the associated I/O devices before you activate the configuration. See z/OS HCD Planning for details about preventing disruption when changing the characteristics of I/O components.

Delete partition access to CHPIDs unconditionally option:
You can also specify how this activation should treat any deletion of a partition from the access or candidate list of a channel path. In the field Delete partition access to CHPIDs unconditionally (FORCE=CANDIDATE), enter either Yes or No. If you specify Yes, the access to the channel path will be revoked even if the channel is configured online to the partition; the channel
will be configured off-line to the partition, and active I/O operations might be
disrupted. If you specify No (the default), the activation will be rejected if it
includes a deletion of partition access to a channel path that is configured
online to that partition.

Note: You cannot unconditionally delete the partition that is invoking the
activate request from the candidate or access list of a channel path if the
channel path is currently configured online.

4. If the dynamic activation completed successfully, HCD displays a message.

Configure channel path online to the partition
When activating a configuration in which a partition is added to the access list of a
channel path, you must configure the channel path online to the partition using
either the MVS CONFIG command or the processor console CHPID command.
PR/SM will configure the channel path to the partition at subsequent activations
only after you configure it using the commands above at least once.

When a particular IOCDS is used in a POR for the first time after it has been
written, the definitions in that IOCDS are used to determine the assignment of
channel paths to logical partitions according to the channel path access lists that
are defined. All previous information about channel configurations associated with
this IOCDS is discarded. The exception to this rule is when a newly written IOCDS
is first used as part of a dynamic I/O change to the system. For example, the new
IOCDS is used as a result of a Switch IOCDS for next POR or the new IOCDS is
the target of the ACTIOCDS= parameter of the MVS ACTIVATE command. When
a new IOCDS is used in this manner, the current state of channel configurations is
preserved and immediately associated with the newly written IOCDS.

See z/OS MVS System Commands for information about the MVS CONFIG
command and see the applicable Processor Resource/Systems Manager Planning Guide
for information about the CHPID command and for a description of automatic
configuration of channel paths to partitions.

If dynamic activation fails
If the activation is rejected, HCD displays a panel that lists the messages and
reasons for a failure. From the displayed message list, you can request further
information. The message list can also be displayed by using the command
SHOWMSG ACTIVATE on any panel that has a command line (except on help
panels).

In some cases a dynamic activation may fail and HCD recommends recovery. HCD
displays a panel where you can specify whether you want to recover:
• If you confirm recovery by specifying Yes, HCD performs hardware-only changes.
  You can specify to recover in two ways:
  – To resume activation of the target IODF. That is, HCD tries to continue with
    the activation.
  – To reset the configuration to the source IODF. That is, HCD activates the
    configuration that existed before the failure occurred.
• If you do not confirm recovery by specifying No, HCD allows you to continue
  with software-only changes. Hardware changes are activated up to the point where
  the failure occurred.

Detection of illegal split/merge of LCU: If a request for activating a new
configuration causes a logical control unit (LCU) to be split or merged illegally,
HCD rejects the request. HCD considers an LCU to be:
Illegally split if a physical control unit (PCU) is removed from the LCU that has devices remaining connected to it.

Illegally merged if a physical control unit is added to an LCU that has devices connected to it.

HCD detects every split/merge during activation and informs you by message. The message tells between which PCU and which device the split/merge occurred; it also explains how to correct the condition.

For more information on how to recover after a system failure, refer to z/OS HCD Planning.

**How to activate if software-only changes are allowed**

The following procedure describes how to activate a configuration dynamically if only software changes are allowed. Refer to z/OS HCD Planning on information when both hardware and software configuration changes are allowed and when only software configuration changes are allowed.

1. On the Primary Task Selection panel, select *Activate or process configuration data*, and from the resulting panel select *Activate configuration dynamically*. HCD displays the following panel:

```
Activate Software Configuration Only

Only software changes are allowed.
Specify or revise the values for IODF activation.

Currently active IODF : SYS1.IODF31
  Configuration ID : MVSVM     MVS Testsystems on VM
  EDT ID . . . . . . : 00

IODF to be activated : SYS1.IODF31
  Configuration ID : MVSVM +
  EDT ID . . . . . . : 00 +

Test only . . . . . . : Yes (Yes or No)
```

The panel contains information about the currently active IODF.

2. On the Activate Software Configuration Only panel, you can change the fields that relate to the IODF that is to be activated. It is recommended that you first test an activation before you actually dynamically activate a configuration.

3. If the dynamic activation completed successfully, HCD displays a message.

**If dynamic activation fails**

If the activation is rejected, HCD displays a panel that lists the message(s) and reasons for a failure. From the displayed message list, you can request further information. The message list can also be displayed by using the command SHOWMSG ACTIVATE on any panel that has a command line (except on help panels).
Activate a configuration sysplex-wide

HCD offers you a single point of control for systems in a sysplex. You can now dynamically activate the hardware and software configuration changes for each system in a sysplex from any other system in the same sysplex. You can:

- Display active sysplex members
- Activate Software Configuration Changes Only
- Activate Software and Hardware Configuration Changes
- Switch IOCDS for the next POR

z/OS HCD Planning gives a detailed description of how to dynamically activate a configuration. It describes the prerequisites for a dynamic activation, explains when hardware and software changes or software-only changes are allowed, and describes the actions necessary to change your I/O configuration dynamically. The following sections describe how to use the HCD dialog for dynamically activating systems in a sysplex.

Displaying active sysplex members

Before you can make any change to a configuration in a sysplex, you must display the active sysplex member list. From this list you then select different actions.

1. On the Primary Task Selection panel, select Activate or process configuration data, and from the resulting panel select Activate configuration sysplex-wide. HCD displays the Active Sysplex Member List.

You can see the system names, and the processor IDs and partition names associated with the system names. You can also see the IODF to be activated, the name of the sysplex, the active IODFs, the configuration IDs and EDT IDs used for IPL, and the Activate status, which is empty initially.
HCD requests the information from the sysplex couple data set and the HSA of every CPC and displays it in a formatted list.

**Refreshing the Active Sysplex Member List:**

The Active Sysplex Member list will be refreshed whenever you press the Enter key. If a system joins the sysplex, it will be added to the list in alphabetical order. If a system leaves the sysplex, it will be deleted from the list.

**Empty Processor ID:**

The entry in the Processor ID column is empty when the operating system runs as a VM guest.

This does not mean that you cannot initiate an activation request for that system. It means that you have to specify the correct processor ID later on by yourself, as required.

2. Select a system name and specify *View the configuration status* from the context menu (or select action code SB ) to see the following information:
   - information about the currently active hardware configuration token stored in the HSA
   - information about the free space in the HSA.

### Activate software configuration changes only

The following procedure describes how to change the software configuration for one or more systems in a sysplex.

1. Select one or more systems from the Active Sysplex Member list (see Figure 92 on page 224) and the *Activate software configuration only* action from the context menu (or action code SF). The Activate Software Configuration Only panel is displayed.

   ![Activate Software Configuration Only](image)

   **Figure 93. Activate Software Configuration Only**

   All systems you have selected are shown together with the associated processor IDs.
**Empty Processor ID:**
If the Processor ID field is empty, prompt for the processor IDs and select the actual one.

**Empty Configuration or EDT ID:**
If the Config. ID and the EDT ID fields are empty, it is an indication that the ID of the currently active configuration is not defined in the IODF to be activated. Prompt for the new ID. Updating the processor ID, the Config. ID or the EDT ID fields might be required if your IODF to be activated contains IDs different from those displayed as default IDs.

2. On the Activate Software Configuration Only panel, update the fields of one or more systems. It is recommended that you first test an activation before you actually dynamically activate a configuration.

3. If the hardware token matches, the Valid. HW Ch. option is set to ‘Yes’. It is recommended to validate hardware changes when performing a software change. This is required when the configuration change contains coupling facility control units or devices.

4. After pressing the Enter key, the Active Sysplex Member list is displayed again, but now the Active Status column shows the status In progress. If you refresh the list occasionally, you can see that one system after the other completes the activation request. This is indicated by the status Messages.

5. Select a system and the View activate messages action from the context menu (or action code /SF580000m/SF590000). The messages returned from that system as the result of the activation request are displayed.

```
Message List
Save Query Help
--------------------------------------------------------------------------
Row 1 of 4

System name: SC52
/ Message Text
  _ IOS500I ACTIVATE RESULTS 084
  # ACTIVATE COMPLETED SUCCESSFULLY
  # NOTE = 0100,SOFTWARE-ONLY CHANGE
  # COMPID=SC1C3
*****************************************************************************
```

Figure 94. Message panel with ACTIVATE messages

6. If you do not need the messages any longer, you can delete them by using the option Delete activate messages (or action code /d).

**If dynamic activation fails**
If the activation is rejected, HCD displays a panel that lists the messages and reasons for a failure.

To get more information you also use the View configuration status option (or action code /v). This option provides you the same information as when entering the command D IOS,CONFIG(ALL) at the system console.
HCD offers you two possibilities after an activation failed:

- Select **Resume activation of target configuration** (or action code `/SF580000t/SF590000`) to force the activation of the system.
- Select **Reset source configuration** (or action code `/SF580000r/SF590000`) to reset the original configuration.

**Activate software and hardware configuration changes**

The following procedure describes how to change the software and hardware configuration for one or more systems in a sysplex.

1. Select one or more systems from the Active Sysplex Member list and the **Activate software and hardware configuration** action from the context menu (or action code `/a`). The Activate Hardware and Software Configuration panel is displayed.

   The systems you have selected are shown together with the associated processor IDs.

   **Empty Processor ID:**

   ```
   ┌─────────────── Activate Hardware and Software Configuration ───────────────┐
   │ Row 1 of 4 More: > │
   │ │
   │ Specify or revise the values for activation, then press Enter. │
   │ IODF to be activated: BBEI.IODF0A │
   │ System Processor Partition Config. EDT -FORCE Option- Switch Test │
   │ Name ID + Name ID + ID + DEVICE CANDID. IOCDS Only │
   │ SYSTE ME PROC E MYS E 0E No No — Yes │
   │ SYSTEMF PROCF PARTF MYSF OF No No — Yes │
   │ SYSTEM0 PROC0 PART0 MYS0 00 No No — Yes │
   │ SYSTEM1 PROC1 MYS1 01 No No — Yes │
   └────────────────────────────────────────────────────────────────────────────┘
   ```

   **Figure 96. Activate Hardware and Software Configuration**
If the Processor ID field is empty, prompt for the processor IDs and select the actual one.

*Empty Configuration or EDT ID:*

If the Config. ID and the EDT ID fields are empty, it is an indication that the ID of the currently active configuration is not defined in the IODF to be activated. Prompt for the new ID. Updating the Processor ID, the Config. ID or the EDT ID fields might be required if your IODF to be activated contains IDs different from those displayed as default IDs.

*Switch IOCDS:*

In the Switch IOCDS column, you can define the IOCDS name used for the next POR.

2. On the Activate Hardware and Software Configuration panel, update the fields of one or more systems. It is recommended that you first test an activation before you dynamically activate a configuration.

If you plan to delete a device for a specific system, specify Yes in the FORCE DEVICE field of that system. If you plan to remove a partition from the access or candidate list of a channel path belonging to a specific system, specify Yes in the FORCE CANDID. field of that system.

3. After updating the panel, press the Enter key. The Active Sysplex Member list will be displayed again, but now the Active IODF fields have changed for the affected systems and now contain the name of the IODF to be activated. If you have defined different processor IDs, configuration IDs or EDT IDs, and you have activated their configurations, these fields also have been changed.

4. Select a system and the View activate messages action from the context menu (or action code ). The messages returned from that system as the result of the activation request are displayed.

5. If you do not need the messages any longer, you can delete them by using the option Delete activate messages (or action code d).

*If Dynamic Activation Fails*

Refer to “If dynamic activation fails” on page 226 for information on how to proceed, if your activation has been rejected.

**Switch IOCDS for the next POR**

In addition to dynamically activating a sysplex, you can also specify the IOCDSs to be used for the next POR from the Active Sysplex Member List.

Refer to “Switch IOCDS for systems in a sysplex” on page 232 for a description of how to switch the IOCDS for the next POR.

**Build CONFIGxx member**

After dynamic changes have been made to a system it is recommended to update the corresponding CONFIGxx member to reflect these changes. HCD provides a function to build a CONFIGxx member containing the CHP, DEVICE, and SWITCH statements of the local system or of the selected system in a sysplex.

A CONFIGxx member can be built by:

- Selecting the Build CONFIGxx member action from the Activate or Verify Configuration panel (for the local system)
- Selecting the Build CONFIGxx member action from the Active Sysplex Member panel (for a system in a sysplex)
• Using a batch utility (see “Build I/O configuration data” on page 320 for details)

After selecting **Build CONFIGxx member**, the Identify System I/O Configuration panel is displayed (see Figure 90 on page 217). After selecting a system, and an I/O cluster name for managed channel paths, the Restrict Ports Eligible for Dynamic CHPID Management panel is displayed if the configuration contains managed channel paths for the selected I/O cluster. This panel shows all control units known by the selected system and manageable by DCM and their switch ports set to eligible for DCM (indicated by a ‘Y’). You can specify ports as ineligible for DCM by overtyping ‘Y’ with ‘N’.

![Restrict Ports Eligible for Dynamic CHPID Management](image)

**Figure 97. Restrict Ports Eligible for Dynamic CHPID Management**

The Build CONFIGxx Member panel is then displayed.

![Build CONFIGxx Member](image)

**Figure 98. Build CONFIGxx Member**

The initial value for the partitioned data set name is ‘SYS1.PARMLIB’.

If the specified CONFIGxx member already exists, the Confirm Update CONFIGxx Member panel is displayed.
If you select Update member, the CHP, DEVICE, and SWITCH statements are replaced and all other statements remain unchanged. If you select Replace member, the content of the CONFIGxx member will be CHP, DEVICE, and SWITCH statements exclusively. All other statements formerly present in the member will be removed.

The following illustrates sample generated statements:

* CHP, DEV AND SWITCH STATEMENTS GENERATED BY
  * BUILD CONFIGXX UPDATE REQUEST
  * 2001-01-09 13:56:28 IODF: BOKA.IODF38
  * PROCESSOR: FR38LPAR PARTITION: F38H OS CONFIGURATION ID: B710
  * I/O CLUSTER: UTCPLX38
  * CHP (00,01,04),ONLINE
  * CHP (06,07,08,09,0A,0B,0C,0D,0E,10),ONLINE
  * CHP (11),ONLINE,MANAGED
  * DEVICE (0B00-0B1F),(1C),ONLINE
  * DEVICE (1400-143F),(0C,22,33),ONLINE
  * DEVICE (1440-147F),(10,1C,44),ONLINE
  * SWITCH (B565,42),NODCM
  * SWITCH (B565,B6,BC-BE),DCM

The default name for the backup member is CONFBKxx. If the name is blanked out, no backup is saved.

You can also invoke this task in batch mode. See "Build I/O configuration data" on page 320 for a description of the job control information that you need to specify when building a CONFIGxx member.

**Process 'Display M=CONFIG(xx)' command**

HCD provides a dialog function to compare the information in the CONFIGxx member for the system in a sysplex with the hardware configuration. The comparison is carried out at the target system and any responses are displayed in a message list.

You can invoke the function from the Active Sysplex Member list by selecting the action **Process DISPLAY M=CONFIG(xx) command**. This displays the Process Display M=CONFIG(xx) Command panel.
Here, the suffix for the member to be used must be specified.

The results of this action are displayed in the HCD message panel.

---

**Switch IOCDS for next POR**

HCD allows you to specify an IOCDS that will be used for the next POR either while building IOCDSs or as a separate action without the need to build an IOCDS. Depending on the environment you are working, you have to use different panels to switch the IOCDS:

- Switch IOCDS for processor without SNA address defined
- Switch IOCDS for a processor in an S/390 microprocessor cluster with SNA address defined
- Switch IOCDS for systems in a sysplex

---

**Switch IOCDS for processor without SNA address**

The following procedure is only recommended for processors that do not have an SNA address defined. For a detailed description of the following dialog sequence, refer to "Build an IOCDS" on page 201.

1. On the Primary Task Selection panel, specify the name of a production IODF and select *Activate or process configuration data*.
2. From the resulting panel select *Build IOCDS*. HCD displays the Processor List panel.
3. On the Processor List panel, select the processor and press the Enter key. HCD displays the IOCDS List panel (see Figure 85 on page 201).
4. On the IOCDS List panel, select the IOCDSs you want to use for next POR and select *Switch IOCDS* from the context menu (or action code 5).

If the HSA token is available, the HSA token is compared with the processor token. If the HSA token matches the processor token, the *Switch IOCDS* action is performed. A warning message is issued, if the date in the IOCDS update record is an earlier date than the date of the last CSS update. If the HSA token does not match the processor token in the IODF, the action *Switch IOCDS* is not performed.

If the HSA token is not available, the serial number of the processor defined in the IODF is compared with the serial number of the active processor. If the serial numbers cannot be found, the types of the processors are compared. If the
processor definition in the IODF matches the active processor, the action Switch IOCDS is performed, otherwise the procedure is not performed.

**Switch IOCDS for processors in an S/390 microprocessor cluster with SNA address defined**

The following procedure describes how to build an IOCDS for processors in an S/390 microprocessor cluster with an SNA address defined. For a detailed description of the following dialog sequence, refer to “Build S/390 microprocessor IOCDSs” on page 203.

1. On the Primary Task Selection panel, select Activate or process configuration data and from the resulting panel select Build and manage S/390 microprocessor IOCDSs and IPL attributes. The S/390 Microprocessor Cluster List panel is displayed (see Figure 86 on page 204).

2. On the S/390 Microprocessor Cluster List panel, select the CPCs for which you want to switch the IOCDSs and Work with IOCDSs from the context menu (or action code ₳). HCD displays the IOCDS List panel (see Figure 87 on page 205).

3. Use the Switch IOCDS action (or action code ₳) to mark an IOCDS as the IOCDS that is used for next POR. The Status field will be set accordingly.

You can only switch to an IOCDS that has an IOCDS/HSA token match or to an IOCDS of a processor that is not activated (‘POR-required’ status).

**Switch IOCDS for systems in a sysplex**

In addition to dynamically activating a sysplex, you can also specify the IOCDSs to be used for the next POR.

1. Select one or more systems from the Active Sysplex Member list (see Figure 92 on page 224) and the Switch IOCDS for next POR action from the context menu (or action code ₳). The Switch IOCDS panel is displayed.

   Specify the IOCDS(es) for next POR, then press Enter.

   ![Switch IOCDS panel](image)

2. In the column Switch IOCDS, specify the IOCDS that is to be used for the next POR and press the Enter key.

**Specify an IODF for IPL**

z/OS HCD Planning gives a detailed description of how to specify an I/O configuration at IPL. This topic summarizes the main aspects you have to consider when specifying an IODF for IPL.
Note:
A production IODF must have a single extent. If the production IODF has multiple extents, the IPL process results in a WAIT state (wait state code '0B1', reason code '002'). HCD issues error message CBDA009I if a production IODF cannot be built in a single extent.

IODF processing at IPL
When you perform an IPL, the production IODF that defines the configuration to the system is selected and used.

On the LOAD parameter, you specify the device containing the IODF for IPL and the identifier for the LOADxx member. In the LOADxx member of SYSn.IPLPARM or SYS1.PARMLIB, you identify the IODF by the IODF statement. The IODF statement consists of an IODF prefix and an IODF suffix.

- The IODF prefix is an 8-byte high-level qualifier of the IODF data set name. For example, BPAN is the IODF prefix for the IODF data set BPAN.IODF01.
- the IODF suffix is the two-digit hexadecimal number that is part of the IODF name. For example, 01 is the IODF suffix for IODF01. If you do not specify a suffix, the system searches for an IODF sequentially in a numerically ascending order starting with the IODF suffix 00. If you specify ** as the suffix, the system uses the descriptor fields to find the current IODF.

During IPL, the system uses the LOADxx member that it finds first when searching in the following order:
1. The system first searches the IODF volume for SYS0.IPLPARM through SYS9.IPLPARM, in that order.
2. If it does not find a SYSn.IPLPARM, it searches the IODF volume for a SYS1.PARMLIB.
3. If it does not find SYS1.PARMLIB on the IODF device, it searches for SYS1.PARMLIB on the IPL device.
4. If it does not find a SYS1.PARMLIB on the IPL device, a coded non-restartable wait state is loaded (WAIT code X'0B1').

For a detailed description of this process refer to z/OS HCD Planning.
Chapter 9. How to print and compare configuration data

Overview
This information unit describes how to:

- Print configuration reports (channel subsystem, switch, OS configuration data, and CTC connections)
- Print a report of the I/O paths of the actual system compared to the defined I/O configuration
- Print a report of the supported hardware or an I/O definition reference
- Create or view a graphical report of the I/O configuration
- Compare functions (IODFs and CSS/operating system views)
- Print List Panels
- View and print the activity log

Print configuration reports
You can use HCD to generate several types of reports about the configuration data in an IODF:

- Channel Subsystem (CSS) Report
- Switch Report
- Operating System (OS) Report
- CTC Connection Report
- I/O Path Report
- Supported Hardware Report
- I/O Definition Reference

In the HCD profile definition, you have the option of printing textual reports in upper case only or defining the number of lines per page (see "Options for text reports" on page 28).

Examples of these reports are shown in "Appendix B. Configuration reports" on page 377.

Channel Subsystem Report
The Channel Subsystem Report contains all configuration data that is used by the channel subsystem. If the IODF contains data for more than one processor or logical partition, you can limit the report to the data for one processor or partition. If you limit the report to one partition, it will generate information only for channel paths, which have the partition in the access list. Channel paths that have that partition in a candidate list will not be taken into consideration.

You can select four types of reports:

- **CSS summary reports** include summary reports about:
  - Processors
  - Channel subsystems
  - Partitions
  - IOCDSs
  - Channel paths
  - Control units
  - Devices
The processor and partition reports are not printed if you limit the CSS summary reports to the data for one processor or partition.

- **Channel path detail reports** include reports about:
  - Channel paths
  - CF channel path connectivity
- **Control unit detail report**
- **Device detail report**

**Switch Report**

The Switch Report contains details about the switch definition, its configurations and the port definitions.

If the IODF contains data for more than one switch, you can limit the report to the data for one switch and the configurations for this switch. In this case, you do not get a switch summary report.

**Operating System Report**

The Operating System Report contains the configuration data that is used by z/OS or z/VM. If the IODF contains data for more than one operating system, you can limit the report to the data for one operating system. You can select three types of reports:

1. The OS device report includes reports about operating systems and OS devices. The operating system summary report is not printed if you limit the OS device report to the data for one operating system.
2. OS console report
3. EDT report (MVS-type only)

**CTC Connection Report**

The CTC Connection Report contains CTC connections of your configuration that are defined through a switch. In case of incorrect definitions, the report also contains a list of messages with diagnostic information.

If the IODF contains more than one processor or logical partition, you can limit the report to the data for one processor or partition.

**I/O Path Report**

The I/O Path report shows the physically sensed I/O paths (with physical types) of the active system compared with the logical definitions of the paths (also the object types) of a specific IODF.

On the Limit Reports panel (Figure 102 on page 238), the active configuration to sense the configuration from, can be specified by indicating a SYSPLEX and/or SYSTEM name. If nothing is specified, the data is taken from the local system.

For more information, see "Job statement information used in panels" on page 75.

See "Prerequisites" on page 9 for the prerequisites for the I/O Path report.
Supported Hardware Report

The Supported Hardware Report contains information about the processors, control units, and devices supported in your installation. This report can only be generated using the batch facility as described in "Print configuration reports" on page 324.

This report is generated directly from the UIMs. Therefore, it reflects the latest UIM levels installed.

I/O Definition Reference

The I/O Definition Reference contains a description of the parameters to define the device to the Channel Subsystem, and a description of the parameters and features to define the device to the operating system.

This report is generated directly from the UIMs. Therefore, it reflects the latest UIM levels installed.

How to print a textual report

1. On the Primary Task Selection panel, select Print or compare configuration data.

2. On the resulting Print and Compare Configuration Data panel, select Print configuration reports. HCD then displays the following panel:

   ┌────────────────────── Print Configuration Reports ──────────────────────┐
   │ │
   │ Select the types of report you want, and specify the values below. │
   │ │
   │ IODF name : 'DOCU.IODF01' │
   │ │
   │ Types of report Limit report(s) │
   │ _ CSS report 1  1. Yes │
   │ _ Switch report 2. No │
   │ _ OS report │
   │ _ CTC connection report │
   │ _ I/O path report │
   │ Job statement information │
   │ // JOB (ACCOUNT), 'NAME' │
   │ //* │
   │ //* │
   │ //* │
   │ //* │
   │ F1=Help F2=Split F3=Exit F5=Reset F9=Swap F12=Cancel │
   └─────────────────────────────────────────────────────────────────────────┘

   Figure 101. Print Configuration Reports

3. Enter the required data.

   When you select a CSS or OS report an additional panel appears on which you can select one or more report types.

   If a data set is pre-allocated the logical record size must be 133. You can allocate the report output data set HCDRPT using the job step name GO.
4. When you select to limit the reports, possible for CSS, Switch, OS and CTC connection reports, the Limit Reports panel appears that allows you to specify a processor ID, partition name, OS configuration ID and a switch ID. When you select an I/O Path report, the Limit Reports panel always appears. This is because limiting an I/O Path report is required. Default values for the processor ID, the partition name (for an LPAR processor) and the OS configuration ID are then already filled in. These values are based on the active configuration. The system name identifies the system of a sysplex for that the I/O Path report is to be generated. The default is the local system. The sysplex name specifies the sysplex of the system for that the I/O Path report is to be generated. If you specify the sysplex, you must also specify the system name. If you do not specify the sysplex, the system name is the VTAM application name of the host that the I/O Path report is to be generated for. If you selected to print more than one report type, the limitations specified on the Limit Reports panel apply to all of them.

When limiting a CSS report to a single partition, the report will show channel paths, control units and devices attached by the access list as well as those attached by the candidate list.

---

### Limit Reports

To limit the reports, specify the following criteria related to the IODF in access.

<table>
<thead>
<tr>
<th>Field</th>
<th>Applicable for</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor ID</td>
<td>CSS, CTC, I/O path reports</td>
</tr>
<tr>
<td>Partition name</td>
<td>CSS, CTC, I/O path report</td>
</tr>
<tr>
<td>OS configuration ID</td>
<td>OS, I/O path report</td>
</tr>
<tr>
<td>Switch ID</td>
<td>switch report</td>
</tr>
<tr>
<td>Sysplex name</td>
<td>I/O path report</td>
</tr>
<tr>
<td>System name</td>
<td>I/O path report</td>
</tr>
</tbody>
</table>

Specify the sysplex and system name to gather the actual configuration from. (Blanks default to the local system.)

F1=Help  F2=Split  F3=Exit  F4=Prompt  F5=Reset  F9=Swap  F12=Cancel

Figure 102. Limit Reports

The submitted job only starts if the IODF is accessed in read mode. If it is accessed in update mode, the job waits until you access another IODF or exit HCD.

You can also print reports using the batch mode. See ["Print configuration reports"](page 324) for a description of the job control information that you need to specify when printing a report.

---

### Create or view graphical configuration reports

HCD offers you to print and view a graphical representation of the I/O configuration based on the definitions in the IODF. The reports can be either stored in a data set for printing on an AFP printer (such IBM 3820 or IBM 3800) or via GDMD later on, or displayed on an IBM 3270 terminal with graphical capability.

The graphical report function allows you to print or view five types of reports:

- The **LCU report** shows all logical control units defined for one processor.
• The **CU report** takes a control unit as focal point and shows the connections to the processors and the devices of the IODF. On request, it shows the switches as well.

• The **CHPID report** shows the defined channel paths for a processor and the switches, control units, and devices attached to the CHPID.

• The **Switch report** takes a switch (ESCON director) as focal point and shows the processors, chained switches, and control units with devices attached to the switch.

• The **CF connection report** takes a coupling facility as focal point and shows all connections that exist between the coupling facility and the other processors defined in the IODF.

**Prerequisites**

**For printing**
To process the reports for printing you need one of the following:
- BookMaster Release 3.0 or higher
- DCF/GML Release 4.0
- GDDM Version 2.1 or later

To print the reports you need an AFP printer, such as IBM 3820 or IBM 3800 (not required for GDDM).

To store the output in GDF format, you have to use a terminal with a screen size of 80 columns, for example a 3278-2.

In the HCD profile, specify whether the output of this function can be processed with BookMaster, DCF, GML, or GDF (keyword GCR_FORMAT). BookMaster is the default. To use DCF or GML format, specify a mono-space font using the keyword GCR_FONT. For example, specify `GCR_FONT = X06T20` (Gothic Text 20-pitch) for a 3820 printer. For more information about the HCD profile, refer to "Defining an HCD profile" on page 23.

**For viewing**
To view the report on an IBM 3270 terminal with graphical capability, GDDM must be installed on your system. Refer to "Setting up HCD" on page 19 on how to setup the GDDM support.

Use a terminal with a screen size of 80 columns, for example a 3278-2. This display function does not work on terminals (or terminal emulations) with a screen size of 132 columns. HCD uses ISPF to create the GDDM display, which means that terminals running in partition mode or terminals with multiple screen widths, including 3290 and the 3278 Mod 5, are not supported for graphics interface mode.

In the HCD profile, you can specify the colors used for displaying the graphic (see "Defining an HCD profile" on page 23). If you change the default colors, make sure that foreground and background color match.

**How to create a graphical configuration report**
To print or view a configuration, use **Create or view graphical configuration report** on the Primary Task Selection panel. The dialog is described in "Using the Create or View Graphical Configuration Report’ option” on page 240.
To view objects in context of their attached objects you can also select an object from an object list panel and use the View graphically action from the context menu (or action code ). The following object lists support this possibility:
- Channel path list
- Control unit list
- I/O device list (only for devices that connect to a control unit)
- Partition list
- Switch list

For example, from the Switch List panel, you can view a switch together with all objects that are attached to the switch. This can help you, for example, to immediately verify your definitions while defining your configuration.

**Using the 'Create or View Graphical Configuration Report' option**

1. Select Create or view graphical configuration report. on the Primary Task Selection panel.
2. The Create or View Graphical Configuration Report panel appears.

   ┌────────────── Create or View Graphical Configuration Report ────────────────┐
   │ Select the type of report you want, and specify the values below.             │
   │ IODF name . . . . . . : 'BPAN.IODF00.WORK'                                  │
   │ Type of report . . . . 2 . 1 . LCU report                                  │
   │ 2. CU report                                                               │
   │ 3. CHPID report                                                           │
   │ 4. Switch report                                                          │
   │ 5. CF connection report                                                    │
   │ Processor ID . . . . . ________ + (for an LCU or a CHPID report)            │
   │ Partition name . . . . ________ + (to limit an LCU or a CHPID report)       │
   │ Output data set . . . . 'BPAN.IODF00.PRINT'                                 │
   │ Output . . . . . . . . . . 1 . Write to output data set                      │
   └──────────────────────────────────────────────────────────────────────────────┘

Type of report: Select the type of report you want to create.
Processor ID and Partition name: Enter the required data for an LCU or a CHPID report.
Output: Select whether you want to write the output to an output data set for printing or to display the output on your terminal.
Output data set: For BookMaster, GML, or DCF processing, the output data set must be a sequential data set or a member of a partitioned data set. If the PDS or the sequential data set does not exist, it will automatically be allocated (record length 200, record format fixed blocked).
For creating output for GDF, specify a member of a partitioned data set. If the data set does not exist, it will automatically be allocated (record length 400, record format fixed blocked). If the data set already exists, it is overwritten with the new data, you are not asked to confirm replacement. The output is written into different members, one for each segment (see "Printing the output" on page 241 for a definition what a segment is). The member names are up to eight characters long. They are derived from taking up to seven characters from the member name specified in the output data set field and adding a number. For instance, if the name was specified as 'BPAN.IODF00.PRINT(SWITCHES)', the member names would be SWITCH01, SWITCH02, ... SWITCH10, and so on.
3. When pressing the Enter key, the Define Report Layout panel appears. The following example shows the panel for a CU report.

```
 ┌──────────────────────────── Define Report Layout ──────────────────────┐
 │ │
 │ │
 │ Specify the values below for report type: CU │
 │ │
 │ Include index ... 1. Yes Include partitions ... 1. Yes │
 │ 2. No 2. No │
 │ │
 │ Only for a CU or CHPID report: │
 │ Include CTC, CF CUs ... 1. Yes Include switches ... 1. Yes │
 │ 2. No 2. No │
 │ │
 │ Show CU ...... 1. Serial number │
 │ 2. Description │
 │
 │ To limit a CU report, specify only one of the following: │
 │ Range ...... - - │
 │ Type ...... - - + │
 │ Group ...... + │
└────────────────────────────────────────────────────────────────────────┘
```

Select what you want to include in the graphical report. To limit the control units to be shown in a CU report, you can specify either the range, type, or group (for example, DASD) of the control units.

4. After pressing the Enter key, the report is written to an output data set or shown on the terminal. See "Printing the output" and "Viewing the output" on page 242 on how to proceed.

You can also create graphical reports using the batch mode. See "Create a graphical configuration report" on page 327 for a description of the job control information that you need to specify when printing a report.

**Printing the output**

1. Process the output data set using BookMaster, DCF, GML, or GDF. While processing it is recommended that you specify the following parameters:

   **Indexing**
   To print the index you selected on the Define Report Layout panel specify INDEX for BookMaster and GML processing

   **Full page**
   To use the full page for the report SYSVAR S is to be set to 1 for DCF processing. Note that in many installations offset is used as default.

   **Rotate printout**
   Specify a parameter to print the report in landscape format, that means to rotate the printout by 90 degree.

2. Print the report.

HCD tries to display a report on one page. If a report is too large for one page, HCD divides the report into segments and shows each segment on an extra page. If a CU report, for example, shows more than 8 control units, HCD shows the control units of the same type in one segment on an extra page. If a segment is too large for one page, HCD continues the segment on the next page. You can specify the GCR_COMPACT=YES keyword in the HCD profile to see more objects on one page.

For an example of a report, see "Graphical configuration reports" on page 435.
Viewing the output

The following figure shows an example of the panel when viewing a report.

HCD tries to display the entire configuration on one panel. If a report is too large for one panel, HCD divides the report into segments and shows each segment on an extra panel. If a CU report, for example, shows more than 8 control units, HCD shows the control units of the same type in one segment on an extra panel. You can move from segment to segment by using the Next and Previous function keys.

If a segment is too large for the panel, you can scroll upwards, downwards, to the left and to the right.

The following list describes specific function keys while viewing a graphical report.

F4=Jump
You can use this function only when you have displayed the graphical report using the task Create or view graphical configuration report on the Primary Task Selection panel. It displays the action list of the HCD dialog that contains the object you selected with the cursor. Any change made to the configuration on the action list will not be reflected in the graphical report when you return to it. Use the REFRESH command to reflect the changes made to the objects currently shown on the display (see "Refresh Command" on page 244).
F5=Zoom in
   Makes the graphical display bigger, so you can see the details of an object.
   The position of the cursor identifies the lower left corner of the part you
   want to enlarge. If the cursor is not positioned, HCD takes the center of
   the currently displayed report.

F6=Zoom out
   Makes the graphical display smaller, so you can see more of a report on
   one panel.

F7=Up
   Scrolls upwards.

F8=Down
   Scrolls downwards.

F10=Previous
   Moves to the previous segment of a report, if any.

F11=Next
   Moves to the next segment of a report, if any.

F19=Left
   Scrolls to the left.

F20=Right
   Scrolls to the right.

Save command
You can use the SAVE command to store a graphical configuration displayed on
your screen in a member of a partitioned data set (PDS) in GDF format for
printing with GDDM.

Before you use this command, the partitioned data set has to be allocated to
ddname ADMGDF. The records of this PDS must have a record length of 400.

Specify SAVE (or just SA) and the member name, into which you want to save the
data, on the command line. The syntax of the SAVE command is as follows:

```
SAVE Command

SAVE member_name
SA
```

member_name
   specifies the name of the member to contain the graphical configuration you
   want to print. Use a different name for each SAVE command, otherwise the
data will be overwritten. The name may be up to eight characters long.

Locate Command
You can use the LOCATE command to center and highlight a specific object of the
report. When you locate a partition, channel paths, or coupling facility partition,
the connections to the object are highlighted instead of the object itself.

Specify LOCATE (or just L) and the object you want to locate in the command line.
For objects that are associated with another object (for example, CHPIDs that are
associated with a processor), you have to specify this object as well.
Locating multiple objects with the same ID: The LOCATE command first searches for an object in the currently displayed panel. If the object is not found it starts with the first segment and continues to the right until an object has been found. To find the other objects with the same ID, use the + and − parameter of the LOCATE command. The + parameter searches for the next object to the right, the − parameter to the left.

The syntax of the LOCATE command is as follows:

```
LOCATE command
```

- **CF** Coupling Facility
- **CHPID** Channel path
- **CU** Control unit
- **DEV** Device
- **PART** Partition
- **PR** Processor
- **SW** Switch

**Example:** To locate CHPID 27 of processor SYSA, type
```
L CHPID SYSA 27
```

**Refresh Command**

When you jump to an action list using F4=Jump and change objects in this list, use the REFRESH command on return to the graphical display to refresh the graphic with the changes made.

REFRESH applies to all objects that were currently shown on the graphical report when pressing the F4=Jump key, that is, added objects or those, for which you changed the ID, will not appear in the refreshed graphic.

Valid abbreviation of the REFRESH command is RE.
How to print list panels

You can use the SAVE command to save the data that is currently displayed on HCD list panels into a data set. The data set can be used for printing.

You can also save and print lists that are filtered. If you use the Set Filter option from the Filter action bar choice, you can, for example, print all channels of a processor that are not connected to a control unit.

The SAVE command is available on the following list panels:
- Operating system configuration List
  - EDT List
  - Esoteric List
  - Esoteric/Device List
  - Generic
  - Generic/Device List
  - Console List
- Switch List
  - Port List
  - Switch Configuration List
  - Port Matrix
- Processor List
  - Channel Subsystem List
  - Channel Path List
  - Partition List
- Control Unit List
- I/O Device List
- Configuration Package List
  - Configuration Package Object List
- CTC Connection List
- I/O Path List

On these list panels, perform the following steps:
1. Enter the command
   SAVE
2. The Save List panel appears:

   ┌──────────────────────────────── Save List ────────────────────────────┐
   │ │
   │ Specify the following values. │
   │ │
   │ Output data set ________________________________ │
   │ │
   │ Additional remarks (for example, the filter criteria) │
   │ _____________________________________________________________________ │
   │ _____________________________________________________________________ │
   │ │
   └───────────────────────────────────────────────────────────────────────┘
   
   The output data set can be a sequential data set or a member of a partitioned data set. If the data set does not exist, it will be automatically allocated (record length 300, record format fixed block). The name of the data set is saved for the next HCD session.

   In addition you can specify two lines of optional comments that will be shown under the header of your report.
3. A result of a printed data set may look like the following example. Note that the column headers are the same as shown on the panel itself.

```
/|SF5800001|/SF590000|
Header with IODF name, date, time, list name
/|SF5800002|/SF590000|
Optional comments specified on Save List panel
/|SF5800003|/SF590000|
Identifier of higher-level object, for example the processor name (and channel subsystem ID if applicable) when you print the channel path list
/|SF5800004|/SF590000|
Column headers as shown on the panel itself
```

HCD compare functions

HCD offers functions to compare IODFs and device definitions for a selected CSS or operating system and to report the differences:

- “Compare IODFs”
- “Compare CSS / operating system views” on page 250

Compare IODFs

You can use the Compare IODFs function to compare two IODFs and report the differences between them. For greater clarity, you can limit the compare reports to certain perspectives of the IODF:

- The Processor Compare Report shows differences in the properties of channel subsystems, partitions, CHPIDs, control units, and devices.
• The **Switch Compare Report** shows differences in the properties of switches and switch configurations.
• The **OS Configuration Compare Report** shows differences in device parameters, in features, in EDTs, in esoterics, in generics defined for EDTs, and consoles.

To compare IODFs, do the following:
1. Select **Print or compare configuration data** on the Primary Task Selection panel.
2. On the Print or Compare Configuration Data panel, select **Compare IODFs**. The following panel is displayed:

   ![Compare IODFs Panel](image)

   **Figure 104. Compare IODFs**

   On this panel, select one or more compare report(s). In addition, you can set the limit option. When the limit option is set, the related limiting panels will come up.

   On the **Limit Processor Compare Reports** panel, you can limit the processor compare reports by selecting one or more of the specific compare reports. You can limit the reports by specifying values for a processor or either by specifying values for a channel subsystem or a partition. It is possible to compare an SMP processor to a channel subsystem of an XMP processor. If you want to limit by processor, you must specify the processor IDs for both IODFs. If you limit the processor compare report by partition name, you receive the following results:
   • The report will contain the channel subsystem in which the partition is defined.
   • Channel path compare will only contain channel paths which have the limiting partition in their access or candidate list.
   • Control unit compare will only include the control units related to channel paths which have the limiting partition in their access or candidate list.
   • Device compare will only include the devices connected via channel paths which have the limiting partition in their access or candidate list.
On the **Limit Switch Compare Reports** panel, you can limit the switch compare reports by one or more of the specific compare reports. In addition, you can limit the reports by specifying a switch ID for both, the new and the old IODF.

![Limit Switch Compare Reports](image)

On the **Limit Operating System Compare Reports** panel, you can limit the operating system compare reports by one or more of the specific compare reports. In addition, you can limit the reports by specifying an operating system ID for both, the new and the old IODF.

![Limit Operating System Compare Reports](image)
How to print a Compare IODFs Report

After you have selected the specific pairs for the compare reports, you can decide what print options to use. The print options are shown on the Select Print Options panel, see Figure 108.

If you do not select an option (by only pressing the Enter key), the default print options are used (Print inserted data, and Print deleted data).

Examples of these reports are shown in “Appendix B. Configuration reports” on page 377.

When you do not select any option, HCD prints a report of

- IDs of added or deleted objects, and those objects, that have added or deleted relations
- IDs of added relations
- Attributes of objects that are different in both IODFs
When you select the Print inserted data option, HCD prints a report of all attributes, and relations of added objects.

When you select the Print deleted data option, HCD prints a report of all attributes, and relations of deleted objects.

When you select the Print unchanged data option, HCD prints a report of all attributes, and relations of unchanged objects.

When you select the Print unchanged item IDs option, HCD prints a report of the IDs of unchanged objects. But this applies only, if the Print unchanged data option is not selected.

**Compare CSS / operating system views**

You can use the Compare CSS / operating system views function to compare the device definitions of a selected CSS and OS configuration, showing which devices (their numbers and types) are defined to either the CSS or the OS, or both. By using this function you can find out the differences between a hardware (channel subsystem/CSS) and software (operating system/OS) definition in the currently accessed IODF.

![Compare CSS / OS Views](image)

*Figure 109. Select Print Options (for CSS/OS Compare only)*

When using the Compare CSS / operating system views function, you are offered the possibility of selecting what to print on the Compare CSS / OS Views panel, shown in Figure 109. The Print all data option prints a report of all devices either defined in the CSS or the OS.

The Print different data option prints a report of the devices that differ as follows:

- Defined for the CSS, but not for the OS.
- Defined for the OS, but not for the CSS.
- Defined for both, but of different device type.
If you limit the Compare CSS/OS Views Report for the CSS-side to one partition, it will generate information only for those devices that are attached to the channel paths, that have the limiting partition in the access or candidate list.

### View and print the HCD activity log

The HCD activity log is described in “Activity logging and change logging” on page 54.

To look at the activity log for an IODF,

1. Select *Print or compare configuration data* from the Primary Task Selection panel.
2. On the Print or Compare Configuration Data panel, select *View the activity log* or *Print the activity log*.

The log has the same format in both cases. For browsing, the log is displayed by the ISPF/PDF browse facility. For printing, the log is written to the ISPF list data set.

**Note:** HCD maintains an activity log only if this was requested when the IODF data set was created.
Chapter 10. How to query supported hardware and installed UIMs

Overview

This information unit describes how to view system data about:

- Supported processors
- Supported switches
- Supported control units
- Supported devices
- Supported installed UIMs

Your z/OS system has several tables and modules that contain data about the general characteristics of processors, switches, control units, and devices in the system.

HCD uses this data to validate your configuration definition. You might want to look at it during the definition task; it helps you select the correct characteristics when you define your hardware units. You can also use the supported hardware report as a help when defining your configuration. See "Supported Hardware Report" on page 401 for an example of a supported hardware report and "I/O Definition Reference" on page 431 for an example of an I/O definition reference.

You can view the system data by selecting Query supported hardware and installed UIMs from the Primary Task Selection panel or the Query action from the action bar. Then select the subtask or pull-down choice that you want.

Query supported processors

The option List supported processors shows which processors are supported by the system. You can also see which features each processor supports, and its capabilities (such as what types of channel paths the processor supports).
The two lines in Figure 110 marked with \[1\] and \[2\] illustrate that processors with different support levels generate more entries in the list of supported processors.

Horizontal scrolling displays additional information on the processors.

A # preceding a line indicates that this line and the previous line belong together, because the information of a horizontally scrolled screen does not fit in one line.
You can view channel path, control unit, and device information by placing the cursor in front of a processor and pressing the Enter key. As a sample of this type of information, you see the channel path information below:
Pressing the Enter key gives you a list of allowed channel path type mixtures. Pressing the Enter key again gives you similar information for control units and devices.

**Note:**
For zSeries processors, you can retrieve an explanation of the processor support level: Position the cursor on the processor support level description and press PF1 to get an enumeration of functions provided by this support level.

### Query supported switches

The option *List supported switches* shows the characteristics of each type of switch in the system, such as the port range and the supported channel attachments of each switch.

**Supported Switches**

ENTER to continue.

<table>
<thead>
<tr>
<th>Switch Type</th>
<th>Port Numbers</th>
<th>Supported Channel Attachments</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCS</td>
<td>00 - FF</td>
<td>FC</td>
</tr>
<tr>
<td>2032</td>
<td>00 - FF</td>
<td>FC</td>
</tr>
<tr>
<td>9032</td>
<td>C0 - FB</td>
<td>CNC, CTC, CVC, CBY</td>
</tr>
<tr>
<td>9032-3</td>
<td>00 - FB</td>
<td>CNC, CTC, CVC, CBY</td>
</tr>
<tr>
<td>9032-5</td>
<td>04 - FB</td>
<td>CNC, CTC, CVC, CBY, FCV</td>
</tr>
<tr>
<td>9033</td>
<td>C0 - CF</td>
<td>CNC, CTC, CVC, CBY</td>
</tr>
</tbody>
</table>
The option List supported control units displays a panel showing a list of available control unit groups, for example the DASD control unit group. Select one control unit group to limit the list of supported control unit types. The Supported Control Units panel appears showing the characteristics of control unit types contained in a group.

Horizontal scrolling displays additional information on supported channel path type attachments.

You can view which devices can be attached to a certain control unit type by placing the cursor in front of a control unit and pressing the Enter key. As a sample of this type of information, you see the CU - Device Attachment List below:
Query supported devices

The option List supported devices displays a panel showing a list of available device groups, for example the DASD device group. Select one device group to view characteristics of device types contained in this group. You can limit the list of device types to view only the device types supported by a specific operating system type. After selecting a group of devices, the Supported Device Type List panel appears.

You can also see what control units each I/O device type can be attached to by placing the cursor in front of a device and pressing the Enter key.

As a sample of this type of information, you can see the Device - CU Attachment List below:
Query installed UIMs

The option List installed UIMs shows which UIMs are available in the system and which I/O device types are supported by each UIM.

A Y (for Yes) in the E (for Error) column indicates that the respective UIM is in error and treated as not existing.

A # sign in front of a UIM name indicates that it cannot be selected because it is flagged in error.

For each of the installed UIMs you can view a list of supported devices by placing the cursor in front of a UIM and pressing the Enter key. As a sample of this type of information, you can see the View Supported Devices list below:
This UIM supports the listed device types.

UIM name: CBDUS002
UIM FOR 3375, 3380, 3390, 3995-151/153, 9345

ENTER to continue.

Generic
-- or --

<table>
<thead>
<tr>
<th>Device Type</th>
<th>VM D/T</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3375</td>
<td>3375</td>
<td>Direct Access Storage Device</td>
</tr>
<tr>
<td>3380</td>
<td>3380</td>
<td>Direct Access Storage Device</td>
</tr>
<tr>
<td>3380-CJ2</td>
<td>3380</td>
<td>Direct Access Storage Device</td>
</tr>
<tr>
<td>3390</td>
<td>3390</td>
<td>Direct Access Storage Device</td>
</tr>
<tr>
<td>3995-151</td>
<td>3390</td>
<td>Direct Access Storage Device</td>
</tr>
<tr>
<td>3995-153</td>
<td>3390</td>
<td>Direct Access Storage Device</td>
</tr>
<tr>
<td>9345</td>
<td>9345</td>
<td>Direct Access Storage Device</td>
</tr>
</tbody>
</table>

F1=Help    F2=Split    F3=Exit    F7=Backward    F8=Forward
F9=Swap    F12=Cancel   F22=Command
Chapter 11. How to migrate existing input data sets

Overview

This information unit describes how to:

- Prepare the input data sets before migrating them
- Migrate the input data sets using the HCD dialog
- Migrate the input data sets using the HCD batch utilities
- Replace existing configuration data through migration
- Change I/O configurations by editing data sets
- Understand and resolve errors that occurred during migration

HCD allows you to migrate existing configuration data that was defined in IOCP, MVSCP, and HCPRI0 input data sets to an IODF.

As input, you can use one of the following:

- An IOCP input data set
- An MVSCP input data set
- An HCPRI0 input data set

You can also use the migration to create I/O definitions by editing control statements. Data sets containing the statements corresponding to a specific IODF can be generated using a batch utility. Refer to "Build I/O configuration data" on page 320 for details on which data sets can be built and how to run the build process.

When migrating from input data sets, HCD checks the syntax of the input statements and runs a validation process that checks that the definitions being migrated do not conflict with the I/O configuration rules and with existing definitions in the IODF or with other definitions being migrated.

If HCD detects an error in the input data sets, it issues messages after the migration process has ended.

Note: The migration function has a prerequisite to the High Level Assembler.

Migration sequence

If you want to migrate more than one input data set into a single IODF, comply to the following migration sequence:

1. Migrate all IOCP input data sets
2. Migrate all MVSCP or HCPRI0 input data sets

LPAR considerations

If you have a combined IOCP/MVSCP input data set containing definitions for more than one LPAR and the same device number specified for more than one LPAR, migrate this input data set in the following way:

1. Migrate the input data set as IOCP only input data set.
2. Remove the duplicate device number definition and repeat the migration as an MVSCP only input data set. In the appropriate IODEVICE statement, specify the control unit number the device attaches to by means of the CUNUMBR parameter.

# Preparing your input data sets for migration

Before you can successfully migrate IOCP, MVSCP, or HCPRIO input data sets, you may need to change the input because HCD does a more rigorous checking of the input statements than IOCP or MVSCP.

To ensure that the migration is successful and that the resulting IODF accurately reflects the physical configuration, ensure that your input data sets apply to the validation rules described below. For assistance when checking the definitions in the input statements, you can use the following possibilities:

- Select *Query supported hardware and installed UIMs* from the Primary Task Selection panel
- Use the *Query* action bar choice
- Use the batch facility "Print a Configuration or Supported Hardware Report"

## Data requiring attention

This section details which data may need to be changed and how to change it to ensure a successful migration.

### Control unit types

HCD checks whether a specified control unit type is valid. Review your input for invalid control unit types. If applicable, correct the UNIT parameter of the CNTLUNIT statement.

If you do not want to change the type in your input data set, you can edit the HCD profile and specify how a control unit type in the IOCP input data set is mapped to a control unit type in the IODF. Specify one or more of the following keyword:

```
MAP_CUTYPE = xxxx,yyyy-yy
```

`xxxx` is the control unit type specified in the IOCP input data set

`yyyy-yy` is the control unit type and model (optional) to be used in the IODF

For more information about the HCD profile, refer to "Defining an HCD profile" on page 23.

### Control unit models

HCD requires, for certain control unit types, the specification of a model. For example, the IBM 3880 control unit requires a model specification. If a control unit type requires a model specification, and if you do not specify one in the input data set, HCD assigns a model to the control unit definitions based on the attached devices and the used control unit protocol. This control unit model is indicated as default model in the UIM (information message CBDA534I is issued). As processing goes on, it can be necessary to change the default model to another model to support the specified protocol (warning message CBDA536I is issued). Or, the default model is changed to attach a device type which is not supported by the default control unit model (warning message CBDA265I is issued).
Note: The sequence of messages is shown in reverse order in the migration log file since the messages are sorted according to decreasing severities. For an example, please see "Errors detected during assembly process" on page 303.

To assign a model to a control unit, change the UNIT parameter of the CNTLUNIT statement in the input data set. Append the model number separated by a dash to the control unit type specification. For example:

CNTLUNIT ...,UNIT=3880-23

To avoid changing the input data sets, you can also add a model number by using the MAP_CUTYPE parameter in the HCD profile as described under "Control unit types" on page 262.

Protocol support for control units

HCD checks the protocols supported by a control unit type. For example, in the source you may have an IBM 3745 with protocol S specified as control unit type 3705. The IOCP program does not check the protocol S specification. To be accepted by HCD as valid input, you have to change the control unit type to 3745.

Device types

For IOCP input data sets: HCD checks the device types for validity and that they can be attached to the specified control unit. Check your IODEVICE statements in the IOCP input data set and make sure that the device types are valid and reflect the true physical device they are defining. A common error, for example, is the IBM 3745 device type defined as 3705.

For MVSCP input data sets: HCD supports device types that previously had to be defined as "look-alike" devices for MVSCP. For example, for an IBM 3251 graphic device (previously defined as 2250-3), it is mandatory that the correct device type is defined in the UNIT parameter. HCD validates the PCU parameter, which is different for the IBM 3251 and the IBM 2250-3 graphic devices.

A device whose device type supports the dynamic capability may be defined as dynamic or not dynamic by means of the DYNAMIC parameter. However, there are programs, including customer programs, supplier programs and IBM products, that depend on device related data structures such as UCB and EDT, or use existing operating system services which access these data structures, and are unprepared to handle dynamic changes to these structures.

Therefore, HCD considers devices that are not specified with the DYNAMIC parameter in the IODEVICE statement as "installation-static", as it does if DYNAMIC=NO. That means, that the device might support the dynamic capability, but the installation requests that the device is not treated as dynamic.

Installation-static devices can be dynamically added to the software I/O configuration, but can not be deleted or modified while z/OS is running.

When migrating the MVSCP input data sets, HCD shows no value as default with the DYNAMIC and the LOCANY parameter, instead of specifying NO.

For HCPRI0 input data sets: HCD supports device types that previously had to be defined as "look-alike" devices for HCPRI0. For example, you can define an IBM 6262 printer device (previously defined as an IBM 4248 printer device in the HCPRI0 input data set) with a device type of 6262.
The device type support for VM has been brought into line with the MVS type support. However, there might be some differences to HCPRIO device type support (for example, concerning the MODEL parameter).

**Esoteric token**

HCD introduces an esoteric token that will be used by allocation to find the appropriate esoteric for a data set that has been cataloged using the esoteric. You no longer have to maintain a chronological order and may delete and add esoterics without getting access problems for data sets that are cataloged using esoterics.

You may use the HCD profile (see “Defining an HCD profile” on page 23) to tell HCD to assign a token in ascending order to each esoteric when migrating an MVSCP input data set.

If you do not want to assign tokens in ascending order or when you migrate only parts of a configuration using the incremental update function (see “Updating parts of a configuration by migrating input data sets” on page 292), you can use a new parameter on the UNITNAME statement. This parameter lets you specify a token for an esoteric to be migrated to HCD, as follows:

```
UNITNAME=...,TOKEN=n
```

`n` is a number from 1 to 8999.

You have to specify a token for all esoterics or for none at all. For more information on catalog considerations, refer to “Data sets cataloged with an esoteric device group name” on page 343.

**Assembler statements**

HCD generates own macro instructions into the logical input data set before processing. This may cause problems with assembler statements you inserted into your input data sets. For example, coding the ISEQ, CSECT, or RMODE instructions may cause a warning message issued by the assembler. As a consequence the migration will be terminated. To avoid this, remove your assembler statements. The generated HCD instructions look as follows:

```
PRINT OFF
COPY CBDZPARS
TITLE 'xxx LISTING'
CBDTXT CSECT
CBDTXT RMODE ANY
PRINT ON Nogen
.
input data set statements
.
PRINT OFF
HCDEND
END CBDTXT
```

**Preparing additional input data sets for migration**

This section describes what you have to consider when migrating more than one MVSCP, IOCP, or HCPRIO input data set to one IODF.

When you migrate additional input data sets to an IODF that already contains definitions, these input data sets may contain control units and devices that are already defined in the existing IODF. HCD assumes that control units being added that already exist in the IODF refer to the same physical control unit, and that the
The following sections describe the rules when a control unit or device is mapped to an existing one and when it is newly defined.

**Migrating additional IOCP input data sets**

When migrating additional IOCP input data sets, the mapping of control units and devices depends on whether the attached control units are already defined in the IODF or not.

**The same control unit number is already defined in the IODF:** The control unit is mapped, if both:
- The control unit type is the same, and
- The number and type of attached devices are the same

If one of these conditions is not fulfilled, the control unit definition is rejected.

**Figure 111 to Figure 113 on page 266** show examples of how control units are mapped. These examples do not show more than two control units attached to devices, but the same rules also apply if more control units are attached.

<table>
<thead>
<tr>
<th>IODF</th>
<th>IOCP Input Data Set</th>
<th>Resulting IODF</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>3990-3</td>
<td>3990-3</td>
<td>3990-3</td>
</tr>
<tr>
<td>05C0,32</td>
<td>05C0,32</td>
<td>05C0,32</td>
</tr>
<tr>
<td>3390</td>
<td>3390</td>
<td>3390</td>
</tr>
</tbody>
</table>

CNTUNIT CUNUMBR=100,UNIT=3990,...
IODEVICE ADDRESS=(5C0,32),UNIT=3390,
CUNUMBR=100

*Figure 111. IOCP Migration. Control units and devices are mapped, because they already exist in the IODF.*

<table>
<thead>
<tr>
<th>IODF</th>
<th>IOCP Input Data Set</th>
<th>Resulting IODF</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>3990-3</td>
<td>3990-3</td>
<td>3990-3</td>
</tr>
<tr>
<td>05C0,32</td>
<td>05C0,32</td>
<td>05C0,32</td>
</tr>
<tr>
<td>3390</td>
<td>3390</td>
<td>3390</td>
</tr>
</tbody>
</table>

CNTUNIT CUNUMBR=100,UNIT=3990,...
IODEVICE ADDRESS=(5C0,32),UNIT=3390,
CUNUMBR=100

*Figure 112. IOCP Migration. Control units and devices are mapped, although the devices also attached to another control unit in the IODF.*

The following example shows the same control units in the IODF and IOCP input data set, but the attached devices are shared in the IOCP input data set. In this case, the devices are merged and will be shared after the migration (only if the
control units do not connect to the same processor configuration).

control units do not connect to the same processor configuration).

The same control unit number is not yet defined in the IODF: In this case:
• A new control unit is defined.
• A new device is defined, unless
  – The device in the input data set is attached to a control unit, to which it is already attached in the IODF. In this case, the new device is mapped to the existing one and attached to both control units (see Figure 114).
  - or -
  – A device with same device number and type already exists in the IODF and is not attached to any control unit. In this case, the new device is mapped to the existing one and attached to the new control unit (see Figure 115 on page 267).

If none of these conditions is fulfilled, a new device is defined (see Figure 116 on page 267).

Figure 113. IOCP Migration. Control units and devices are mapped (only if the control units do not connect to the same processor configuration).

Figure 114. IOCP Migration. Devices are mapped, because one attached control unit is already defined in the IODF.
Migrating additional MVSCP or HCPRIO input data sets

A device can only be mapped if the device number and device type are the same. If the device number or type is not the same, a new device is defined.

If the device number and type are the same, HCD maps the device according to the following rules:

1. If you specify an associated processor and partition on the Migrate IOCP / MVSCP / HCPRIO Data panel or with the batch migration utility, HCD maps the new device to a device with the same device number and type connected to this processor and partition (provided that such a device exists).

2. If you do not specify an associated processor and partition or the new device does not attach to the specified processor and partition, HCD checks if the device is attached to a control unit. The device in the input data set is mapped, if
   - The device in the IODF is attached to the same control unit.
   - or -
   - The device in the IODF is not attached to any control unit.

This is illustrated in Figure 115. IOCP Migration. Devices are mapped, because the attached control unit is not yet defined in the IODF.

Figure 115. IOCP Migration. Devices are mapped, because the attached control unit is not yet defined in the IODF.

Figure 116. IOCP Migration. Devices are not mapped, because the control unit number does not exist in the IODF.

Migrating additional MVSCP or HCPRIO input data sets

A device can only be mapped if the device number and device type are the same. If the device number or type is not the same, a new device is defined.

If the device number and type are the same, HCD maps the device according to the following rules:

1. If you specify an associated processor and partition on the Migrate IOCP / MVSCP / HCPRIO Data panel or with the batch migration utility, HCD maps the new device to a device with the same device number and type connected to this processor and partition (provided that such a device exists).

2. If you do not specify an associated processor and partition or the new device does not attach to the specified processor and partition, HCD checks if the device is attached to a control unit. The device in the input data set is mapped, if
   - The device in the IODF is attached to the same control unit.
   - or -
   - The device in the IODF is not attached to any control unit.

This is illustrated in Figure 117 on page 268.
3. If the new device is not attached to a control unit, the device is mapped to the first device found with the same device number and type. If the IODF contains several devices with the same device number and type, the device of the MVSCP input data set can be erroneously mapped to a wrong device. To avoid this, you can specify the associated processor and partition on the Migrate IOCP / MVSCP / HCPRIO Data panel.

**Figure 117. MVSCP Migration.** Devices are mapped, because the devices are not attached to a control unit in the IODF.

**Figure 118. MVSCP Migration.** Devices are mapped to the first devices found with the same type and number.

If only a subset of devices in the input data set is already defined in the IODF, this subset is mapped to the existing devices and the remaining new definitions are added.

**Figure 119. MVSCP Migration.** 32 devices are mapped, the other 32 devices are newly defined.
Migrating input data sets using the HCD dialog

The following steps describe how to migrate IOCP, MVSCP, or HCPRIO input data sets to an IODF using the HCD dialog.

**Step 1: Specify the work IODF**

Before starting the migration, you require a work IODF. You can create a new work IODF or use an existing one.

1. On the Primary Task Selection panel enter the name of the IODF to which you want to migrate your input data sets.
2. Select *Migrate configuration data.*
   
   If you create a new work IODF, a panel appears on which you have to enter IODF specifications. (Refer to Figure 12 on page 37.)
3. On the following Migrate Configuration Data panel, select *Migrate IOCP/OS data.* The Migrate IOCP / MVSCP / HCPRIO Data panel shown in Figure 120 appears.

**Step 2: Migrate the input data sets**

1. Specify the identifier of the processor or operating system with which the input data sets will be associated.
   
   - For an IOCP input data set migration, specify a processor ID.
   - For an MVSCP or HCPRIO input data set migration, specify an OS configuration ID.

   If the specified processor or operating system does not exist in the IODF, a panel appears that allows you to define a new processor or operating system in the IODF.

   Migrating of a single channel subsystem (CSS) to an XMP processor is supported via the incremental migrate option (see Figure 120). Thus, you can consolidate multiple SMP processors on a single XMP processor using the migrate function. When migrating an SMP processor to an XMP processor, you must specify the target CSS. As default, CSS 0 is used.

2. Specify the input data set:
• If you are migrating a combined MVSCP/IOCP input data set, specify the Combined IOCP/MVSCP input data set field.
• If you are migrating an IOCP input data set only, specify the IOCP only input data set field.
• If you are migrating an MVSCP or HCPRIO input data set only, specify MVSCP only or HCPRIO input data set field.
• If you have separate IOCP and MVSCP (or IOCP and HCPRIO) input data sets, but want to migrate both into one IODF, specify both the IOCP only input data set and the MVSCP only or HCPRIO input data set field.

3. The specification of the Associated with processor and partition field is only applicable if you migrate MVSCP or HCPRIO input data sets to an IODF. Specify a processor and partition with which you want to associate definitions in the input data sets. HCD uses this information to map devices correctly if the IODF contains duplicate device numbers. For more information about this mapping, refer to “Migrating additional MVSCP or HCPRIO input data sets” on page 267.

4. Specify the processing mode:
   • **Validate** causes HCD to check the input and to inform you if errors are discovered. HCD does not store the input in the IODF, even if the input data set is free of errors. Only the new processor and/or operating system definitions that you defined are stored.
     The validate mode provides detailed messages how HCD treats control units and devices that already exist in the IODF.
   • **Save** causes HCD to check the input, and if free of errors, to store the data in the IODF. If there are errors in the input data set(s), HCD informs you by a message, and depending on the severity of the error, does not write the input to the IODF.

5. Specify the migrate option:
   - **complete**
     Select this option if you want to add a complete processor and/or OS configuration.
   - **incremental**
     If the specified processor or OS configuration already contains definitions, you can add and replace existing objects with the new information defined in the input data sets. Select this option for this partial migration on the panel from Figure 120 on page 269. Refer to “Updating parts of a configuration by migrating input data sets” on page 292 for more information about the partial migration.

**PCHIDs**

The migration task also allows updating the PCHIDs of a processor configuration in a validated work IODF with an IOCP input data set that has been generated by the CHPID Mapping Tool. HCD checks that the tokens in the IODF and in the IOCP input data set are matching. For more information on this process, see “How to interact with the CHPID Mapping Tool” on page 213.

6. If the CBDZPARS macro, which contains the migration parsing macros, is not in SYS1.MACLIB, specify the name of the library that contains it. If the library is not cataloged, specify the volume serial number.

After the input has been accepted, HCD issues a message informing you that the migration of input data sets is in process.
Step 3: Analyze errors and upgrade the input data set

During the migration process, HCD first invokes the assembler that parses the input statements. If it detects an error, migration is terminated.

- HCD writes a message to the terminal indicating that migration completed with return code RC=12.
- HCD writes a message to the HCD message list indicating that the assembler completed with a return code other than zero.
- The assembler writes information to the assembly listing that describes the problem in more detail.

If the assembler does not detect any errors, HCD runs a validation check. If it detects an error, HCD writes:

- A message to the terminal indicating that migration completed with a return code higher than 4.
- Error messages to the HCD message list describing the validation problem.

If the return code is 0 or 4, the IODF is updated and saved (if you specified to save the data). It is, however, recommended that you review the message log. HCD may have made assumptions that are contrary to your configuration requirements. Your actions:

1. Review the message list. For explanations and examples refer to “Resolving migration errors” on page 302.
2. Edit and correct the IOCP, MVSCP, and HCPRI0 input data sets.
3. Migrate your input data sets again.

Step 4: Update configuration data

If the protocol or the attached devices specified in the IOCP input data set do not match the supported control unit model, HCD may change the model definitions.

If HCD changes definitions, you are informed by messages. Review the messages, and follow the recommendation provided in the individual message.

If the type/model designated by HCD does not match the real type/model of the control unit, use the HCD dialog to specify the correct type/model.

Also, if this control unit is to be shared with another processor, update the IOCP input data set that is to be migrated accordingly.

Note: The configuration stored in an IODF may not match the IOCP/MVSCP or HCPRI0 input. If discrepancies occur, you can make corrections by using the dialog.

The following notes apply only if you do not use the extended migration function as described in “Changing I/O configurations by editing data sets” on page 273.

For coupling facility migration: After the IOCP input has been accepted, HCD issues messages informing you that the CF control unit and CF device definitions of the IOCP input data set were ignored.
Migrating input data sets using the batch utility

The following steps explain how to migrate your input data sets using the HCD batch utility instead of the HCD dialog. The batch utility is an easy way to migrate your input data sets if you are not familiar with the dialog yet. You have to use it for migrating your input data sets from MVS/XA SP 2.n or MVS/ESA SP 3.n.

**Step 1: Create the work IODF**

If an IODF does not yet exist, you first have to create a work IODF into which you want to migrate your MVSCP, IOCP, or HCPRIO data sets. Refer to “Initialize IODF” on page 309 on how to create and initialize an IODF.

**Step 2: Migrate input data set**

The HCD utility function for migration allows you to migrate the content of MVSCP, IOCP, and HCPRIO input data sets and to store the definitions into an IODF. For a detailed description of the utility function for migration, refer to “Migrate I/O configuration statements” on page 311.

**Step 3: Analyze errors and upgrade the input data set**

During the migration process, HCD first invokes the assembler that parses the input statements. If it detects an error, the migration process is terminated.

- HCD writes a message to the data set allocated by HCDMLOG that the migration completed with return code RC=12.
- HCD writes a message to the HCD migration log (HCDPRINT) indicating that the assembler completed with a return code other than zero.
- The assembler writes information to the assembly listing (HCDAASMP) that describes the problem in more detail.

If the assembler does not detect any errors, HCD runs a validation check. If it detects an error:

- HCD writes a message to the data set allocated by HCDMLOG that the migration completed with return code higher than 4.
- HCD writes error messages to the HCD migration log (HCDPRINT) describing the validation problem.

If the return code is 0 or 4, the IODF is updated. It is, however, recommended that you review the migration log. HCD may have made assumptions that are contrary to your configuration requirements. Your actions:

1. Review the migration log. For explanations and examples refer to “Errors detected during validation process” on page 304.
2. Edit and correct the IOCP, MVSCP, and HCPRIO input data sets.
3. Migrate your input data sets again.

**Step 4: Build production IODF**

Before you can use the IODF to IPL your operating system you have to convert the work IODF into a production IODF. Refer to “Build a Production IODF” on page 314 for an example on how to build a production IODF using the work IODF.
Changing I/O configurations by editing data sets

With z/OS HCD the extended migration function and the added possibilities for writing and migrating configurations allow defining or changing configuration definitions without using the HCD ISPF dialog front-end.

Using I/O configuration statements with IOCP/MVSCP syntax, you can also define all configuration objects with their attributes and their connections. Type these statements into a data set as input to the migrate function. For example, a switch together with its ports can be defined via this method.

It is also possible to recreate data sets containing I/O configuration statements for the processor, operating system and switch configurations from an IODF.

However, for some tasks, such as deleting certain configuration objects, it is necessary or easier to make the configuration changes directly in the HCD dialog. For more details on the capabilities of the migration function refer to the table in Table 7 on page 292.

Processor configurations
The migrate IOCP function is extended to allow the specification of parameters, additional to IOCP, in the input data set. Analogously, when building an IOCP input data set from the IODF, information is generated which describes the additional parameters of the configuration objects.
Processor configuration data sets can be built using the Build IOCP Input Data Set function or the new CONFIG PR batch utility. For details, refer to "Build an IOCP input data set" on page 208 and to "Build I/O configuration data" on page 320.

Operating system configurations
The migrate MVSCP function is extended to allow for the specification of all configuration data of an OS configuration, for example the device preference values for esoterics or the user parameters for devices. Also, it is possible to generate an OS configuration data set from the IODF using the Build OS Configuration Data Set dialog or the CONFIG OS batch utility. For details, refer to "Build I/O configuration data" on page 215 and to "Build I/O configuration data" on page 320.

Switch configurations and switch-to-switch connections
It is possible to migrate switch definitions with all ports, switch-to-switch connections and all switch configurations from a data set. Analogously, it is possible to build such a data set from the IODF using the CONFIG SW batch utility. For details, refer to "Build I/O configuration data" on page 320.

Chapter 11. How to migrate existing input data sets 273
**Important Note:**

It may not be possible to remigrate an IOCP input data set generated by HCD back into the IODF. The reasons are:

- HCD uses the High Level Assembler program for parsing the IOCP statements. The High Level Assembler earlier than V1.5 is restricted to 255 characters for any keyword value. IOCP statements, however, may contain keywords with a value greater than 255 characters. High Level Assembler V1.5 removes this restriction.

- HCD keeps additional data for a processor configuration that is not contained in an IOCP input data set. This data may be used for validation and, therefore, missing at the migrate step leading to validation errors. For example, the partition usage is defaulted to CF/OS. For a shared CF peer channel, this may lead to a validation error, because only a CF partition may be specified in the access or candidate list.

- Since the IOCP data are only a subset of the processor configuration data, you may lose this additional configuration data if you update a processor configuration from an IOCP input data set.

For updating the IODF via I/O control statements, it is recommended to use the extended I/O configuration statements of HCD instead of an IOCP input data set (see “IOCP input data sets using extended migration” on page 210).

---

### Additional parameters and statements

In the following, the configuration objects and their attributes you can define via I/O control statements and migrate into HCD are described. For a detailed description of the IOCP keywords and parameters, refer to the **IOCP User's Guide** for your processor.

<table>
<thead>
<tr>
<th>Operating system</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Name</td>
</tr>
<tr>
<td></td>
<td>Type</td>
</tr>
<tr>
<td></td>
<td>Description</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NIP console</th>
<th>Device number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Order</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EDT</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Preference value</td>
</tr>
<tr>
<td></td>
<td>Description</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Esoteric</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Device numbers</td>
</tr>
<tr>
<td></td>
<td>Token</td>
</tr>
<tr>
<td></td>
<td>VIO indication</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Generic</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Preference value</td>
</tr>
<tr>
<td></td>
<td>VIO indication</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Switch</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Unit</td>
</tr>
<tr>
<td></td>
<td>Model</td>
</tr>
<tr>
<td></td>
<td>Description</td>
</tr>
<tr>
<td></td>
<td>Serial number</td>
</tr>
<tr>
<td></td>
<td>Installed ports</td>
</tr>
<tr>
<td></td>
<td>Chained switch connection</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>Port</th>
<th>Address</th>
<th>Name</th>
<th>Occupied indication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch configuration</td>
<td>ID</td>
<td>Switch ID</td>
<td>Description</td>
</tr>
<tr>
<td>Port configuration</td>
<td>ID</td>
<td>Allowed dynamic connections</td>
<td>Prohibited dynamic connections</td>
</tr>
<tr>
<td>Processor</td>
<td>ID</td>
<td>Unit</td>
<td>Model</td>
</tr>
<tr>
<td>Channel Subsystem</td>
<td>ID</td>
<td>Description</td>
<td>Maximum number of devices</td>
</tr>
<tr>
<td>Partition</td>
<td>Name</td>
<td>Number</td>
<td>Usage</td>
</tr>
<tr>
<td>Channel path</td>
<td>ID</td>
<td>Type</td>
<td>Operation mode</td>
</tr>
<tr>
<td>Control unit</td>
<td>Number</td>
<td>Unit</td>
<td>Model</td>
</tr>
</tbody>
</table>
Device | Number
---|---
Unit
Model
Serial number
Volume serial number
Connected control units
Description
Unit address
Preferred channel path
TIMEOUT indication
Status detection indication
OS parameters/features
Subchannel set ID

**Operating system**
An operating system is specified with the IOCONFIG statement. It contains the following parameters:

**ID** Specifies the numerical identifier of the OS (mandatory).

This keyword is maintained for compatibility with the MVSCP syntax.

**TYPE** Specifies the OS configuration type. This is required only, if the migration is performed using the wildcard * in the parameter string of the batch utility. Otherwise, it is ignored and the OS configuration type is taken from the invocation parameters.

**NAME** Specifies the OS configuration ID. This is required only, if the migration is performed using the wildcard * in the parameter string of the batch utility. Otherwise, it is ignored and the OS configuration ID is taken from the invocation parameters.

**DESC** Specifies a description of the operating system (optional). The description of the OS configuration is added or updated.

**Syntax:**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID=id</td>
<td>2 alphanumeric characters</td>
</tr>
<tr>
<td>NAME=os_name</td>
<td>up to 8 alphanumeric characters</td>
</tr>
<tr>
<td>TYPE=type</td>
<td>MVS or VM</td>
</tr>
<tr>
<td>DESC='description'</td>
<td>up to 32 characters</td>
</tr>
</tbody>
</table>

**Example:** The following example defines an OS configuration named NEWOS01B of type MVS together with the given description.

IOCONFIG ID=01,NAME=NEWOS01B,DESC='MVS 5.1 LPAR system',TYPE=MVS

**NIP consoles**
A NIP console is specified with the NIPCON statement. It contains the DEVNUM keyword.

**DEVNUM** Specifies a list of device numbers to be used as NIP consoles (mandatory). All devices specified must be defined in your configuration.

NIP tries to use the devices in the order they are listed (left-to-right).
Syntax:

DEVNUM=(device_number[,device_number]...) list of device numbers

Example: In the example the devices with numbers 102E and 102F are assigned to be used as consoles. NIP will try device 102E first.

NIPCON DEVNUM=(102E,102F)

EDT

An Eligible Device Table is specified with the EDT statement.

Note: If you want to define only one EDT, place the EDT statement ahead of all UNITNAME statements (see "Esoteric" and "Generic" on page 278). If multiple EDT statements are written, each statement must precede the UNITNAME statements defining the corresponding EDT.

The EDT statement contains the following parameters:

ID Specifies the identifier of the EDT. The default is the ID specified on the IOCONFIG statement. If multiple EDT statements are written, the ID parameter is mandatory for all but one statement.

DEVPRF Specifies a list of devices in the order of preference (optional).

This parameter conforms to legacy syntax but is still supported. However, it is recommended to use the DEVPRF parameter with the UNITNAME statement, see "Generic" on page 278, instead.

DESC Specifies a description of the EDT (optional).

Syntax:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID=id</td>
<td>2 hexadecimal characters</td>
</tr>
<tr>
<td>DEVPRF=(generic_name[,generic_name]...)</td>
<td>list of generic device types, generates the preference value of generics according to the list position</td>
</tr>
<tr>
<td>DESC='description'</td>
<td>up to 32 characters</td>
</tr>
</tbody>
</table>

Example: The following example defines EDT 01 with the given description.

EDT ID=01,DESC='Eligible Device Table 1'

Esoteric

An esoteric is specified with the UNITNAME statement. For use with esoterics it contains the following parameters:

NAME Specifies the name of the esoteric (mandatory).

Note: Do not use esoteric names SYSALLDA, SYS3480R, or SYS348XR.

UNIT Specifies a sequence of consecutive device numbers (optional). The numbers specified must be defined in the configuration.

TOKEN Allows controlling the order of esoterics in the EDT (optional). Only relevant if you have data sets that are cataloged using esoterics.
Tokens prevent the order of esoterics from becoming alphabetical after IPL, thus avoiding access problems for data sets that are cataloged using esoterics.

If a token is specified for one esoteric you must also specify tokens for all other esoterics.

**VIO**

States whether or not the devices are eligible for VIO. May only be set to YES if the esoteric contains at least one DASD device type (optional).

**Syntax:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>8 alphanumeric characters</td>
</tr>
<tr>
<td>UNIT</td>
<td>4 hexadecimal characters followed by a decimal number</td>
</tr>
<tr>
<td>TOKEN</td>
<td>number in the range 1 to 8999</td>
</tr>
<tr>
<td>VIO</td>
<td>YES or NO</td>
</tr>
</tbody>
</table>

**Example:** The following example assigns 8 device numbers 01D1 through 01D8 and the token value to esoteric device group ES002. The group is not eligible for VIO.

```
UNITNAME NAME=ES002,
UNIT=((01D1,8)), *
TOKEN=2015, *
VIO=NO
```

**Generic**

A generic is specified with the UNITNAME statement. For use with generics it contains the following parameters:

**NAME** Specifies the name of the generic (mandatory; must be a valid generic name).

**VIO** Specifies whether or not the devices are eligible for VIO (optional). VIO=YES may only be specified if the generic name specifies a DASD device type.

**DEVPREF** Allocates a position in a preference order of generics (optional).

For default values see the information on MVS devices within the "Supported Hardware Report" on page 401. This parameter value must be unique for the OS configuration.

**Syntax:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>8 alphanumeric characters</td>
</tr>
<tr>
<td>VIO</td>
<td>YES or NO</td>
</tr>
<tr>
<td>DEVPREF</td>
<td>decimal number between 1 and 99999</td>
</tr>
</tbody>
</table>

**Example:** The following example defines the generic device group 3390 with the preference value 150. The devices are eligible for VIO.

```
UNITNAME NAME=3390,
VIO=YES, *
DEVPREF=150
```
Switch
A switch is specified with the SWITCH statement. It contains the following parameters:

- **SWID**: Specifies an identifier for the switch (mandatory).
- **UNIT**: Specifies the switch unit (mandatory).
- **MODEL**: Specifies the switch model (optional).
- **DESC**: Specifies a description of the switch (optional).
- **SERIAL**: Specifies a serial number (optional).
- **PORT**: Specifies the installed ports (default from UIM, optional).
  
  Specify only ports that are supported by the switch type. At least the minimum installed port range is set to installed. If the switch already exists with an installed port range that differs from the specified installed port range(s), only the new specified ports are set to installed if possible. That means, any existing installed port that does not hold a connection to a channel path or control unit is set to not installed if not specified with the PORT parameter.

- **SWPORT**: Specifies the chained switch connections (optional). If operand exceeds 255 characters, repeat the SWITCH statement with the remaining values.
  
  If the switch already exists, all existing connections to other switches are broken. The connections to other switches are established as specified by the SWPORT parameter.

- **ADDRESS**: Specifies the switch address for a FICON switch (optional).

  The corresponding switch control unit and device are specified by corresponding CNTLUNIT and IODEVICE statements.

**Syntax:**

- **SWID=id**: 2 hexadecimal characters
- **UNIT=switch_unit**: like CU type
- **MODEL=switch_model**: like CU model
- **DESC='description'**: up to 32 characters
- **SERIAL=serial_no**: up to 10 characters
- **PORT=((low_port_id,high_port_id),...)**: up to 32 port ranges
- **SWPORT=((from_port,to_switch,to_port),...)**: up to 32 switch-to-switch connections

**Example:** In the following example, switch 02 with an installed port range 80 to FB is chained to port D1 of switch 01 via port C0.

```
SWITCH SWID=02,UNIT=9032,MODEL=3, *
    PORT=((80,FB)), *
    DESC='Aspen switch, installed 10/09/94', *
    SERIAL=55-8888, *
    SWPORT=(C0,01,D1)
```

**Note:** Put the serial numbers in quotes, if you use characters such as blanks or commas as part of your serial numbers.
**Port**

A switch port is specified with the PORT statement. It contains the following keywords:

- **ID** Port address identifying the port (mandatory).
- **NAME** Specifies a port name (optional).

To be accepted within I/O Operations (ESCON Manager) commands the port name must not include commas, asterisks, or blanks. It must not contain X'FF' or any extended binary-coded decimal interchange code (EBCDIC) character less than X'40'. It must also not begin with a left parenthesis and end with a right parenthesis.

- **OCC** Indicates that the port has an external connection (optional).

  *External* means a connection to a processor, switch, or control unit.

**Syntax:**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>port_id (2 hexadecimal characters)</td>
</tr>
<tr>
<td>PORTNM</td>
<td>portname (up to 24 characters)</td>
</tr>
<tr>
<td>OCC</td>
<td>no value assigned</td>
</tr>
</tbody>
</table>

**Example:** In the following example port D5 is named 'connected_to_CU_7230' and indicated as occupied.

```
PORT ID=D5, *
PORTNM='connected_to_CU_7230', *
OCC
```

**Switch configuration**

A switch configuration is specified with the SWCONF statement. It contains the following parameters:

- **ID** Specifies the switch configuration ID (mandatory).
- **SWID** Identifies the switch owning the configuration (mandatory).
- **DESC** Specifies a description of the switch configuration (optional).
- **DEFCONN** Specifies whether the default port connections are set to allowed or prohibited (mandatory).

**Syntax:**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>switch_configuration_id (8 characters)</td>
</tr>
<tr>
<td>SWID</td>
<td>switch_id (2 hexadecimal characters)</td>
</tr>
<tr>
<td>DESC</td>
<td>description (up to 32 characters)</td>
</tr>
<tr>
<td>DEFCONN</td>
<td>dynamic_default_connection (A or P)</td>
</tr>
</tbody>
</table>

**Example:** In the following example the default connection for switch configuration BASECONF of switch 01 is set to allowed.

```
SWCONF ID=BASECONF, *
  SWID=01, *
  DESC='basic configuration', *
  DEFCONN=A
```

**Port configuration**

A port configuration is specified with the POCONF statement.
The POCONF statement is an optional extension to the switch configuration. With POCONF, port connections are defined explicitly and existing defaults are overridden. In all, the statement may be a maximum of 255 characters long. To express longer statements the POCONF may be repeated. POCONF includes the following parameters:

**ID**

Port address identifying the port (mandatory).

**PORTCF**

Specifies the type of connections to target ports (mandatory).

In the PORTCF operand:

- **A** sets the dynamic connection to the succeeding list of target port IDs to allowed
- **P** sets the dynamic connection to the succeeding list of target port IDs to prohibited
- **D** sets a dedicated connection to the succeeding target port ID
- **BLOCKED** blocks the port

**Syntax:**

\[
\text{ID=port\_id} \\
\text{2 hexadecimal characters}
\]

\[
\text{PORTCF=\{[A,(id1,...,idn)],} \\
\text{\[P,(id1,...,idm)],} \\
\text{\[D,(id),\]} \\
\text{\[\text{BLOCKED}]\}}
\]

One or more of the following specifications:

- list of ports in installed range (A, connection allowed)
- list of ports in installed range (P, connection prohibited)
- port with dedicated connection (D)
- BLOCKED attribute

**Example:**

In the following example port D2 has allowed dynamic connections to ports B1 and B3, and prohibited dynamic connections to B5 and B7. Port D4 has a dedicated connection to port C0.

POCONF ID=D2, * 
POCONF PORTCF=(A,(B1,B3),P,(B5,B7)) 
POCONF ID=D4, * 
POCONF PORTCF=(D,(C0))

**Processor**

A processor is specified with the ID statement. If specified, it must precede all other statements in the configuration data set. It contains the following keywords:

**NAME**

Specifies the processor ID. If the ID does not exist, it is created (1).

**MSG1**

Specifies the identification information that is printed on the ID1 line of the heading in IOCP configuration reports (optional). Only supported for compatibility with IOCP.

**MSG2**

Specifies the identification information that is printed on the ID2 line of the heading in IOCP configuration reports (optional). Only supported for compatibility with IOCP.

**SYSTEM**

Specifies the machine limits and rules that IOCP will enforce for a deck verification. The system parameter includes a specification of the processor machine type number and, optionally, a machine limits number (optional). Only supported for compatibility with IOCP.

**LSYSTEM**

Specifies the system name (CPC designator) of the local system; that is, the system which uses this IOCDS.
UNIT Specifies the processor unit (1).
MODEL Specifies the processor model (1).
LEVEL Specifies the processor support level (1). For further information on support levels refer to "support_level_ID" on page 313.
MODE Specifies the processor configuration mode as LPAR or BASIC (1).
SNAADDR Specifies the SNA address (network name, system name) for a processor in an S/390 microprocessor cluster (optional).
SERIAL Specifies the processor serial number (optional).
DESC Specifies a description for the processor (optional).

Note (1): UNIT, MODEL and LEVEL are processed only if the migration is performed using the wildcard * in the parameter string of the batch utility.

Syntax:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>processor_id 8 characters</td>
</tr>
<tr>
<td>MSG1</td>
<td>message up to 64 characters; first 8 characters are taken as IOCDS name</td>
</tr>
<tr>
<td>MSG2</td>
<td>message up to 64 characters</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>(processor_unit[,limits_number]) 4 characters followed by a decimal number</td>
</tr>
<tr>
<td>UNIT</td>
<td>processor_unit 8 characters</td>
</tr>
<tr>
<td>MODEL</td>
<td>processor_model 4 characters</td>
</tr>
<tr>
<td>LEVEL</td>
<td>support_level 8 characters</td>
</tr>
<tr>
<td>LSYSTEM</td>
<td>local_cpc_designator 8 characters</td>
</tr>
<tr>
<td>SNAADDR</td>
<td>(network_name,system_name) list of 2 entries, each up to 8 characters</td>
</tr>
<tr>
<td>MODE</td>
<td>processor_mode BASIC or LPAR</td>
</tr>
<tr>
<td>SERIAL</td>
<td>serial_number up to 10 characters</td>
</tr>
<tr>
<td>DESC</td>
<td>description up to 32 characters</td>
</tr>
</tbody>
</table>

Example: In the following example processor PROC01 of type 2094, model S28 is defined with the serial number 0518712094, in LPAR mode, support level H050331.

```
ID NAME=PROC01,UNIT=2094,MODEL=S28, *
       DESC='XMP, Basic 2094 support',SERIAL=0518712094, *
       MODE=LPAR,LEVEL=H050331
```

Channel subsystem and partition

Depending on whether you use the RESOURCE statement for XMP processors or SMP processors, it specifies the channel subsystems, the logical partitions (names and numbers) and groups the logical partitions to the channel subsystems. It contains the following keywords:

PART or PARTITION Specifies a list of partition names with an optional addition of the corresponding partition numbers (mandatory).

DESC Specifies a list containing descriptions for the defined partitions (optional).
USAGE Specifies a list describing the partition usage type for each partition.

MAXDEV Is only allowed for XMP processors and specifies for each channel subsystem the maximum number of devices, including those defined in the IOCDS, to be allowed using dynamic I/O. For z9 EC processors and later models, it specifies the maximum number of devices in each subchannel set.

CSSDESCL specifies a list of channel subsystem descriptions, one list entry for each channel subsystem listed in the MAXDEV keyword.

Note: If the partition type is missing, it is set automatically depending on which types of channel paths are assigned to the partition and the capability of the processor.

Syntax:

For XMP processors:

PARTITION =
((CSS(0),(lname[,lnumber])[,,(lname[,lnumber])...])
...,
(CSS(n),(lname[,lnumber])[,,(lname[,lnumber])...]))

lpname: up to 8 alphanumeric characters for the LPAR name; for XMP processors, an * is accepted as lpname to indicate a reserved partition;

lnumber: 1 hexadecimal character for the LPAR number.
The CSS(n) parameter(s) must be used for XMP processors and must not be used for SMP processors.

DESCL=('descp1_css0','descp2_css0','...,
'descp1_css1','descp2_css1','...,
...,'descp1_cssn','descp2_cssn','...')

description for all partitions in the processor complex, up to 32 characters per description.

CSSDESCL=('desc_css0','desc_css1','...')
description syntax of channel subsystems for XMP processors

USAGE=(usage1_css0,usage2_css0,...,usage1_css1,usage2_css1,...,
...usage1_cssn,usage2_cssn,...)

usage of each partition in the processor complex (CF, OS, or CF/OS)

MAXDEV=((CSS(0),maxnum1[,maxnum2]),...,(CSS(n),maxnum1[,maxnum2]))

maximum number of devices for each channel subsystem and each subchannel set

Example for an SMP processor: In the following example, partitions LP1, LP2, and CF1 are defined and described.

RESOURCE PART=((LP1,1),(LP2,2),(CF1,3)), *
DESCL=('Logical Partition 1', *
'System 3', *
'Coupling Facility'), *
USAGE=(CF/OS,OS,CF)

Example for an XMP processor: In the following example, the XMP processor contains three channel subsystems with three partitions in each. Channel subsystem CSS(2) contains two subchannel sets. This is indicated by the MAXDEV statement containing two maximum numbers of devices (35 and 20).

RESOURCE PART=((CSS(0),LP01,1),(LP02,2),(LP03,3)), *
(CSS(1),LP11,1),(LP12,2),(LP13,3)), *
(CSS(2),LP21,1),(LP22,2),(LP23,3)), *
DESCL=('LP01_of_CSS0','LP02_of_CSS0','LP03_of_CSS0', *
'LP01_of_CSS1','LP02_of_CSS1','LP03_of_CSS1', *
'LP01_of_CSS2','LP02_of_CSS2','LP03_of_CSS2'), *

'LP11_of_CSS0','LP12_of_CSS0','LP13_of_CSS0', *
'LP11_of_CSS1','LP12_of_CSS1','LP13_of_CSS1', *
'LP11_of_CSS2','LP12_of_CSS2','LP13_of_CSS2'), *

'LP21_of_CSS0','LP22_of_CSS0','LP23_of_CSS0', *
'LP21_of_CSS1','LP22_of_CSS1','LP23_of_CSS1', *
'LP21_of_CSS2','LP22_of_CSS2','LP23_of_CSS2'), *

'LP31_of_CSS0','LP32_of_CSS0','LP33_of_CSS0', *
'LP31_of_CSS1','LP32_of_CSS1','LP33_of_CSS1', *
'LP31_of_CSS2','LP32_of_CSS2','LP33_of_CSS2'), *

Chapter 11. How to migrate existing input data sets 283
For HCD, it is also possible to specify a separate RESOURCE statement for each channel subsystem. This may be required if you do not use a High Level Assembler V1.5 or later. You can split the previous RESOURCE example for an XMP processor into the following parts:

```
RESOURCE PART=((CSS(0),(LP01,1),(LP02,2),(LP03,3))),
  DESCL=('LPAR1_of_CSS0','LPAR2_of_CSS0','LPAR3_of_CSS0'),
  USAGE=(CF/OS,OS,CF),
  MAXDEV=(CSS(0),63),
  CSSDESCL=('first CSS(0)')
RESOURCE PART=((CSS(1),(LP11,1),(LP12,2),(LP13,3))),
  DESCL=('LPAR1_of_CSS1','LPAR2_of_CSS1','LPAR3_of_CSS1'),
  USAGE=(CF/OS,OS,CF),
  MAXDEV=(CSS(1),50),
  CSSDESCL=('second CSS(1)')
RESOURCE PART=((CSS(2),(LP21,1),(LP22,2),(LP23,3))),
  DESCL=('LPAR1_of_CSS2','LPAR2_of_CSS2','LPAR3_of_CSS2'),
  USAGE=(CF/OS,OS,CF),
  MAXDEV=(CSS(2),35,20),
  CSSDESCL=('third CSS(2)')
```

**Channel path**

A channel path is specified with the CHPID statement. For an XMP processor, a spanned CHPID is defined for one or more channel subsystems. Therefore, the CHPID statement must contain channel subsystem data. If only one CSS is defined, it is not required to specify it on the PATH keyword.

The CHPID statement contains the following keywords:

- **PATH** Specifies the CHPID number and, if required, the CSS IDs of the channel path (mandatory).
- **TYPE** Specifies the channel path type of I/O operation for the channel path (mandatory).
- **SHARED** Specifies that the channel path on the CHPID statement is shared (optional).
- **REC** Specifies that the channel path on the CHPID statement is reconfigurable (optional).
- **PART, PARTITION, or NOTPART**
  - **PART** and **PARTITION** specify the access list of logical partitions that will have the channel path configured online after POR, and the candidate list identifying the logical partitions which can access the device.
  - **NOTPART** specifies the access list of logical partitions that will not have the channel path configured online after POR, and the list of logical partitions which cannot access the device.
  - **REC** REC in the **PART** or **PARTITION** keyword allows the channel path to be dynamically moved between partitions after POR.
- **CHPARM,OS** Specifies channel path data that is used by the operating system. For example, CHPARM=01 indicates that the channel path is managed by DCM. Or, CHPARM=40 indicates that the maximum frame size for an IQD channel is 24K.
I/O CLUSTER  Specifies an I/O cluster name. An I/O cluster is a sysplex that
owns a managed channel path for an LPAR processor
configuration.

SWITCH  Specifies a number for a switch (ESCON Director) which is used as
a dynamic switch for all paths from the channel path (CHPID) to
the connected control units (required for dynamic connections
through a switch).

DESC  Specifies a description of the channel path (optional).

TPATH  Specifies a connected CF channel path (optional).

The TPATH parameter can be specified for a CF channel path,
either CF receiver, CF sender, or CF peer channel path.

For a CF sender or CF receiver channel path, the TPATH parameter
must contain:
• the target processor
• the target CSS ID for XMP processors
• the target channel path ID

When specified with a CF sender channel path id, the TPATH
parameter must also contain:
• the CF sender control unit and device numbers used for the CF
  connection. This information is optional for a CF receiver
  channel path.

For a CF peer channel path, the TPATH parameter contains the
following items for the target and source channel paths of the CF
connection:
• the target processor
• the target CSS ID for XMP processors
• target channel path ID
• control unit number (only if this is a sending CF channel, that is,
  the channel path connects to a target CF logical partition)
• starting device number (only if this is a sending CF channel)

A CF connection uses two (CF sender channel) or seven (sending
CF peer channel) devices. Only the starting one can be specified
(with four digits). The remaining devices are automatically
assigned to the next consecutive device numbers.

Any CF control units and CF devices specified via CNTLUNIT and
IODEVICE statements are ignored.

The connection can only be established if the target channel path
exists. If the target channel path is already connected, the existing
connection is broken and a new connection is established.

SWPORT  Specifies an entry switch port (optional).

PCHID  Specifies a physical channel identifier (optional).

AID  Specifies the ID of the host communication adapter (HCA) on
which the channel is defined.

PORT  Specifies the port on the HCA on which the channel is defined.

Syntax:

PATH=[(CSS(n,...),]chpid_number[]]  2 hexadecimal characters
TYPE=type
valid channel path type

SHARE
no value assigned

REC
no value assigned

For XMP processors:
PART[ITION]=((CSS(0),(acc_lp1,...,acc_lpn)
 [,(cand_lp1,...,cand_lp2)][,REC])
 ... 
[(CSS(n),(acc_lp1,...,acc_lpn)
 [,(cand_lp1,...,cand_lp2)][,REC]])
}

For SMP processors:
PART[ITION]=((acc_lp1,...,acc_lpn)
[,(cand_lp1,...,cand_lp2)][,REC]
)

For XMP processors:
NOTPART=((CSS(0),(acc_lp1,...,acc_lpn)
 [,(cand_lp1,...,cand_lp2)])
 ... 
[(CSS(n),(acc_lp1,...,acc_lpn)
 [,(cand_lp1,...,cand_lp2)])
 )
For SMP processors:
NOTPART=((acc_lp1,...,acc_lpn)
[,(cand_lp1,...,cand_lp2)]
)

OS=xx, CHPARM=xx
2 hex character OS parameter

IOCLUSTER=sysplex
8 character sysplex name for managed CHPID

SWITCH=xx
2 hexadecimal characters

DESC='description'
up to 32 characters

TPATH=((proc,chpid[,[CFS CU,CFS device]})
(target CHPID for
(for legacy CF channel path only)
connected CHPID pairs
(CF connection):
8 character processor name
2 hex character CHPID
4 hex character CU number
4 hex character device number

TPATH=((proc,chpid[[CFP CU,CFP device]],
 (proc,chpid[[CFP CU,CFP device]])
 (pair of target and source CHPIDs
 (for CF peer channel path only)
CF peer connection):
8 character processor name
2 hex character CHPID
4 hex character CU number
4 hex character device number

TPATH=((CSS(n),proc,chpid[[cu,device]],...)
for XMP processors

SWPORT=(swid,port)
switch and port to
which the CHPID connects

PCHID=xxx
three hexadecimal characters for the physical channel ID

AID=xx
two hexadecimal characters

PORT=n
one numeric character

Example for an SMP processor: In the following example channel path 35 of type
CNC is connected to port FB of switch 03, which is used as a dynamic switch.
Channel path 35 is defined as shared with all partitions of the processor in its
access list.
Channel path 10 of type CFS, defined to partition LP4, is connected to CFR channel 11 of processor PROC1 using control unit FFFE and devices FFF0,2 for the coupling facility connection.

Channel path 12 of type CFR, defined as dedicated to partition C1, is connected to CF sender channel 70 of processor PROC2. (The used coupling facility devices and control units are defined with CHPID 70 of processor PROC2.)

```
CHPID PATH=(35),SWITCH=03,TYPE=CNC, *
   SWPORT=(03,FB),SHARED, *
   DESC='Chpid connected to switch'
CHPID PATH=(10),TYPE=CFS,PART=((LP4),(LP4)), *
   TPATH=(PROC1,11,FFFE,FFF0) *
CHPID PATH=(12),TYPE=CFR,PART=((C1),(C1)), *
   TPATH=(PROC2,70)
```

Example for an XMP processor: In the following example, the spanned channel path 33 of type IQD is shared by partitions from channel subsystems 0 and 1.

```
CHPID PATH=(CSS(0,1),33), *
   TYPE=IQD, *
   PART=((CSS(0),(LP01,LP02)), *
      (CSS(1),(LP11,LP12)))
```

Control unit

A control unit is specified with the CNTLUNIT statement. It contains the following keywords:

- **CNUMBR** Specifies a number assigned to the control unit (mandatory). The number assigned to each control unit must be unique within an IODF.
- **UNIT** Specifies the type of control unit (mandatory).
- **SERIAL** Specifies a serial number (optional).
- **SWPORT** Specifies switch ports to which the control unit is connected (optional). If operand exceeds 255 characters, repeat the CNTLUNIT statement with the remaining values.
- **DESC** Specifies a description of the control unit (optional).
- **PATH** For each channel subsystem, this keyword specifies the channel paths attached to the control unit (mandatory).
- **LINK** For each channel subsystem, this keyword specifies the link address to which the control unit is attached (optional). The order in which the link addresses are specified corresponds to the order in which the channel paths are specified in the PATH keyword.
- **UNITADD** Specifies the unit address ranges that are recognized by the control unit (mandatory).
- **CUADD** Specifies the logical address for the control unit (optional).
- **SHARED** Specifies the level of concurrency of I/O requests that the parallel channel path allows for the control unit (optional).
- **PROTOCOL** Specifies the interface protocol that the parallel control unit uses when operating with the channel paths specified in the PATH keyword (optional).
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUNUMBR</td>
<td>4 hexadecimal characters</td>
</tr>
<tr>
<td>UNIT</td>
<td>valid control unit type</td>
</tr>
<tr>
<td>SERIAL</td>
<td>up to 10 characters</td>
</tr>
<tr>
<td>SWPORT</td>
<td>list of up to 32 sublists (switch ID, port ID)</td>
</tr>
<tr>
<td>DESC</td>
<td>up to 32 characters max.</td>
</tr>
<tr>
<td>For XMP processors</td>
<td>PATH=((CSS(0),chpid[,chpid,]...)</td>
</tr>
<tr>
<td></td>
<td>[,CSS(1),chpid[,chpid,]...])</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>[,CSS(n),chpid[,chpid,]...])</td>
</tr>
<tr>
<td>For SMP processors</td>
<td>PATH=(chpid[,chpid]...)</td>
</tr>
<tr>
<td>LINK</td>
<td>0,1,... for the CSS ID; 2 hexadecimal characters for each static CHPID 'res' for each managed CHPID</td>
</tr>
<tr>
<td>UNITADD</td>
<td>2 hexadecimal characters for each unit address followed by a decimal number</td>
</tr>
<tr>
<td>CUADD</td>
<td>1 or 2 hexadecimal characters</td>
</tr>
<tr>
<td>SHARED</td>
<td>Y or N</td>
</tr>
<tr>
<td>PROTOCOL</td>
<td>D, S, or S4</td>
</tr>
</tbody>
</table>

**Example for an SMP processor:** In the following example, control unit 0CC0 of type 3995-151 is connected to channel path 20 using link address E4. The control unit is connected to port E4 of switch AA.

```plaintext
CNTLUNIT CUNUMBR=0CC0,PATH=(20),UNITADD=((00,256)),LINK=(E4), *  
  UNIT=3995-151,SWPORT=((AA,E4)), *  
  SERIAL=0123456789,DESC='Building 12'
```

**Note:** Put the serial numbers in quotes, if you use characters such as blanks or commas as part of your serial numbers.

**Example for an XMP processor:** In the following example, control unit 0780 of type 2105 is connected to channels 11 and 14 in channel subsystem 0 and to channels 21 and 24 in channel subsystem 1. All channels in both channel subsystems use link address E8. Channels from CSS 0 are connected to the control unit via switch 01, channels from CSS are connected via switch 02.

```plaintext
CNTLUNIT CUNUMBR=0780,PATH=((CSS(0),11,14),(CSS(1),21,24)), *  
  UNITADD=((00,128)),LINK=((CSS(0),E8,E8),(CSS(1),E8,E8)),*  
  CUADD=I,UNIT=2105  
  DESC='ESS12 780 CU F (3390-mix) VSE128'  
  SWPORT=((01,E8),(02,E8))
```

**Device**

A device is specified with the IODEVICE statement. It contains the following keywords:

- **ADDRESS** Specifies the device number and how many devices are to be defined (mandatory).
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT</td>
<td>Specifies the device type (mandatory).</td>
</tr>
<tr>
<td>MODEL</td>
<td>Specifies the model number of the device, if available (optional).</td>
</tr>
<tr>
<td>PART, PARTITION or NOTPART</td>
<td>PART and PARTITION specify the candidate list identifying the logical partitions which can access the device (optional). NOTPART specifies the logical partitions which cannot access the device (optional). If for an XMP processor the device has access to more than one CSS, the CSS subkeyword is required to indicate to which channel subsystem the partition belongs.</td>
</tr>
<tr>
<td>SERIAL</td>
<td>Specifies the serial number of the device (optional).</td>
</tr>
<tr>
<td>VOLSER</td>
<td>Specifies the volume serial number (optional).</td>
</tr>
<tr>
<td>CUNUMBR</td>
<td>Specifies the number(s) of the control unit(s) the device is attached to (mandatory).</td>
</tr>
<tr>
<td>DESC</td>
<td>Specifies a description of the device (optional).</td>
</tr>
<tr>
<td>UNITADD</td>
<td>Specifies the unit address that is transmitted on the channel path to select the I/O device (optional). If not specified, the last two digits of the device number are used.</td>
</tr>
<tr>
<td>PATH</td>
<td>Specifies the preferred channel path (optional).</td>
</tr>
<tr>
<td>TIMEOUT</td>
<td>Specifies whether the I/O interface timeout function is to be active (optional).</td>
</tr>
<tr>
<td>STADET</td>
<td>Specifies whether the Status Verification Facility is to be enabled or disabled (optional).</td>
</tr>
<tr>
<td>SCHSET</td>
<td>Specifies for z9 EC processors or later models the subchannel set ID where the device is located.</td>
</tr>
</tbody>
</table>

**OS parameters/features:** In the following section device specific parameters and features are described. To find out which parameters, private parameters, and features are available to you for a particular device run your Supported HW Report and I/O Definition Reference. (See “Print configuration reports” on page 324 for details on how to run the report function.) Examples of a Supported HW Report and of an I/O Definition Reference are shown in “Supported Hardware Report” on page 401 and in “I/O Definition Reference” on page 431, respectively.

**USERPRM** Allows the specification of OS private parameters. To locate the private parameters available to you for a particular device refer to your Supported HW Report. The OS private parameters are listed in column SUPPORTED PARAMETERS / FEATURES, following PRIVATE: and extending to the slash (/). For example, device 3590 listed in the sample Supported Hardware Report - MVS Devices [Figure 173 on page 402] supports the private parameters LIBRARY and AUTOSWITCH.

**FEATURE** If included in the device specific parameters, FEATURE allows device specific features to be assigned. The features available depend on your UIMs. To locate the features available to you for a particular device type refer to your Supported HW Report. The features are listed in column SUPPORTED PARAMETERS / FEATURES, following the slash (/). For example, device 3590 listed in the sample Supported Hardware
Report - MVS Devices (Figure 173 on page 402) supports the features SHARABLE and COMPACT.

ADAPTER Specifies either the terminal control or transmission adapter used to connect a telecommunications line to a transmission control unit, or the type of channel adapter that connects a communications controller to a channel path (optional).

DYNAMIC Specifies if the device is eligible for dynamic I/O configuration (optional).

LOCANY Specifies if UCB can reside in 31 bit storage (optional).

NUMSECT Specifies the number of guaranteed 256-byte buffer sections in a 2840 display-control buffer allocated to a device 2250-3 (optional).

OFFLINE Specifies if the device is considered online or offline at IPL (optional).

OWNER Specifies the subsystem or access method using the device (optional).

PCU Only applicable to a display device 2250-3 attached to a control unit 2840-2 (optional)

Identifies the 2840-2 control unit the 2250-3 is attached to. For all 2250-3 attached to the same control unit the same value is specified.

If coded, no separate IODEVICE statement UNIT=2840 must be used.

SETADDR Specifies which of the 4 set address (SAD) commands is to be issued to the transmission control unit for operations on the line specified by the ADDRESS operand (optional).

TCU Specifies the transmission control unit for a telecommunications line (optional).

Syntax:

ADDRESS=(device_number,number_of_units) 4 hexadecimal characters followed by a decimal number in the range 1 to 4095

UNIT=device_type up to 8 alphanumeric characters

MODEL=model_number up to 8 alphanumeric characters

For XMP processors:

PARTITION=((CSS(0),(lpname1[,lpname2,...]))
    ..., (CSS(m),(0))
    ...
    ..., (CSS(n),(lpname1[,lpname2,...])))

For SMP processors:

PARTITION=(lpname1[,lpname2,...]) list of partition names with up to 8 alphanumeric characters; 0,1,... for the CSS ID;

For XMP processors, a 0 is accepted as lpname to indicate a null device candidate list; that is, the control unit shared by several CSSs cannot access the device through CSS(m).
**For XMP processors:**
NOTPART=((CSS(0),(lpname1[,lpname2,...])))
... 
[,(CSS(m),(0))]
... 
[,(CSS(n),(lpname1[,lpname2,...]))]

**For SMP processors:**
NOTPART=(lpname1[,lpname2,...])

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SERIAL=serial_number</td>
<td>up to 10 numeric characters</td>
</tr>
<tr>
<td>VOLSER=volume_serial_number</td>
<td>up to 6 characters</td>
</tr>
<tr>
<td>CUNUMBR=(number1[,number2]...)</td>
<td>up to 8 hexadecimal numbers of 4 characters</td>
</tr>
<tr>
<td>DESC='description'</td>
<td>up to 32 alphanumeric characters</td>
</tr>
<tr>
<td>UNITADD=unit_address</td>
<td>2 hexadecimal characters</td>
</tr>
</tbody>
</table>
| PATH=((CSS(0),chpid[,chpid,...])
   [,(CSS(1),chpid[,chpid,...])
   ...
   [,(CSS(n),chpid[,chpid,...])
   )
   )
| TIMEOUT=value      | Y or N                                           |
| STADET=value       | Y or N                                           |
| SCHSET=n           | 0 or 1; use the short form SCHSET=n if the placement of the device is the same for all CSSs; SCHSET=0 is the default. |
| USERPRM=((param1,value1)[,(param2,value2)]...) | list of device specific parameter/value pairs |
| FEATURES=(feature1[,feature2]...) | list of device specific features |
| ADAPTER=adapter    | up to 5 alphanumeric characters                 |
| DYNAMIC=value      | Y or N                                           |
| LOCANY=value       | Y or N                                           |
| NUMSECT=number     | decimal number                                   |
| OFFLINE=value      | Y or N                                           |
| OWNER=value        | VTAM or OTHER                                    |
| PCU=number         | decimal value in the range 1 to 4095             |
| SETADDR=value      | 0, 1, 2, or 3                                    |
| TCU=value          | 2701, 2702, or 2703                              |

**Example for an SMP processor:** In the following example device numbers 0A90 to 0A9F of type 3490 are defined with unit addresses 00 to 0F. Each device is attached to control unit 0A00. It is sharable between systems, data is compacted, it is considered off-line at IPL, and supports dynamic configuration. The device does not support auto tape library nor is it automatically switchable and its UCB cannot reside above 16 MB.
IODEVICE ADDRESS=(0A90,16),UNIT=3490,UNITADD=00, *
FEATURE=(SHARABLE,COMPACT),OFFLINE=YES,DYNAMIC=YES, *
LOCANY=NO,USERPRM=((LIBRARY,NO),(AUTOSWITCH,NO)), *
CUNUMBR=0A00,DESC='VIRTUELL TAPE',SERIAL=033401

Note: Put the serial numbers in quotes if you use characters such as blanks or 
commas as a part of your serial numbers.

Example for an XMP processor: In the following example, the devices numbers 
7400 to 741F of type 3390A are defined together with an explicit candidate list: 
they can be accessed by partition TRX1 from channel subsystem 0 and from 
partition TRX2 from channel subsystem 1. The Status Verification Facility is 
enabled. The devices are placed in subchannel set 0 for channel subsystem 0 (this 
is the default and needs not be specified) and in subchannel set 1 for channel 
subsystem 1.

IODEVICE ADDRESS=(7400,032),UNITADD=50,CUNUMBR=(7300), *
STADET=Y,PARTITION=((CSS(0),TRX1),(CSS(1),TRX2)), *
SCHSET=((CSS(1),1)),UNIT=3390A

Updating parts of a configuration by migrating input data sets

The HCD incremental update function allows you to modify objects in your IODF 
by specifying the objects with I/O control statements in data sets, e.g. IOCP, 
MVSCP, or HCPRIO input data sets, and migrating these input data sets into your 
existing IODF.

Possible actions on objects using the incremental update

Table 7 shows what actions you can perform on objects by using the HCD 
incremental update function. The meaning of the markup is as follows:

| Action possible, you can make the change. |
| Action not possible, you cannot make the change with the batch migration utility but must use the HCD dialog. |
| Action possible. These attributes are deleted when the object itself is deleted. |
| Action not possible. These attributes cannot be deleted because the object itself cannot be deleted. |

See the notes below the table for further instructions on how to add, delete, or change certain objects, their attributes and connections.

<table>
<thead>
<tr>
<th>Object/Attributes</th>
<th>Add</th>
<th>Delete</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Type/model</td>
<td>x</td>
<td>(—)</td>
<td>—</td>
</tr>
<tr>
<td>Support level</td>
<td>x</td>
<td>(—)</td>
<td>—</td>
</tr>
<tr>
<td>Configuration mode</td>
<td>x</td>
<td>(—)</td>
<td>—</td>
</tr>
<tr>
<td>Serial number</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>SNA address</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>Channel Subsystem</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Maximum Number Devices</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>
Table 7. Actions on IODF Objects (continued)

<table>
<thead>
<tr>
<th>Object/Attributes</th>
<th>Add</th>
<th>Delete</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Partition</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>x</td>
<td>x 1)</td>
<td>x 11)</td>
</tr>
<tr>
<td>Image number</td>
<td>x</td>
<td>(x)</td>
<td>x 11)</td>
</tr>
<tr>
<td>Usage type</td>
<td>x 10)</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td><strong>Channel path</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHPID</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>PCHID</td>
<td>x</td>
<td>x</td>
<td>x 5)</td>
</tr>
<tr>
<td>Type</td>
<td>x</td>
<td>(—)</td>
<td>x 5)</td>
</tr>
<tr>
<td>Operation mode</td>
<td>x</td>
<td>(—)</td>
<td>x 5)</td>
</tr>
<tr>
<td>Access list</td>
<td>x</td>
<td>x</td>
<td>x 5)</td>
</tr>
<tr>
<td>Candidate list</td>
<td>x</td>
<td>x</td>
<td>x 5)</td>
</tr>
<tr>
<td>Dynamic switch</td>
<td>x</td>
<td>x</td>
<td>x 5)</td>
</tr>
<tr>
<td>Switch connection</td>
<td>x</td>
<td>x</td>
<td>x 5; 15)</td>
</tr>
<tr>
<td>CF connection</td>
<td>x</td>
<td>x</td>
<td>x 5)</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>x</td>
<td>x 5)</td>
</tr>
<tr>
<td>OS parameter</td>
<td>x</td>
<td>x</td>
<td>x 5)</td>
</tr>
<tr>
<td>I/O cluster</td>
<td>x</td>
<td>x</td>
<td>x 5)</td>
</tr>
<tr>
<td><strong>Control unit</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number</td>
<td>x</td>
<td>x 2)</td>
<td>x 4)</td>
</tr>
<tr>
<td>Unit/model</td>
<td>x</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Serial number</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Switch connection</td>
<td>x</td>
<td>x</td>
<td>x 15)</td>
</tr>
<tr>
<td>Channel paths</td>
<td>x</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td>DLA</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Logical address (CUADD)</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Unit addresses</td>
<td>x</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td>Protocol</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>IOCL</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td><strong>Device</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number</td>
<td>x</td>
<td>x 3)</td>
<td>x 4)</td>
</tr>
<tr>
<td>Unit/model</td>
<td>x</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Serial number</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Volume serial number</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Control units</td>
<td>x</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td>Processor connect</td>
<td>x</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td>Unit address</td>
<td>x</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td>Preferred CHPID</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>TIMEOUT</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>STADET</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Candidate list</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>OS connect</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Subchannel set</td>
<td>x</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td>Parameters</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Features</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>User parameters</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td><strong>Operating system</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Type</td>
<td>x</td>
<td>(—)</td>
<td>—</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td><strong>EDT</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Object/Attributes</td>
<td>Add</td>
<td>Delete</td>
<td>Change</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td><strong>Esoteric</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>VIO</td>
<td>x</td>
<td>(—)</td>
<td>x</td>
</tr>
<tr>
<td>Device list</td>
<td>x</td>
<td>(x 6)</td>
<td>x 7)</td>
</tr>
<tr>
<td>Token</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td><strong>Generic</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VIO</td>
<td>x</td>
<td>(—)</td>
<td>x</td>
</tr>
<tr>
<td>Preference value</td>
<td>x</td>
<td>(—)</td>
<td>x</td>
</tr>
<tr>
<td><strong>Console</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Device list</td>
<td>x</td>
<td>x 9)</td>
<td>x 13)</td>
</tr>
<tr>
<td>Order</td>
<td>(x)</td>
<td>(x)</td>
<td>x</td>
</tr>
<tr>
<td><strong>Switch</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Unit/model</td>
<td>x</td>
<td>(—)</td>
<td>x</td>
</tr>
<tr>
<td>Ports (installed range)</td>
<td>x</td>
<td>(—)</td>
<td>x 8)</td>
</tr>
<tr>
<td>Serial number</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>Switch connection</td>
<td>x</td>
<td>(x 14)</td>
<td>x 15)</td>
</tr>
<tr>
<td>Address</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td><strong>Ports</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Name</td>
<td>x</td>
<td>(—)</td>
<td>x</td>
</tr>
<tr>
<td>Occupied indicator</td>
<td>x 15)</td>
<td>x 15)</td>
<td>n/a</td>
</tr>
<tr>
<td><strong>Switch configuration</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>x</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Description</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>Default connection</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td><strong>Port configuration</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Allowed connection</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>Prohibited connection</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>Dedicated connection</td>
<td>x</td>
<td>—</td>
<td>x</td>
</tr>
<tr>
<td>Blocked indicator</td>
<td>x</td>
<td>x</td>
<td>n/a</td>
</tr>
</tbody>
</table>

References to IOCP and MVSCP in the following notes refer to data sets with extended syntax as described in "Changing I/O configurations by editing data sets" on page 273.

**Notes:**

1. To delete a partition, specify all connected channel paths (defined via access or candidate lists) together with their attached I/O units without referring to the partition and without repeating the partition in the RESOURCE statement.

2. A control unit is implicitly deleted, if its channel paths are respecified in the IOCP input data set together with their attached I/O units (except the control unit) and it no longer has any connection to a processor.

3. A device is implicitly deleted, if its attaching control units are implicitly deleted using the incremental update specification, and there is no additional connection left to any control unit or operating system.

4. To change control unit and device attributes, specify the entire logical control unit(s) (LCU) the control unit or device is part of. For migration to an OS configuration only (MVSCP data set), respecify the I/O device with the changed parameters. HCD will redefine the device in the corresponding subchannel set.
5. To change channel path attributes, specify all logical control units the channel path is connected to in the IOCP input data set. Otherwise, the channel path is disconnected from the corresponding control units.

6. To delete an esoteric device list, specify all device definitions in the MVSCP input data set, but do not connect them to the esoteric name.

7. To add a device to an esoteric device list, specify the esoteric with the device number.

   To delete a device from the esoteric device list, specify the device in an IODEVICE statement) but do not specify the device number for the esoteric device list.

8. An installed port can be set to uninstalled only if it does not belong to the minimum installed port range and does not hold a connection to a channel path or control unit.

9. To delete a console list, specify all contained devices using the IODEVICE statement but do not include the devices in a NIPCON statement.

10. If the usage type is not specified and you add a partition, the usage type is automatically defined: if the IOCP input data set contains a CF receiver channel path with the partition in its access or candidate list, the usage type is set to CF/OS, if not, the usage type is set to OS.

11. To change the partition name or partition number specify the whole partition configuration including all channel paths with attached I/O units which have the partition in their access and candidate lists.

12. (No longer used.)

13. To change a complete console device list, use the NIPCON statement.

   To remove a single device from the console list, specify the corresponding IODEVICE statement and omit the NIPCON statement.

14. To delete a switch-to-switch connection, specify a switch via a SWITCH statement and omit the switch-to-switch connection in the SWPORT parameter.

15. When updating switch ports new connections always overwrite a previous connection or status. To be updated with an occupied status the port must currently not be connected.

How to invoke the incremental update

1. Specify your objects with IOCP, MVSCP, or HCPRIO control statements. Note that you can add additional parameters and SWITCH statements to exploit the extended migration as described under "Changing I/O configurations by editing data sets" on page 273.

2. Select Migrate configuration data on the Primary Task Selection panel and on the resulting panel the Migrate IOCP/OS data option.

3. On the following Migrate IOCP / MVSCP / HCPRIO Data panel (see Figure 120 on page 269), enter the required data and change the incremental update to Yes.

4. After the input has been accepted, HCD issues a message informing you that the migration of input data sets is in process.

Example 1: Adding a partition

In this example, you specify a partition in your input data set that does not yet exist in the IODF. The attached control units and devices are already defined in the IODF for another processor.
The following figure illustrates the result after the incremental update:

```
<table>
<thead>
<tr>
<th>IODF</th>
<th>IOCP Input Data Set</th>
<th>Resulting IODF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>100 3880-3</td>
<td>200 3990-3</td>
<td>200 3990-3</td>
</tr>
<tr>
<td>06C0,32 3380</td>
<td>05C0,32 3390</td>
<td>05C0,32 3390</td>
</tr>
</tbody>
</table>
```

Figure 121. Partial Migration of an IOCP Input Data Set. A new partition is added. Control units and devices are mapped.

The partition is added and the control unit and devices are mapped. For a detailed description when control units and devices are mapped, refer to "Migrating additional IOCP input data sets" on page 265.
**Example 2: Replacing a channel path and attached control unit**

In this example, you specify a channel path with attached control unit and devices in an IOCP input data set. The channel path, the control unit, and one device already exist in the IODF.

The following figure illustrates the result after the incremental update:

<table>
<thead>
<tr>
<th>IODF</th>
<th>IOCP Input Data Set</th>
<th>Resulting IODF</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="" /></td>
<td><img src="image" alt="" /></td>
<td><img src="image" alt="" /></td>
</tr>
</tbody>
</table>

`CHPID PATH=01,TYPE=CNC,PART=PART1
CNTLUNIT CUNUMBR=100,UNIT=3990,PATH=01,...
IODEVICE ADDRESS=(5C0,32),UNIT=3390,CUNUMBR=100`

*Figure 122. Partial migration of an IOCP input data set.* The whole logical control unit (control unit 100 and 200 and connected devices) are replaced by control unit 100 and its connected devices.

The logical control unit in the IOCP input data set replaces the whole logical control unit in the IODF.
Example 3: Replacing a channel path with a new control unit

In this example, you specify channel paths with control unit and device in the input data set. The channel paths are already defined in the IODF, but connect to another control unit.

The following figure illustrates the result after the incremental update:

![Diagram showing channel paths and control units]

Figure 123. Partial migration of an IOCP input data set. CHPIDs are defined as in the IOCP input data set.

The channel paths in the IODF are deleted and newly defined as they are defined in the IOCP input data set.
Example 4: Replacing a control unit that attaches to two processors
In this example, you want to replace existing control units and devices by another type. The control units and devices are attached to two processors.

The incremental update must be done in several steps:
1. Specify another control unit number in your IOCP input data set than the one in the existing IODF. Specify the CHPID, CNTLUNIT, and IODEVICE control statements.
2. Migrate your input data set for the first processor.
3. Migrate your input data set again for the second processor.

Figure 124. Partial migration of an IOCP input data set. Control unit defined for two processors is migrated to the first processor.
Example 5: Updating an operating system

In this example, you update an operating system by adding and changing EDTs, generics, esoterics, and console devices.

<table>
<thead>
<tr>
<th>IODF</th>
<th>IOCP Input Data Set</th>
<th>Resulting IODF</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROC 1</td>
<td>PROC 2</td>
<td>PROC 2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>A100</td>
<td>100</td>
<td>A100</td>
</tr>
<tr>
<td>3990-3</td>
<td>3880-3</td>
<td>3990-3</td>
</tr>
<tr>
<td>05C0,32</td>
<td>05C0,32</td>
<td>05C0,32</td>
</tr>
<tr>
<td>3390</td>
<td>3380</td>
<td>3390</td>
</tr>
</tbody>
</table>

CHPID | PATH=21, TYPE=CNC,...
CNTLUNIT | CUNUMBR=A100, UNIT=3990, PATH=(21)
IODEVICE | ADDRESS=(5C0,32), UNIT=3390,
CUNUMBR=A100

Figure 125. Partial migration of an IOCP input data set. Control unit defined for two processors is migrated to the second processor.

Example 5: Updating an operating system

In this example, you update an operating system by adding and changing EDTs, generics, esoterics, and console devices.

<table>
<thead>
<tr>
<th>IODF</th>
<th>MVSCP Input Data Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDT ID=01</td>
<td>EDT ID=01</td>
</tr>
<tr>
<td>UNITNAME=POOL1, UNIT=((700,8),(900,8))</td>
<td>UNITNAME=POOL1, UNIT=((800,8)), VIO=YES</td>
</tr>
<tr>
<td>UNITNAME=POOL2, UNIT=((100,16))</td>
<td>UNITNAME=POOL3, UNIT=((0270,16))</td>
</tr>
<tr>
<td>UNITNAME=TAPES, UNIT=((A80,4))</td>
<td>UNITNAME=9345, VIO=YES</td>
</tr>
<tr>
<td>NIPCON DEVNUM=(40,41,42)</td>
<td>NIPCON DEVNUM=(40,50,60)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resulting IODF</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>EDT ID=01</td>
<td>EDT ID=01</td>
<td></td>
</tr>
<tr>
<td>UNITNAME=POOL1, UNIT=((700,8),(800,8),(900,8)), VIO=YES</td>
<td>UNITNAME=POOL1, UNIT=((800,8)), VIO=YES</td>
<td></td>
</tr>
<tr>
<td>UNITNAME=POOL2, UNIT=((100,16))</td>
<td>UNITNAME=POOL3, UNIT=((0270,16))</td>
<td></td>
</tr>
<tr>
<td>UNITNAME=TAPES, UNIT=((A80,4))</td>
<td>UNITNAME=9345, VIO=YES</td>
<td></td>
</tr>
<tr>
<td>NIPCON DEVNUM=(40,41,42)</td>
<td>NIPCON DEVNUM=(40,50,60)</td>
<td></td>
</tr>
</tbody>
</table>

Figure 126. Partial migration of an MVSCP input data set
You can either migrate the changes with a combined input data set or with an MVSCP only input data set. With a combined input data set, you can make CSS and operating system changes at the same time.

If you migrate an MVSCP only input data set, specify a processor and partition, with which device definitions in the input data sets will be associated in the IODF, on the Migrate IOCP / MVSCP /HCPRIO Data panel. HCD uses this information to map devices correctly if the IODF contains duplicate device numbers. For more information about this mapping, refer to “Migrating additional MVSCP or HCPRIO input data sets” on page 267.

The result after the migration is as follows:

- The definition of EDT input statements replaces the definition in the IODF, new definitions are added. That means in the example above, EDT 01 is replaced and EDT 02 is added.
- Devices are added. To add devices, you also have to newly define the esoterics to which the devices are to be assigned.
- Devices in a UNITNAME statement are added to the devices already defined for the esoteric.
- The list of consoles is replaced.

**Note:** If an EDT statement is missing or specified without EDT ID, the EDT ID for esoterics and generics is taken from the ID of the IOCONFIG statement. If the IOCONFIG statement is also not available, the EDT ID will be assumed as ‘00’.
Resolving migration errors

If HCD detects an error when you migrate using the HCD dialog, it displays a message list when the migration has been completed.

Figure 127 is an example of such a message list. It lists all the messages that were issued during the process. The list shows the statement number of the input statement for which the message was issued. You can now:

- Get explanations of messages in the list. To get explanations, use the Explain message action from the context menu or action code $ Explain $.
- Delete messages that are of no further interest. That is, delete those messages that you do not want to appear in the HCD migration log.

When you exit the message list or when you migrated your input data sets using the batch utility, HCD writes the error messages to the HCD migration log.

You can display the HCD migration log through ISPF. The name of the HCD migration log data set is developed from the name of the input data set, without high-level qualifier, and your user ID as follows:

userid.yyy.zzz.MESSAGES

where the input data set, without high-level qualifier, is:

yyy.zzz

or

yyy(zzz)
Errors detected during assembly process

During migration, HCD invokes the assembler to parse the input statements. If it encounters an error, it writes a message to the message list. This message points to an assembler listing for more details.

Figure 128. Message List containing an Assembler message

The name of the assembly listing data set is developed from the name of the input data set, without high-level qualifier, and your user ID as follows:

userid.yyy.zzz.LISTING

where the input data set, without high-level qualifier, is:

yyy.zzz

or

yyy(zzz)

At the bottom of the assembly listing, you find the statement numbers where errors were detected. If you locate the statement numbers in the listing, you see the IOCP statement in error followed by the error message. The following examples show IOCP statements and the assembly messages issued.

Example 1

In this example an entry in the IOCP input data set is commented out, but the continuation character $ is left in column 72.

7493 * CNTLUNIT CUNUMBR=02E,PATH=(02,06),SHARED=N,UNIT=3990, 91A$
7494 * UNITADD=((E),32)),PROTOCL=S4

The assembler listing shows the following message:

IEV144 *** ERROR *** BEGIN-TO-CONTINUE COLUMNS NOT BLANK

Example 2

This example shows a line with only 87A being included in the IOCP input data set.

7493 * CNTLUNIT CUNUMBR=234,PATH=24,SHARED=N,UNIT=3880, 87A
7494 * UNITADD=(30,16),PROTOCL=S4
7495
7496 *IOCP

The assembler listing shows the following message:

IEV144 *** ERROR *** OPERATION CODE NOT COMPLETE ON FIRST CARD
Example 3
This example shows a wrong channel path type in the CHPID statement.
CHPID PATH=((00)),TYPE=CMC

The assembler listing shows the following message:
IEV144 *** MNOTE *** 8,003 TYPE=CMC IS INVALID

Errors detected during validation process
During the validation process, HCD checks that the definitions being migrated do not cause any conflicts with existing definitions in the IODF and with other definitions being migrated. HCD also checks that the contents of the input data sets is valid. The errors are shown in the message list and migration log.

Figure 129 shows error messages in the migration log.

<table>
<thead>
<tr>
<th>Statement</th>
<th>Orig</th>
<th>Sev</th>
<th>Msgid</th>
<th>Message Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 9 (1) E CBDA230I Duplicate unit address F0 on channel path 01 of processor BOEHCD.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 (2) W CBDA265I Type of control unit 0131 assumed as 3880-13 to attach device 01F0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I CBDA516I No output written to IODF. VALIDATE processing forced due to errors.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total Messages: Terminating Error Warning Informational
3 0 1 1 1

Return Code was 8

Figure 129. Example: HCD Migration Log

The messages are sorted according to their severity, and within a certain severity level according to their occurrence.

The value in the Orig column points to the input data set that caused this error. At the top of the migration log you find a reference list that shows the values with the names of the input data sets (see the line marked 1).

In Figure 129 the first message line means, that the statement number 9 in the input data set 1 (data set BBEL.IOCP01.CTL) is the cause of the error message.

The following examples show common validation errors and explain their causes.

Example 1
Statement Orig Sev Msgid Message Text
7 (1) E CBDA154I Channel path type CNC is not supported by channel path ID 3A.

This message is issued, because an ESCON channel is defined although the support level was defined in the IODF as having only parallel channels installed for the specified CHPID. To resolve this, either change the channel type in the IOCP input data set, or change the processor type or support level in the IODF.
Example 2
Statement Orig Sev Msgid Message Text
4 (1) E CBDA234I Unknown type 38823 of control unit 0000 specified.

This message is issued, because HCD does not know the control unit type 38823. Select the Query supported hardware and installed UIMs from the Primary Task Selection panel or use the Query action bar choice for information on valid control unit and device types.

Example 3
Statement Orig Sev Msgid Message Text
228 (1) W CBDA265I Type 3800-3 assumed for control unit DD32 to attach the device 0828.
227 (1) I CBDA534I Control unit DD32 is assumed as 3800-1.

These messages are issued, because HCD has to choose the control unit type among several models.

The control unit model 3800-1 is indicated as default model in the UIM (information message CBDA534I is issued). As processing goes on, it is necessary to change the default model (3800-1) to another model (3800-3) to attach a device type which is not supported by the default control unit model (warning message CBDA265I is issued).

Note: The sequence of messages is shown in reverse order in the migration log file since the messages are sorted according to decreasing severities. To resolve this, either include the model in the IOCP input data set, map the control unit types via HCD profile entries, or update the IODF using the HCD dialog, if HCD has made an incorrect assumption.

Insufficient data set sizes
HCD dynamically allocates the data sets required for migration. It can happen that the default data set sizes are insufficient for the migration of the existing data. During the migration process, the system informs you by a message which data set needs to be enlarged. In that case, the data set needs to be preallocated with a larger size before invoking the HCD migration task again.

HCD uses, by default, the following ddnames and data set sizes:
• HCDPRINT
  Used for the HCD migration log.
  The data set name is built from the input data set name — if two input data sets are specified from the IOCP input data set name — qualified with ‘MESSAGES’.
  If the input is a member of a partitioned data set, an additional qualifier - the name of the member - is inserted before ‘MESSAGES’. The high-level qualifier of that data set name is replaced by the TSO prefix (user ID).
  The minimum allocation (also used as default allocation if the data set does not exist) is: RECFM=FBA, LRECL=133, BLKSIZE=2926, SPACE=(TRK,(1,10)), exclusive access.
• HCDASMP
  Contains the assembler SYSPRINT data set, which contains the assembly listing (input statements with sequence numbers and messages).
The data set name is built from the input data set name qualified with 'LISTING'. If the input is a member of a partitioned data set, an additional qualifier - the name of the member - is inserted before 'LISTING'. The high-level qualifier of that data set name is replaced by the TSO prefix (user ID).

The minimum allocation (also used as default allocation if the data set does not exist) is as follows: RECFM=FBA, LRECL=121, BLKSIZE=1573, SPACE=(TRK,(15,150)), exclusive access.

- **HCDUT1**
  Used by the assembler as a work data set (UNIT=SYSALLDA, LRECL=80, BLKSIZE=3200, SPACE=(TRK,(15,150)), exclusive access).

- **HCDUT2**
  Used as output data set for the modified input stream and by the assembler as SYSIN data set (UNIT=SYSALLDA, LRECL=80, SPACE=(TRK,(10,15)), exclusive access).

- **HCDUT3**
  Used by the assembler as punch data set (SYSPUNCH).
  Used by the loader as input data set (UNIT=SYSALLDA, LRECL=80, SPACE=(TRK,(10,30)), exclusive access).

**Notes:**

1. The sizes of the output data sets HCDUT2, HCDASMP, and HCDPRINT depend on the size of the input and on the number of messages produced. The space given above should be taken as minimum allocation values.

2. For HCDASMP and HCDPRINT, HCD checks whether data sets with the default names exist. If so, the space allocations of these existing data sets are used if they exceed the minimum allocation values. If they are below the minimum allocation value, the data sets are deleted and allocated with a new (minimum) size. Correspondingly, the space of the HCDUT2 data set is made dependent on the input data set(s). This rule is only applicable if the ddnames have not been previously allocated.

3. Preallocate HCDASMP and HCDPRINT if:
   - You want to have them on a different data set than the default one
   - You want to place these data sets on a specific volume
   - The default size is not large enough
   - A size different from the default size should be used.
Chapter 12. How to invoke HCD batch utility functions

Programming Interface information

You can invoke HCD batch utility functions:

- With an input parameter string. The diagram below shows how to invoke an HCD batch utility function. For formats of the input parameter strings and sample batch jobs see "Input parameter string" on page 308.

```
HCD Batch Invocation

EXEC PGM=CBDMGHCP, PARM='Input Parameter String'
```

- By using an ATTACH or LINK module programming statement to invoke the module CBDMGHCP.

When you invoke the module, register 1 must contain the address of a two-word parameter list.

**Word 1**
Address of input parameter (see "Input parameter string" on page 308) preceded by a two byte length field.

**Word 2**
Address of a list of alternate DD names. If not used, the address must be binary zero. For the list format of alternate DD names see "List of alternate DD names" on page 338.

You may overwrite standard DD names listed in Table 8 on page 337 as desired before invoking HCD.

If you specify the UIMs and UDTs in a library other than SYS1.NUCLEUS, you have to add the following statement to your batch jobs:

```
//HCDPROF DD DSN=BPAN.HCD.PROF,DISP=SHR
```

In the HCD profile (in our example BPAN.HCD.PROF) specify the following keyword:

```
UIM_LIBNAME=libname
```

If the keyword is omitted, SYS1.NUCLEUS is assumed. If you specify an asterisk (*) as data set name, HCD assumes that the UIM data set is part of the ISPF load library concatenation chain, contained in the JOBLIB/STEPLIB concatenation chain, or specified in the active LNKLSTxx member. For more information, see "Defining an HCD profile" on page 23.

Running jobs in a sysplex environment:

If you want to execute a job on a specific system in a sysplex, you must specify in your batch job which system is to be used. If you do not specify the exact system of a sysplex for which the batch job is planned to execute on, the job executes on the system that has the free space to run on.
The output from some of the HCD functions depend very much on where the job was executed; for example, downloading IOCDSs and requesting the I/O Path report.

---

**How to Read Syntax Diagrams**

For details on this subject see “How to read syntax diagrams” on page xv.

---

**Note:** Trailing commas in the parameter string can be omitted.

---

**Input parameter string**

---

**Input Parameter String**

- TRACE
- Start the dialog
- Initialize a VSAM data set into an IODF
- Upgrade an IODF
- Migrate I/O configuration statements
- Build a production IODF
- Build a work IODF from a production IODF
- Build an IOCDS or an IOCP input data set
- Build an HCPRIO input data set
- Build I/O configuration data
- Copy an IODF
- Print a configuration report
- Create a graphical configuration report
- Compare IODFs or CSS/OS views
- Import an IODF
- Export an IODF

---

**TRACE**

When specified, the HCD trace will be activated.

---

**Input Parameter String**

You will find a detailed description of the input parameter strings in the following sections.

You can also activate tracing by adding the TRACE command in the HCD profile. This allows you to specify the trace parameters in more detail. In this case, you must allocate DD name HCDPROF to the HCD profile when invoking the batch utility. For more information, see “Defining an HCD profile” on page 23, and “TRACE command” on page 469.

---

**Function parameter strings**

---

**Start the dialog**

This utility function starts the HCD (ISPF) Dialog session. The HCD primary task selection panel is displayed.

This function is invoked by passing the following parameter string.
E | J is a one-character code for national language support in help panels and messages. Specify one of the following (if omitted, the default is E):

  E for English
  J for Japanese

### Initialize IODF

This utility function initializes a defined VSAM DIV file into an IODF. Each IODF contains as first record a header record, called the IHR (IODF Header Record). This record contains, among other information, the size of the IODF, an optional description of up to 128 characters, as well as an option whether activity logging is enabled or disabled. HCD rejects any data set that does not contain such a header record.

The VSAM DIV file must be preallocated using DD name HCDIODFT. You can add an optional IODF description using DD name HCDCNTL.

This function is invoked by passing the following parameter string.

```plaintext
INITIODF SIZE=nnnn

SIZE=nnnn specifies the size of the IODF in 4K blocks. This value must not be greater than the number of records specified with the IDCAMS Define Cluster control statement. If SIZE=0 is specified, the number of allocated records of the VSAM data set is used. If an existing IODF is re-initialized, the specified size value must not be smaller than the number of allocated IODF blocks.

ACTLOG=(YES | NO)

specifies enabling of activity logging. If omitted, the default is YES.

FORCE

indicates that reinitialization of an existing IODF is allowed.

### Batch invocation

A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFT</td>
<td>IODF to be initialized</td>
</tr>
</tbody>
</table>
The following example shows the IDCAMS control statements necessary to define a VSAM DIV file.

```
DEFINE CLUSTER (NAME (SYS1.IODF01.CLUSTER) -
    LINEAR -
    RECORDS (1024) -
    VOLUMES (DATA02) -
) -
    DATA (NAME (SYS1.IODF01))
```

For an example see the batch job on page 335.

Notes:
1. This batch job issues a job message IEC161I, which can be ignored.
2. The VSAM DIV file consists of a data and a cluster file. According to the IODF naming convention (see “IODF naming convention” on page 35), the name of the data file is the IODF name (in this example SYS1.IODF01), and .CLUSTER is appended to the data file for a cluster file. To define your VSAM DIV file, you must append .CLUSTER to the IODF name in the DEFINE CLUSTER statement (in this example SYS1.IODF01.CLUSTER).

**Upgrade IODF**

This utility function upgrades a back-level IODF to be accessible with the current HCD release.

You have to allocate:
- The IODF you want to upgrade with DD name HCDIODFS
- The IODF into which the I/O definitions are to be upgraded with DD name HCDIODFT

This function is invoked by passing the following parameter string,

![Upgrade an IODF](image)

Notes:
1. The target data set must be large enough to hold the source IODF.
2. The target IODF may be created using the utility *Initialize IODF* (described in “Initialize IODF” on page 309).
3. Both data sets must be valid IODFs.

**Batch invocation**

A data set must be allocated to the following DD names when invoking the batch utility.
<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFS</td>
<td>Back-level IODF to be upgraded</td>
</tr>
<tr>
<td>HCDIODFT</td>
<td>IODF into which IODF definitions are to be upgraded (if not specified, the IODF is upgraded in place)</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

**Example:**

```
//BWINJOB JOB (3259,RZ-28), 'BWIN', NOTIFY='BWIN', CLASS=A,
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M
//*
//* UPGRADE IODF
//*
//UPGRADE EXEC PGM=CBDMGHCP,PARM='UPGRADE'
//HCDIODFS DD DSN=BWIN.IODFR2.WORK, DISP=SHR
//HCDIODFT DD DSN=BWIN.IODF00.WORK, DISP=OLD
//HCDMLOG DD DSN=BWIN.HCD.LOG, DISP=OLD
//
```

For considerations concerning the size when upgrading a back-level IODF, refer to Table 4 on page 49.

**Migrate I/O configuration statements**

This utility function allows you to migrate the data set containing I/O configuration statements, e.g. an IOCP, MVSCP, or HCPRI0 input data set and store the definitions into an IODF.

You have to allocate:
- The IODF into which the I/O definitions are to be migrated with DD name HCDIODFT
- The I/O configuration input data set with DD name HCDIN
- The MACLIB containing the parsing macros with DD name HCDLIB

This function is invoked by passing the following parameter string,
Migrate I/O configuration statements

processor parameters:
- procid
- proctype
- procmode
- support_level_ID

OS parameters:
- osid
- ostype

association:
- asproc
- aspart

Function indicator:

I | Migration of processor configuration statements (e.g. IOCP data sets)
IP | Partial migration of processor configuration statements
O | Migration of OS configuration statements (for example MVSCP or HCPRIO data sets)
OP | Partial migration of OS configuration statements
B | Combined migration of processor and OS configuration statements
BP | Partial combined migration of processor and OS configuration statements
Switch migration

C physical channel ID (PCHID) migration

Processor related variables and keywords:

**procid**  
Processor ID (up to 8 characters)

**procid.cssid, procid(#cssid)**  
When migrating an SMP processor to an XMP processor, the channel subsystem ID of the target processor may be appended to the processor ID as one character either by a # and in parenthesis or by a dot (.). The default is 0.

**proctype**  
Processor type and model separated by a hyphen, for example 9672-E08

**procmode**  
Processor mode

- **BASIC**  
  If the processor operates in BASIC mode (default).

- **LPAR**  
  If the processor operates in LPAR mode.

**support_level_ID**  
Support level ID associated with the processor. This parameter is required if the processor does not already exist and several support level IDs are installed for a supported processor type. The support level ID can be obtained by the List supported processors function or by the supported hardware report described in “Print configuration reports” on page 324. For an example of a supported hardware report refer to “Supported Hardware Report” on page 401.

If you do not specify a support level, the highest support level will be used for the processor.

**Note:** The support level ID is unique to HCD and does not correspond to the EC level of the processor.

**asproc**  
Associated processor. For more information, see “Migrating additional MVSCP or HCPRI0 input data sets” on page 267

**aspart**  
Associated partition. For more information, see “Migrating additional MVSCP or HCPRI0 input data sets” on page 267

**OS related parameters:**

**osid**  
Operating system ID (up to 8 characters)

**ostype**  
OS type (MVS or VM)

**asproc**  
Associated processor. For more information, see “Migrating additional MVSCP or HCPRI0 input data sets” on page 267

**aspart**  
Associated partition. For more information, see “Migrating additional MVSCP or HCPRI0 input data sets” on page 267

**allows multiple OS configurations to be migrated. Scans the input data set to determine which OS configurations are to be processed. For successful migration the configurations must include the IOCONFIG statement described in “Operating system” on page 276.**
Switch related wildcard:

* Allows switch configurations of multiple switches to be migrated. Scans the input data set to determine which switch configurations are to be processed. For successful migration the configurations must include the SWCONF statement described in "Switch configuration" on page 280.

Batch invocation

A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFT</td>
<td>IODF into which I/O definitions are to be migrated</td>
</tr>
<tr>
<td>HCDIN</td>
<td>I/O configuration input data set</td>
</tr>
<tr>
<td>HCDLIB</td>
<td>MACLIB containing the parsing macros (CBDZPARS)</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDPRINT</td>
<td>Data set for migration log (see &quot;Insufficient data set sizes&quot; on page 305)</td>
</tr>
<tr>
<td>HCDASMP</td>
<td>Data set for assembly listing (see &quot;Insufficient data set sizes&quot; on page 305)</td>
</tr>
<tr>
<td>HCDPROF</td>
<td>HCD profile (when using extended migrate function)</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

Example:

```
//BWINJOB  JOB (3259,RZ-28), 'BWIN', NOTIFY='BWIN', CLASS=A,
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M
// *
//** MIGRATE AN IOCP DECK
//**
//MIGRATE EXEC PGM=CBDMGHCP,
// PARM='MIGR,1,PROC1,9672-E08,LPAR'
//HCDIODFT DD DSN=BWIN.IODF03.WORK,DISP=OLD
//HCDIN DD DSN=BWIN.IOC.P.DECK,DISP=SHR
//HCDLIB DD DSN=SYS1.MACLIB,DISP=SHR
//HCDMLOG DD DSN=BWIN.HCD.LOG,DISP=OLD
//HCDPRINT DD DSN=BWIN.IOC.P.MESSAGES,DISP=OLD
//HCDASMP DD DSN=BWIN.IOC.P.LISTING,DISP=OLD
//```

Build a Production IODF

This utility function creates a production IODF using the work IODF. The work IODF must be specified with DD name HCDIODFS, the target production IODF with DD name HCDIODFT. First, the target production IODF has to be created by defining a VSAM DIV file and initializing it using the utility Initialize IODF (see "Initialize IODF" on page 309).

If the work IODF has an associated MCF, the MCF data set is copied and associated to the production IODF.

After the production IODF has been built, it is copied back to the work IODF. Thus, the work IODF contains the new tokens and can be used for further updates.
Since the production IODF may be larger than the original work IODF, the work
IODF may be automatically enlarged to accommodate the contents of the
production IODF.

This function is invoked by passing the following parameter string.

```
Build a Production IODF

PRODIODF DESC1=descriptor1 DESC2=descriptor2 DESC1=descriptor1,DESC2=descriptor2
```

**DESC1=descriptor**

Default is the first qualifier of the production IODF name (up to 8
characters).

**DESC2=descriptor**

Default is the second qualifier of the production IODF name, which is
IODFxx (up to 8 characters).

The descriptor fields describe the IODF and will be part of the HSA token.
Attention: If you specify asterisks (**), equals (==), pluses (+++), or minuses (−−) for
the IODF suffix in LOADxx, never change the default descriptor field values,
because z/OS uses these values to find the current IODF during IPL. Take this
relationship also into consideration, if you copy the IODF to a different data set
name.

**Batch invocation**

A data set must be allocated to the following DD names when invoking the batch
utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFS</td>
<td>Work IODF</td>
</tr>
<tr>
<td>HCDIODFT</td>
<td>Production IODF</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

Example:

```c
//BWINJOB JOB (3259,RZ-28), 'BWIN', NOTIFY=BWIN, CLASS=A,
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M
//*
//* BUILD PRODUCTION IODF
//*
//PROD EXEC PGM=CBDMGHCP,
// PARM='PRODIODF DESC1=BWIN,DESC2=IODF03'
//HCDIODFS DD DSN=BWIN.IODF03.WORK,DISP=OLD
//HCDIODFT DD DSN=BWIN.IODF03,DISP=OLD
//HCDMLOG DD DSN=BWIN.HCD.LOG,DISP=OLD
//```
Build a work IODF from a production IODF

This utility function creates a work IODF using an existing production IODF. The production IODF has to be specified with DD name HCDIODFS, the target work IODF with DD name HCDIODFT. First, the work IODF has to be created by defining a VSAM DIV file and initializing it using the utility Initialize IODF (see "Initialize IODF" on page 309).

This function is invoked by passing the following parameter string.

---

**Build a Work IODF**

```
WORKIODF
```

---

Batch invocation

A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFS</td>
<td>Production IODF</td>
</tr>
<tr>
<td>HCDIODFT</td>
<td>Work IODF</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

Example:

```
//BWINJOB JOB (3259,RZ-28),'BWIN',NOTIFY=BWIN,CLASS=A,       
//MSGCLASS=Q,MSGLEVEL=(1,1),REGION=4M
//*/
//* BUILD WORK IODF
//*/
//WORK EXEC PGM=CBDMGHCP,PARM='WORKIODF'
//HCDIODFS DD DSN=BWIN.IODF03,DISP=SHR
//HCDIODFT DD DSN=BWIN.IODF03.WORK,DISP=OLD
//HCDMLOG DD DSN=BWIN.HCD.LOG,DISP=OLD
//*/
```

Build an IOCDS or an IOCP input data set

This utility function builds the IOCDS or the IOCP input data set using the definitions of a production IODF. This function is invoked by passing the following parameter string.
Build an IOCDS or an IOCP input data set

```
  IOCDS, [iocds], [procid], [ind]
```

**ind**

- **D** - Build an IOCP input data set
- **SA**
  - The generated IOCP statements can be used for the stand-alone IOCP program (default).
- **NOSA**
  - Depending on the HCD profile option MIGRATE_EXTENDED=YES, the generated IOCP statements have additional information that can be used for the extended migration. This information is shown as comments to IOCP.
  - **Note**: An IOCP input data set generated with operand NOSA may not be accepted by the stand-alone IOCP program, because of differences between the IOCP program running in z/OS and the stand-alone IOCP program.
- **I** - Build an IOCDS
- **NOCHKCPC**
  - Write an IOCDS regardless of the type of the receiving processor. Refer to "Supported Hardware Report" on page 401 for a list of processor types that can receive an IOCDS in preparation for a processor upgrade and for processor types for which such an IOCDS can be written.
- **LOCALWRT**
  - This parameter enforces a local IOCDS write. A defined SNA address which is normally used by HCD to initiate a remote IOCDS build to the support element with the designed SNA address, is ignored in this case.

**iocds** is a two-character IOCDS identifier, if building an IOCDS

**procid** Processor ID

**D | I | W | S | T**

One-character request code:

- **D** - Build an IOCP input data set
- **SA**
  - The generated IOCP statements can be used for the stand-alone IOCP program (default).
- **NOSA**
  - Depending on the HCD profile option MIGRATE_EXTENDED=YES, the generated IOCP statements have additional information that can be used for the extended migration. This information is shown as comments to IOCP.
  - **Note**: An IOCP input data set generated with operand NOSA may not be accepted by the stand-alone IOCP program, because of differences between the IOCP program running in z/OS and the stand-alone IOCP program.
- **I** - Build an IOCDS
- **NOCHKCPC**
  - Write an IOCDS regardless of the type of the receiving processor. Refer to "Supported Hardware Report" on page 401 for a list of processor types that can receive an IOCDS in preparation for a processor upgrade and for processor types for which such an IOCDS can be written.
- **LOCALWRT**
  - This parameter enforces a local IOCDS write. A defined SNA address which is normally used by HCD to initiate a remote IOCDS build to the support element with the designed SNA address, is ignored in this case.
Build an IOCDS with dual-write option (optionally with NOCHKCPC and LOCALWRT, see option I).

Build an IOCDS and set the IOCDS active for next POR

Build an IOCDS with dual-write option and set the IOCDS active for next POR

**Batch invocation**

A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFS</td>
<td>Source IODF</td>
</tr>
<tr>
<td>HCDDECK</td>
<td>IOCP data set (if build IOCP input data set is requested)</td>
</tr>
<tr>
<td>HCDCNTL</td>
<td>Control data set for specifying the MSG1 IOCP parameter</td>
</tr>
<tr>
<td>SYSPRINT</td>
<td>SYSPRINT data set for IOCP output listing (requested for build IOCDS)</td>
</tr>
<tr>
<td>SYSIN</td>
<td>Temporary work file used as IOCP input deck</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDPROF</td>
<td>HCD profile (when generating additional information for extended migration)</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

**Example 1:**

The following example shows a *Build IOCDS* job.

```plaintext
//BWINJOB JOB (3259,RZ-28),'BWIN',NOTIFY=BWIN,CLASS=A,
// MSGCLASS=Q,MSGLEVEL=(1,1),REGION=5M
//*
//** BUILD IOCDS (WITH DUAL-WRITE OPTION)
//**
//IOCDS EXEC PGM=CBDMGHCP,
// PARM='IOCDS,A0,PROC1,W'
//HCDIODFS DD DSN=BWIN.IODFA3,DISP=OLD
//HCDMLOG  DD DSN=BWIN.HCD.LOG,DISP=OLD
//SYSPRINT DD DSN=&&IOCPOUT,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=6650),
// SPACE=(CYL,(1,1)),DISP=(NEW,PASS,DELETE),UNIT=SYSALLDA
//SYSIN DD DSN=&&TEMP,DISP=(NEW,DELETE),SPACE=(CYL,(1,1)),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),UNIT=SYSALLDA
//HCDNTL DD *
//IOCDSNAM
//** EXECUTED ONLY IF RETURN CODE HIGHER THAN 0
//PRINT EXEC PGM=IEBGENER,COND=(0,EQ,IOCDS)
//SYSUT1 DD DSN=&&IOCPOUT,DISP=(OLD,DELETE)
//SYSUT2 DD SYSUT2
//SYSIN DD DUMMY
//SYSPRINT DD DUMMY
//**
//CLEANUP EXEC PGM=IEFBR14,COND=(0,NE,IOCDS)
//SYSUT DD DSN=&&IOCPOUT,DISP=(OLD,DELETE)
//*
```

*Figure 130. Build IOCDS job*

**Example 2:**

...
The following example shows a **Build IOCP input data set** job.

```plaintext
//BWINJOB  JOB (3259,RZ-28), 'BWIN', NOTIFY=BWIN, CLASS=A,
//         MSGCLASS=Q, MSGLEVEL=(1,1), REGION=5M
// *
//* BUILD IOCP DECK
//*
//IOCP    EXEC PGM=DBDMGHCP,
//        PARM='IOCDS,,PROC1,D,SA'
//HCDDIODFS DD DSN=BWIN.IODFA3,DISP=SHR
//HCDDDECK DD DSN=BWIN.IOCP3.DECK,DISP=OLD
//HCDDMLEG DD DSN=BWIN.HCD.LOG,DISP=OLD
//HCDDCNTL DD *
//IOCDSNAM
/*
// Figure 131. Build IOCP input data set

**Note:** HCDDCNTL specifies the value of the MSG1 parameter (in the example: IOCDSNAM) which is the identification information printed on the first ID line of the heading of the IOCP input data set. Specify the text without any keyword and quotation-marks. The first eight characters are used as IOCDS name.

**Build an HCPRIO input data set**

This utility function creates an HCPRIO input data set using the definitions of a VM operating system in a production IODF.

The IODF from which I/O definitions are extracted has to be specified with DD name HCDIODFS, and the HCPRIO input data set with DD name HCDDECK.

This function is invoked by passing the following parameter string.

```

```

**Build an HCPRIO input data set**

```

```

**H** Function indicator for HCPRIO input data set

**osid**

VM operating system ID (up to 8 characters)

**Batch invocation**

A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFS</td>
<td>IODF from which I/O definitions are extracted</td>
</tr>
<tr>
<td>HCDDECK</td>
<td>VM I/O configuration data set</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

**Example:**
Build I/O configuration data

This utility function allows you to build I/O configuration statements from an IODF and to store them in a data set. The statements describe:
- Operating system configurations
- Processor configurations
- Switch configurations

The data sets created can be edited and re-migrated into the IODF.

In addition, you can build:
- JES3 initialization stream checker data
- CONFIGxx members (from production IODF only)
- FCP device data

You invoke this function by passing the following parameter string:

```
CONFIG , OS , osid
PR , procid
SW , swid
FCP , procid
JES , osid , edt_id
XX , CONFIGxx_parameters
```

**Function indicator:**
- **OS** Build OS configuration statements
- **osid** OS configuration ID (up to 8 characters)
- **PR** Build processor configuration statement

**CONFIGxx_parameters:**
- **xx**
- **procid**
- **partition_id**
- **osid**
- **backup**
- **sysplex**
**procid**  Processor ID (up to 8 characters)
*  If you specify * in place of a processor ID, HCD searches for a list of processor IDs in a data set allocated to HCDCNTL. If no data set has been allocated to DD name HCDCNTL, statements are built for all processors in the IODF.

If only the coupling facility partition and CF receive channels are to be generated for a particular processor, the respective processor ID in the list has to be qualified by the keyword CF, according to the following syntax:

```
procid
```

**SW**  Build switch configuration statements

**swid**  Switch ID (2 hexadecimal characters)
*  If you specify * in place of a switch ID, HCD searches for a list of switch IDs in a data set allocated to HCDCNTL. If no data set has been allocated to DD name HCDCNTL, statements are built for all switches in the IODF.

**FCP**  Export FCP device data into CSV output format

**procid**  Processor ID (up to 8 characters) for which to export the FCP device data.

**JES**  Build JES3 initialization stream checker data

**osid**  OS configuration ID (up to 8 characters)

**edt_id**  EDT ID (2 hexadecimal characters)

**XX**  Build CONFIGxx members

**xx**  Suffix of the CONFIGxx member to be built

**procid**  Processor ID (up to 8 characters)

**partition_id**  Partition name; required entry, if the processor is in LPAR mode (up to 8 characters).

**osid**  OS configuration ID (up to 8 characters)

**U/R**  
U updates the current CONFIGxx member. The CHP and DEVICE statements are replaced and all other statements remain unchanged. This is the default.

R deletes the current CONFIGxx member and generates new CHP and DEVICE statements.

**backup**  Name for the backup copy of the current CONFIGxx member (up to 8 characters).

**sysplex**  Name of the sysplex used for setting managed channel paths to ONLINE.

**Batch invocation**

A data set must be allocated to the following DD names when invoking the batch utility.

---

Chapter 12. How to invoke HCD batch utility functions  321
<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFS</td>
<td>Source IODF</td>
</tr>
<tr>
<td>HCDDECK</td>
<td>Generated output data set</td>
</tr>
<tr>
<td></td>
<td>For CONFIGxx this must be a data set name of a partitioned data set.</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDNTL</td>
<td>Optional for specifying a list of operating systems, processors, or switches</td>
</tr>
<tr>
<td></td>
<td>Not applicable for building CONFIGxx member or JES3 init data set</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Optional for capturing the trace if trace is activated.</td>
</tr>
</tbody>
</table>
Example 1:

The following example shows a job to build a configuration data set containing 
processor configuration PROC1 including its CF connections to processor PROC2.

```
//BWINJOB JOB (3259,RZ-28), 'BWIN', NOTIFY=BWIN, CLASS=A, 
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M
/*
/* BUILD Processor configuration statement
/*
//BUILD EXEC PGM=CBDMGHCP, PARM='CONFIG,PR,*'
//HCDIODFS DD DSN=BWIN.IODF03, DISP=SHR
//HCDDECK DD DSN=BWIN.IODF.03.DECKS(PROC1), DISP=SHR
//HCDMLOG DD DSN=BWIN.HCD.LOG, DISP=OLD
//HCDCNTL DD *
PROC1
PROC2, CF
/*
/*
```

Example 2:

The following example shows a job to update CONFIG03 in data set 
SYS1.PARMLIB from processor configuration PROC1, partition LPAR1 and OS 
configuration MVS1 while saving the existing member under the name 
CONFBK03.

```
//BWINJOB JOB (3259, RZ-28), 'BWIN', NOTIFY=BWIN, CLASS=A, 
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M
/*
/* BUILD CONFIGxx
/*
//BUILD EXEC PGM=CBDMGHCP, PARM='CONFIG,XX,03,PROC1,LPAR1,MVS1,U,CONFBK03'
//HCDIODFS DD DSN=BWIN.IODF03, DISP=SHR
//HCDDECK DD DSN=SYS1.PARMLIB, DISP=SHR
//HCDMLOG DD DSN=BWIN.HCD.LOG, DISP=OLD
/*
```

Copy IODF

This utility function copies the content of the IODF, addressed by DD name 
HCDIODFS, into another data set, addressed by DD name HCDIODFT. If the IODF 
has an associated activity log, that log is also copied. Likewise, if an HCM master 
configuration file (MCF) is associated to the IODF, it is also copied along with the 
IODF. However, a change log file (CHLOG), if available, is not copied.

This function is invoked by passing the following parameter string.

```
Copy an IODF
```

Notes:
1. The target data set must be large enough to hold the source IODF.
2. The target IODF can be created by defining a VSAM DIV file and by 
   initializing it using the utility Initialize IODF (described in "Initialize IODF" 
   on page 309).
3. Both data sets must be valid IODFs.
4. If you copy an IODF which is enabled for multi-user access, this property is not inherited by an existing target IODF. However, a target IODF defined with the multi-user access property will always preserve this property, independent from the source IODF.

**Batch invocation**

A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFS</td>
<td>Source IODF</td>
</tr>
<tr>
<td>HCDIODFT</td>
<td>Target IODF</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

For an example see the batch job on page 335.

**Print configuration reports**

This HCD batch utility function allows you to print reports about:

- The hardware definitions stored in the specified IODF.
- The I/O paths of an actual system compared to the definitions in the IODF.
- The processors, control units, and devices supported in your installation.

This function is invoked by passing the following parameter string.

```
Print a configuration report

- REPORT, C, P, U, D, S, M, E, N, T, I, X, Y
- procid, portnm, osid, swid, system, sysplex, XML

C | P | U | D | S | M | E | N | T | I | X | Y
Type of the report. Specify one or more of the following codes in any order, with no separating characters:
C  CSS report - CSS summary report only
P  CSS report - channel path detail report
U  CSS report - control unit detail report
```
procid  Processor ID to limit a CSS, CTC connection or I/O Path report to a specific processor. If not specified for an I/O Path report, the ID of the active processor configuration is taken (=default).

partnm  Partition name to limit a CSS, CTC connection or I/O Path report to a specific logical partition. The processor ID must also be specified; otherwise, the partition name is ignored for the CSS and CTC connection report. For the I/O Path report, the partition name is defaulted to a partition that contains a device common to the specified or defaulted OS configuration.

osid  Operating system configuration ID to limit an OS report or an I/O Path report to a specific operating system configuration. If not specified for an I/O Path report, the ID of the active operating system configuration is taken (=default).

swid  Switch identifier to limit a Switch report to a specific switch

system  If the sysplex name is also specified, the system name (1 - 8 alphanumeric characters) identifies the system of a sysplex for that the I/O Path report is to be generated. If the sysplex is not specified, the system name is the VTAM application name of the host for that the I/O Path report is to be generated. The default is the local system.

sysplex  Sysplex name (1 - 8 alphanumeric characters) to specify the sysplex of the system for which the I/O Path report is to be generated. If you specify the sysplex, you must also specify the system name.

XML  Specify the XML keyword if you want to print your report in XML output format.

Note: It is recommended to print the I/O Path report separately from the other reports. However, if you want to print an I/O Path report together with any other reports, your limitations for the I/O Path report are propagated to all other specified reports. These limitations to a certain processor, partition or operating system can be either user-specified or HCD takes the active processor, partition or operating system as the default.

Batch invocation
A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>SYS1.SCBDHENU (required for I/O definition reference)</td>
</tr>
<tr>
<td>HCDIODFS</td>
<td>Source IODF (not required when printing the supported hardware report, and I/O definition reference)</td>
</tr>
</tbody>
</table>
DD name | Description
--- | ---
HCDRPT | Output data set: record size 133, record format fixed block
HCDMLOG | HCD Message Log data set
HCDTRACE | Trace data set (if trace is activated)

Example 1:
```//BWINJOB JOB (3259,RZ-28),'BWIN',NOTIFY=BWIN,CLASS=A,
// MSGCLASS=Q,MSGLEVEL=(1,1),REGION=4M
//*****************************************************************************/
//** PRINTS A CSS SUMMARY REPORT FOR PARTITION PART1 OF PROCESSOR
//** PROC1,
//** A SWITCH REPORT FOR SWITCH 00,
//** AN OS REPORT FOR DEVICES, EDT AND NIP CONSOLES OF
//** OS CONFIGURATION MVS1
//*****************************************************************************/
//REPORT1 EXEC PGM=CBDMGHCP,
// PARM='REPORT,CSMEN,PROC1,PART1,MVS1,00'
//HCDIODFS DD DSN=BWIN.IODF03.WORK,DISP=SHR
//HCDRPT DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=6650)
//HCDMLOG DD DSN=BWIN.HCD.LOG,DISP=OLD
//```

Example 2:
```//BWINJOB JOB (3259,RZ-28),'BWIN',NOTIFY=BWIN,CLASS=A,
// MSGCLASS=Q,MSGLEVEL=(1,1),REGION=4M
//*****************************************************************************/
//** PRINTS A SUPPORTED HARDWARE REPORT AND
//** AN I/O DEFINITION REFERENCE
//*****************************************************************************/
//REPORT2 EXEC PGM=CBDMGHCP,
// PARM='REPORT,XY'
//STEPLIB DD DSN=SYS1.SCBDHENU,DISP=SHR
//HCDRPT DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=6650)
//HCDMLOG DD DSN=BWIN.HCD.LOG,DISP=OLD
//```

Example 3:
```//BWINJOB JOB (3259,RZ-28),'BWIN',NOTIFY=BWIN,CLASS=A,
// MSGCLASS=Q,MSGLEVEL=(1,1),REGION=4M
//*****************************************************************************/
//** PRINTS AN I/O PATH REPORT OF THE ACTIVE CONFIGURATION
//** COMPARED TO THE DEFINITIONS IN IODF SYS1.IODF00
//*****************************************************************************/
//REPORT3 EXEC PGM=CBDMGHCP,
// PARM='REPORT,I'
//HCDIODFS DD DSN=SYS1.IODF00,DISP=SHR
//HCDRPT DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=6650)
//HCDMLOG DD DSN=BWIN.HCD.LOG,DISP=OLD
//```

Example 4:
```//BHEIREP JOB (DE03243,,RZ-29),'HEISSER',CLASS=A,REGION=4M,
// MSGLEVEL=(1,1),NOTIFY=BHEI,MSGCLASS=Q
//*****************************************************************************/
//**EXAMPLE OF A BATCH JOB THAT IS CREATED FROM THE DIALOG AND THAT
//**PRINTS AN I/O PATH REPORT OF THE ACTIVE CONFIGURATION OF THE
//**SYSTEM HCDTST3 COMPARED TO THE DEFINITIONS FOR THE PROCESSOR
//**VMABASIC AND THE OPERATING SYSTEM MVSV3 IN BHEI.IODF01.WORK
//*****************************************************************************/
```

326 z/OS V1R11.0 HCD User's Guide
Note: For generating the I/O Path Reports which are printed in examples 3 and 4, System Automation for z/OS (I/O Operations) or the ESCON Manager is required.

Create a graphical configuration report

This utility function allows you to produce a graphical representation of the I/O configuration based on the definitions in the IODF.

This function is invoked by passing the following parameter string.

```
GRAPHIC TYPE= CU
      SWITCH
      CF
      CHPID , PROC= procid , PART= partname
      LCU , PROC= procid , PART= partname

Options

, OPT= ( EP )
      ES
      EC
      SD
      EI
      DCF
      GML

TYPE Type of the report. Specify one of the following codes:
CU CU report
SWITCH Switch report
CF CF connection report
CHPID CHPID report
LCU LCU report
Procid Processor ID for which the LCU or CHPID report is produced.
Partname Partition name to limit an LCU or CHPID report to one partition.

OPT Options of the report. Specify one or more of the following codes in any order, separated by a comma:
EP Exclude partition
ES Exclude switch
```
EC  Exclude CTC control units
SD  Show control unit description (instead of serial number)
EI  Exclude index
DCF DCF output format
GML GML output format

Notes:
1. If no output format is specified, the specification in the HCD profile is used. If the HCD profile does not specify a formatting type either, the default BookMaster format is used.
2. The output format GDF is not supported in batch mode.

Batch invocation
A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFS</td>
<td>Source IODF</td>
</tr>
<tr>
<td>HCDRPT</td>
<td>Output data set: record size 200, record format FB.</td>
</tr>
<tr>
<td></td>
<td>Note: This must be a cataloged data set.</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message log data set</td>
</tr>
<tr>
<td>HCDPROF</td>
<td>HCD profile data set (if profile contains keywords concerning the graphical report)</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

Example:
//BWINGCR1 JOB (DE3259,,71034-83),BWIN,NOTIFY=BWIN,CLASS=A,MSGCLASS=Q,MSGLEVEL=(1,1),REGION=4M
//* --------------------------------------------------------
//* Graphical Configuration Report
//* --------------------------------------------------------
//GCREP EXEC PGM=CBDMGHCP,PARM='GRAPHIC TYPE=CHPID,PROC=TEST3'
//HCDIODFS DD DSN=USER.IODF00.DBR4,DISP=SHR
//HCDRPT DD DSN=USER.IODF00.DBR4.REPORT,DCB=(RECFM=FBA,LRECL=200,BLNKSIZE=6400),SPACE=(TRK,(50,50)),DISP=(NEW,KEEP),UNIT=SYSALLDA
//HCDMLOG DD DSN=USER.HCD.LOG,DISP=OLD
//HCDPROF DD DSN=USER.HCD.PROF,DISP=OLD

Compare IODFs or CSS/OS Reports
This utility function allows you to compare two IODFs and report the differences. You can compare the IODFs from the CSS, OS, and switch perspective.

In addition, you can limit the CSS, OS, and switch perspective by single compare reports, and the CSS perspective by LPARs.

You have to allocate the new IODF with DD name HCDIODFS, and the old IODF with DD name HCDIODFT for comparing IODFs. If you compare the CSS to the OS definition, you always compare within one IODF that must be allocated to HCDIODFS.

You invoke this function by passing the following parameter string.
Note: This parameter string must not exceed 100 characters.

Compare IODFS or CSS/OS Reports

Print Options for IODF Compare Report:

Print Options for CSS/OS Compare Report:

Notes:
1 Required if processor runs in LPAR mode.

Print options for IODF Compare Report
Specify one or more of the following print options, without separating characters, in exactly this order. For more information about the print option types, see "How to print a Compare IODFs Report" on page 249.

A Print added data
B Print deleted data
C Print unchanged data
D Print unchanged item IDs
CL | C | OL | O | SL | S

Type of the report. Specify one or more of the following codes in any order, with no separating characters:

CL   Limit the CSS compare report by single compare reports and LPARs

   PR   Processor compare
   CS   Channel subsystem compare
   PA   Partition compare
   CP   Channel path compare
   CA   Control unit attachment compare
   DA   Device attachment compare
   CU   Control unit compare
   DV   Device compare

procid1
New processor ID

partn1 Partition name of the new processor

cssid1 selected CSS ID of the new XMP processor, either appended by a # and in parenthesis or appended by a dot (.). If the CSS ID is specified for one processor only (old or new), the CSS ID for the other processor is defaulted to CSS ID 0.

procid2
Old processor ID

partn2 Partition name of the old processor

cssid2 selected CSS ID of the old XMP processor, either appended by a # and in parenthesis or appended by a dot (.)

C Indicates CSS compare report

procid1
New processor ID

cssid1 selected CSS ID of the new XMP processor, either appended by a # and in parenthesis or appended by a dot (.). If the CSS ID is specified for one processor only (old or new), the CSS ID for the other processor is defaulted to CSS ID 0.

procid2
Old processor ID

cssid2 selected CSS ID of the old XMP processor, either appended by a # and in parenthesis or appended by a dot (.)

OL Limit OS compare report by single compare reports

   OS   Operating system compare
   ED   EDT compare
   GE   Generic compare
   GU   Generic update compare
   ES   Esoteric compare
   NI   OS console compare
   OD   OS device compare
osid1  New operating system ID
osid2  Old operating system ID
O  Indicates OS compare report
osid1  New operating system ID
osid2  Old operating system ID
SL  Limit switch compare report by single compare reports

SW  Switch compare
PO  Switch port compare
SC  Switch configuration compare
PC  Port configuration compare

swid1  New switch ID
swid2  Old switch ID

S  Indicates SWITCH compare report
swid1  New switch ID
swid2  Old switch ID

Print options for CSS/OS Compare Report
C  Print all devices. If C is not selected, only devices are printed that
are
• Defined for the CSS, but not for the OS
• Defined for the OS, but not for the CSS
• Defined for both, but of different device type
D  Indicates CSS/OS compare
procid  Processor ID
partn  Partition name. This is a required parameter, if the processor runs
in LPAR mode. For more information, see "Compare CSS /
operating system views" on page 250.
osid  Operating system ID

Batch invocation
A data set must be allocated to the following DD names when invoking the batch
utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>SYS1.SCBDHENU (required for OS device compare)</td>
</tr>
<tr>
<td>HCDIODFS</td>
<td>New IODF</td>
</tr>
<tr>
<td>HCDIODFT</td>
<td>Old IODF (only for IODF compare)</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDRPT</td>
<td>Report data set; record size 133, record format fixed block</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

Example 1:
The following example shows a job to compare two IODFs.

```
//BWINJOB JOB (3259,RZ-28), 'BWIN', NOTIFY=BWIN, CLASS=A,
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M
//*
//* COMPARE IODFs WITH ADDED AND DELETED DATA
```
Example 2:
The following example shows a job to **compare CSS/OS reports**.

```bash
//BWINJOB JOB (3259,RZ-28),'BWIN',NOTIFY=BWIN,CLASS=A,
// MSGCLASS=Q,MSGLEVEL=(1,1),REGION=4M
/*
/* COMPARE CSS/OS CONFIGURATION BETWEEN
/* DEVICES CONNECTED TO PROC1, PART1 ON CSS SIDE
/* DEFINED TO OS MVSI
/*
/*COMPARE2 EXEC PGM=CBDMGHCP,
// PARM='COMPARE,AB,D,PROC1,PART1,MVSI'
//HCDIODFS DD DSN=BWIN.IODF06.WORK,DISP=SHR
//HCDRPT DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BKSIZE=6650)
//HCDMLOG DD DSN=BWIN.HCD.LOG,DISP=OLD
/*
```

**Import an IODF**

This utility function allows you to import configuration data (previously exported from another system) into an IODF. It is assumed that the mentioned configuration data has been received outside HCD, for example, using the TSO RECEIVE command, and stored in a sequential data set.

The data set containing IODF data to be imported has to be specified with DD name HCDIN.

This function is invoked by passing the following parameter string.

```
Import an IODF

IMPORT, IODF name, volume, REPLACE
```

**IODF name** Specifies the name of the target IODF (fully qualified).

**volume** Specifies the volume serial number of the IODF destination. This parameter is neglected if the target IODF already exists and REPLACE is specified, or, if the data set is SMS managed.

**REPLACE** Specifies that an IODF with the same name will be replaced by the received IODF. If REPLACE is not specified, the IODF is not replaced.
Batch invocation

A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIN</td>
<td>The data set containing IODF data to be imported</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

Example:

//BWINJOB JOB (3259,RZ-28), 'BWIN', NOTIFY=BWIN, CLASS=A, 
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M 
//* 
//* IMPORT AN IODF 
//* 
//IMPORT EXEC PGM=CBDMGHCP, 
// PARM='IMPORT,BWIN.IODF08,DATA04' 
//HCDIN DD DSN=BWIN.EXPORTED.IODF03, DISP=SHR 
//HCDMLOG DD DSN=BWIN.HCD.LOG, DISP=OLD 
//

Export an IODF

This utility function allows you to send an IODF to another system.

You have to preallocate the IODF you want to export with DD name HCDIODFS.

If you want to send an IODF to an unattended z/OS system, you have to allocate a data set with DD name HCDCNTRL. From this data set, HCD extracts information to set up the JCL to run on the unattended target system. You have to modify or adapt the JOB statement, JES routing statement(s), and JOBLIB information in this data set before you call the export utility.

This function needs to be executed in an TSO environment, for example, by invoking HCD under control of the TSO terminal monitor program IKJEFT01.

This function is invoked by passing the following parameter string.

Export an IODF

---

user id

node id

IODF name

volume

ACTLOG

NOREPLACE

---

user id    Specifies the user ID of the target, or a nickname (nickname only if the IODF is not sent to an unattended z/OS system).

node id    Specifies the node ID of the target system, but only if the IODF is not sent to an unattended target system (otherwise it is ignored in favor of information provided by the JCL).

IODF name  Specifies the name of the target IODF. Default is the name of the source IODF prefixed with the specified target user ID. This
parameter is only applicable if the IODF is sent to an unattended z/OS system, otherwise it is ignored.

**volume**  
Specifies the volume serial number of the DASD on which the target IODF is created if it does not exist. This parameter is only applicable if the IODF is sent to an unattended z/OS system, and the IODF data set is not managed by SMS, otherwise it is ignored.

**ACTLOG**  
Specifies that the appropriate Activity Log file should also be sent. If this parameter is missing, or the target is an unattended MVS system, it is not sent.

**NOREPLACE**  
For unattended exports, this keyword provides overwrite protection for an IODF at the target system with the same name as the IODF to be exported.

**Batch invocation**  
A data set must be allocated to the following DD names when invoking the batch utility.

<table>
<thead>
<tr>
<th>DD name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCDIODFS</td>
<td>IODF to be exported</td>
</tr>
<tr>
<td>HCDCNTL</td>
<td>JCL data set containing the JOB statement, the JES routing statement(s) and the JOBLIB information for sending the IODF to an unattended z/OS system.</td>
</tr>
<tr>
<td>SYSTSPRT</td>
<td>Print data set</td>
</tr>
<tr>
<td>SYSTSIN</td>
<td>SYSIN data set</td>
</tr>
<tr>
<td>HCDMLOG</td>
<td>HCD Message Log data set</td>
</tr>
<tr>
<td>HCDTRACE</td>
<td>Trace data set (if trace is activated)</td>
</tr>
</tbody>
</table>

You can also find the following examples in the data set SYS1.SAMPLIB.

**Example 1:**

The following example shows a job to export an IODF to an attended system.

```
//BWINEX1 JOB (3259,7030-83), 'BWIN',CLASS=A,USER=BWIN,  
// MSGLEVEL=(1,1),NOTIFY=BWIN,MSGCLASS=Q,REGION=4M  
/*/  
//EXPORT1 EXEC PGM=IKJEFT01  
//SYSPRINT DD SYSPUT=+  
//HCDIODFS DD DSN=BWIN.IODF52.WORK,DISP=SHR  
//HCDMLOG DD DSN=BWIN.HCD.LOG,DISP=OLD  
//SYSTSPRT DD SYSPUT=+  
//SYSTSIN DD *  
    CALL 'SYS1.LINKLIB(CBDMGHCP)', +  
    'EXPORT,BMGN,BOETST1'  
/*/  
//
```

**Example 2:**

The following example shows a job to export an IODF to an unattended z/OS system.

```
//BWINEX2 JOB (3259,7030-83), 'BWIN',CLASS=A,USER=BWIN,  
// MSGLEVEL=(1,1),NOTIFY=BWIN,MSGCLASS=Q,REGION=4M  
//************************************************************  
//* MODIFY AND ADAPT DATA SET ALLOCATED WITH DDNAME HCDCNTL  
```
Example 2.1:

The following example shows the JCL statements that may be specified in a data set allocated with DD name HCDCNTL for a JES3 system.

```jcl
//BWINEX1 JOB (3259,7030-83), 'BWIN', CLASS=A, NOTIFY=BWIN,
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M
//*
//* JCL STATEMENTS SPECIFIED WITH DDNAME HCDCNTL
//*
//*ROUTE XEQ BOETST1
//BBMGNIM JOB (3259,7030-83), 'BMGN', CLASS=A,
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M,
// USER=BMGN, PASSWORD=password
//OUT1 OUTPUT JESDS=ALL, DEFAULT=YES, DEST=BOEVS01.BWIN
```

Example 2.2:

The following example shows the JCL statements that may be specified in a data set allocated with DD name HCDCNTL for a JES2 system.

```jcl
//BBMGNIM JOB (3259,7030-83), 'BMGN', CLASS=A,
// MSGCLASS=Q, MSGLEVEL=(1,1), REGION=4M,
// USER=BMGN, PASSWORD=password
//*ROUTE XEQ BOETST1
```

**Note:** You can replace the /*ROUTE statement by the /*XMIT statement.

```jcl
//XMIT XEQ BOETST1 DLM=xx
```

When you use the DLM parameter with the /*XMIT statement, you specify a two-character delimiter to terminate the data being transmitted. (For the end of the records to be transmitted, the default is /* in the input stream.)

---

**Batch IODF copy example**

You may want to use the batch initialize and copy IODF functions when performing system maintenance. For example, these functions can be used when copying all the data sets from one volume (that contains IODFs) to another volume.

The following sample jobs show how to copy multiple IODFs to an alternate volume. In the example it is assumed that the following IODFs exist: SYS1.IODF00 and SYS1.IODF03. The first sample job allocates corresponding data sets on volume DATA02. The second sample job initializes these data sets into IODFs, and copies
the source IODFs into the newly created IODFs. Jobs similar to these are contained in member CBDSALIO and CBDSCPLO in SYS1.SAMPLIB.

```* JOB TO DEFINE IODF
//DEFIODF JOB REGION=4M,...
*
// DEFINE NEW IODF DATASETS SYS2.IODF00, SYS2.IODF03
*
//ALLOC EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=* 
//SYSIN DD *

DEFINE CLUSTER (NAME (SYS2.IODF00.CLUSTER) - 
  LINEAR -
  RECORDS (1024) -
  VOLUMES(DATA02) -
) -
DATA (NAME (SYS2.IODF00))

DEFINE CLUSTER (NAME (SYS2.IODF03.CLUSTER) - 
  LINEAR -
  RECORDS (1024) -
  VOLUMES(DATA02) -
) -
DATA (NAME (SYS2.IODF03))
```

```* JOB TO COPY IODF
//INITIODF JOB REGION=4M,...
*
// INITIALIZE AND COPY SYS1.IODF00 to SYS2.IODF00
*
//INIT1 EXEC PGM=CBDMGHCP,PARM='INITIODF SIZE=1024,ACTLOG=NO'
//HCDCNTL DD *
This IODF is a copy of SYS1.IODF00
/
//HCDDODFT DD DSN=SYS2.IODF00,DISP=OLD 
//HCDDMOLOG DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=6650)
/*/ 
//COPY1 EXEC PGM=CBDMGHCP,PARM='COPYIODF'
//HCDDODFS DD DSN=SYS1.IODF00,DISP=SHR
//HCDDODFT DD DSN=SYS2.IODF00,DISP=OLD
//HCDDMOLOG DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=6650)
/*/ 
// INITIALIZE AND COPY SYS1.IODF03 to SYS2.IODF03
/*/ 
//INIT2 EXEC PGM=CBDMGHCP,PARM='INITIODF SIZE=1024,ACTLOG=NO'
//HCDCNTL DD *
This IODF is a copy of SYS1.IODF03
/
//HCDDODFT DD DSN=SYS2.IODF03,DISP=OLD
//HCDDMOLOG DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=6650)
/*/ 
//COPY2 EXEC PGM=CBDMGHCP,PARM='COPYIODF'
//HCDDODFS DD DSN=SYS1.IODF03,DISP=SHR
//HCDDODFT DD DSN=SYS2.IODF03,DISP=OLD
//HCDDMOLOG DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=6650)
*/
```

**Notes:**

1. This batch job issues job message IEC161I, which can be ignored.
2. The VSAM DIV file consists of a data and a cluster file. According to the naming convention, the name of the data file is the IODF name (in this example SYS1.IODF01), and .CLUSTER is appended to the data file for a cluster file. Append .CLUSTER to the IODF name in the DEFINE CLUSTER statement (in this example SYS1.IODF01.CLUSTER).
When designating the number of records to be allocated in an IODF (specified on the DEFINE CLUSTER statement and as a PARM value on the INITIODF job step), it is important that the target IODF be allocated at least as big as the source IODF. While using the HCD dialog, you can use the SHOWIODF command from the command line, or the View action bar choice to display the number of records allocated in the source IODF.

Note: Remember to also copy the associated LOADxx members. For information on the LOADxx members, see z/OS MVS Initialization and Tuning Reference and z/OS MVS Initialization and Tuning Guide.

List of standard DD names

Standard DD names are used in the job control statements that define the data sets used by HCD. These names are shown in Table 8. If you want to change these names, you must create a list of alternate DD names, using the standard format for such a list.

Table 8. Standard DD names Used by HCD

<table>
<thead>
<tr>
<th>DD name</th>
<th>HCD Task</th>
<th>Data Set Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>not used</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>not used</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>not used</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>HCDLIB</td>
<td>Migration Assembler macro library</td>
</tr>
<tr>
<td>5</td>
<td>HCDIN</td>
<td>Migration Migration input</td>
</tr>
<tr>
<td>6</td>
<td>HCDPRINT</td>
<td>Migration Migration log (messages)</td>
</tr>
<tr>
<td>7</td>
<td>not used</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>HCDUT1</td>
<td>Migration Assembler work file</td>
</tr>
<tr>
<td>9</td>
<td>HCDUT2</td>
<td>Migration Modified IOCP, MVSCP, and HCPRIO input to assembler</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Activity Log Target activity log during copy</td>
</tr>
<tr>
<td>10</td>
<td>HCDUT3</td>
<td>Migration Assembler output (object) data</td>
</tr>
<tr>
<td>11</td>
<td>not used</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>HCDTERM</td>
<td>Migration Assembler and loader messages</td>
</tr>
<tr>
<td>13</td>
<td>not used</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>not used</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>not used</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>not used</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>HCDRPT</td>
<td>Query/print HCD reports</td>
</tr>
<tr>
<td>18</td>
<td>HCDALOG</td>
<td>All Activity log</td>
</tr>
<tr>
<td>19</td>
<td>HCDJES3</td>
<td>All JES3 initialization stream checker data</td>
</tr>
<tr>
<td>20</td>
<td>HCDASMP</td>
<td>Migration Assembler output listing</td>
</tr>
<tr>
<td>21</td>
<td>HCDDECK</td>
<td>Activation IOCP and HCPRIO input data set (output)</td>
</tr>
<tr>
<td>22</td>
<td>HCDIODFP</td>
<td>All First IODF</td>
</tr>
<tr>
<td>23</td>
<td>HCDIODFS</td>
<td>Maintain IODF Source IODF (for COPY, for example)</td>
</tr>
<tr>
<td>24</td>
<td>HCDIODFT</td>
<td>Maintain IODF Target IODF</td>
</tr>
</tbody>
</table>
Table 8. Standard DD names Used by HCD (continued)

<table>
<thead>
<tr>
<th>DD name</th>
<th>HCD Task</th>
<th>Data Set Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>HCDPROF</td>
<td>Tailor HCD defaults HCD profile definitions</td>
</tr>
<tr>
<td>26</td>
<td>HCDMLOG</td>
<td>All Message log</td>
</tr>
<tr>
<td>27</td>
<td>HCDTRACE</td>
<td>All Trace data set (if trace is activated)</td>
</tr>
<tr>
<td>28</td>
<td>HCDCTRL</td>
<td>Activation Maintain IODF Control file for Build IOCDS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Control file for Build IOCP input data set</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL data set for Import/Export IODF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IODF description</td>
</tr>
<tr>
<td></td>
<td></td>
<td>List of configurations for</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Build I/O configuration data</td>
</tr>
</tbody>
</table>

**List of alternate DD names**

If used, this optional list, must start on a halfword boundary that is not also a fullword boundary.

- The first two bytes must contain a binary count of the number of bytes in the rest of the list.
- The rest of the list specifies alternate DD names that you wish to use in place of the standard DD names.
  - DD names in the alternate list must appear in the same sequence as they appear in the standard list.
  - Each name must be eight characters long. If a name contains fewer than eight characters, pad it with blanks. If you omit an alternate DD name, set that entry in the alternate DD names list to binary 0.
  - Entries in the alternate DD names list that correspond to empty entries in the standard DD names list must be set to binary 0.

______________ End of Programming Interface information ______________
Chapter 13. Security and other considerations

Overview

This information unit discusses the following topics:

- Security-related considerations
- Catalog-related considerations
- Considerations concerning data sets cataloged with an esoteric device group name
- SMS-related considerations
- ISPF and TSO/E aspects that you need to consider
- VM-related considerations

Security-related considerations

An appropriate resource-level security facility, such as Resource Access Control Facility (RACF®) 1.9 or an equivalent security product, is required to control access to the data sets used by HCD. You perform the access control in two steps:

1. Define the necessary RACF profiles
2. Give users access authority

Note: If no security product is installed, you cannot perform the activate function from HCD.

For additional RACF definitions necessary to run the HCD LDAP Backend against IODF data, see Chapter 14, “How to provide LDAP support for HCD,” on page 347.

Defining RACF profiles

You define three types of profiles:

1. Data Set Profiles
   Define data set profiles for all data sets used by HCD.

2. OPERCMDS Class Profile:
   Define the profile MVS.ACTIVATE to invoke the dynamic reconfiguration function under HCD or to use the MVS operator command ACTIVATE from an MVS console. (For a description of the command syntax, see z/OS MVS System Commands.)

   If you issue the ACTIVATE command, the I/O supervisor calls jobname IEASYSAS stepname IOSAS to assist in the activate procedure. IOSAS requires read access to the IODF data sets. Because the default entry for IOSAS in the Program Properties Table (PPT) is PASS, RACF checking occurs. ICH408I is the result of an ACTIVATE IODF=XX command. To ensure the successful completion of the activate process, you have to choose one of the following alternatives:
   - Place the IOSAS task into the RACF started task table (ICHRIN03) and indicate that the user is authorized.
   - Define the IODF data sets to RACF with UACC=READ.
• Add IOSAS as an entry in the Started Procedures Table with a valid user ID. This user ID must have read access to the SYS1.NUCLEUS and the IODF data sets.

The ACTIVATE command needs UPDATE access, regardless whether the TEST option is specified or not.

You also have to define the profile MVS.DISPLAY.IOS with read access if you wish to work from the sysplex member list to view the active configuration status or to process a CONFIGxx member.

3. FACILITY Class Profiles:

Define the following two profiles:

- CBD.CPC.IPLPARM to query and update the IPLADDR and IPLPARM attribute values of the last IPL, and to be used for next IPL.
- CBD.CPC.IOCDS to query and update IOCDS control information.

**Giving users access authority**

The access authority you can give to a user depends on the profile.

**Access to data set profiles**

You can give READ, UPDATE, or ALTER access to IODFs in general or to a specific IODF.

**Access to profile MVS.ACTIVATE**

You can give UPDATE access to allow the user to activate a configuration change. You can give READ access if you want to restrict the activate function to the test option.

**Access to profile CBD.CPC.IPLPARM**

<table>
<thead>
<tr>
<th>NONE</th>
<th>Indicated that the user is not allowed to query or change the IPLADDR and IPLPARM attribute values. This is also the case if profile CBD.CPC.IOCDS is not defined or RACF is not installed,</th>
</tr>
</thead>
<tbody>
<tr>
<td>READ</td>
<td>Allows the user to query the IPLADDR and IPLPARM attribute values; however changing the IPLADDR and IPLPARM attribute values is not allowed.</td>
</tr>
<tr>
<td>UPDATE</td>
<td>Allows the user to update the IPLADDR and IPLPARM attribute values.</td>
</tr>
</tbody>
</table>

*Figure 132* shows the relationship between HCD IPL attribute management functions and the CBD.CPC.IPLPARM access authority. Option 2.11 in the figure refers to option 2 on the primary selection panel and option 11 on the resulting panel.

<table>
<thead>
<tr>
<th>Option</th>
<th>HCD IPL Attribute Management Functions</th>
<th>RACF Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.11</td>
<td>List S/390 microprocessor cluster</td>
<td>READ (or READ authority in CBD.CPC.IOCDS)</td>
</tr>
<tr>
<td>2.11</td>
<td>View IPL attributes</td>
<td>READ</td>
</tr>
<tr>
<td>2.11</td>
<td>Update NEXT IPL attributes</td>
<td>UPDATE</td>
</tr>
</tbody>
</table>

*Figure 132. CBD.CPC.IPLPARM access authority and HCD IPL attribute management functions*
Access to profile CBD.CPC.IOCDS

If profile CBD.CPC.IOCDS is not defined or RACF is not installed, the local IOCDS functions (that is for processors with no SNA address specified) work as before, that is, the operator will be requested to approve the write-IOCDS request.

The new remote IOCDS functions (that is for processors with an SNA address specified) require RACF authorization.

NONE The user is not allowed to query or change IOCDS control information, or to write an IOCDS (neither by HCD nor IOCP).

READ Allows the user to query IOCDS control information. Changing IOCDS control information or writing an IOCDS is not allowed (neither by HCD nor IOCP).

UPDATE Allows the user to write IOCDSs (by HCD or IOCP), or to change and view IOCDS control information. If profile CBD.CPC.IOCDS is defined, then the operator will not be requested to approve the writing of an IOCDS. (That is, only users with update access to profile CBD.CPC.IOCDS are allowed to write an IOCDS.)

Table 9 shows the relationship between IOCDS management functions and the CBD.CPC.IOCDS access authority. The first column in the table refers to the options you have to select to get to the HCD functions, that is, you start with option 2 on the primary selection panel and select options 2, 6, or 11 on the resulting panel.

Table 9. CBD.CPC.IOCDS access authority and HCD IOCDS management functions

<table>
<thead>
<tr>
<th>Option</th>
<th>HCD IOCDS Management Functions</th>
<th>RACF Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.11</td>
<td>List S/390 microprocessor cluster</td>
<td>READ (or READ authority in CBD.CPC.IPLPARM)</td>
</tr>
<tr>
<td>2.11</td>
<td>View IOCDS control information</td>
<td>READ</td>
</tr>
<tr>
<td>2.11</td>
<td>Update IOCDS control information (switch IOCDS, enable or disable write protection)</td>
<td>UPDATE</td>
</tr>
<tr>
<td>2.2 or 2.6</td>
<td>Build IOCDS (SNA address not defined for processor or batch IOCP job runs on SP 4.3 system)</td>
<td>UPDATE ¹ or Profile not defined to RACF ²</td>
</tr>
<tr>
<td>2.2 or 2.6 or 2.11</td>
<td>Build IOCDS (SNA address defined for processor and batch IOCP job runs on SP 5.1 system)</td>
<td>UPDATE ¹</td>
</tr>
<tr>
<td>--</td>
<td>Direct invocation of IOCP</td>
<td>UPDATE ¹ or Profile not defined to RACF ²</td>
</tr>
</tbody>
</table>

¹ The build IOCDS function does not require authorization by the system operator, that is, no WTOR message is written.

² A WTOR message will be issued to the operator to authorize the build IOCDS function.

For more information on security considerations for IOCDS management, refer to the IOCP User’s Guide.
Providing additional security for devices

If your system has stringent security requirements and includes Resource Access Control Facility (RACF), you can ensure that only certain programs can allocate unit record, communication, or graphics devices. These programs include Print Services Facility (PSF) for printers, Advanced Communication Facility/Virtual Telecommunications Access Method (ACF/VTAM) for communication or graphics devices, and JES2 or JES3 for unit record, communication, or graphics devices.

When a user attempts to allocate a device, the system uses SAF (the system authorization facility) to issue an authorization check. If RACF is installed, it checks a profile in the DEVICES class to determine whether the user can access the device. If the user does not have authority to access the device, the allocation fails. (Note that the system does not retry an allocation request that fails because the user is not authorized to access the device.)

Work with your RACF security administrator to set up profiles in the DEVICES class:

1. Determine your exact security requirements. Consider questions such as these:
   - Are there some devices that only a few users can use?
   - Are there some devices that all users can use?
   - Do some devices share the same security requirements?

2. Work with your RACF security administrator to assign profile names for the devices to be protected. Assign a discrete profile name to each device that has a unique security requirement. Assign a generic profile name to each device group that shares security requirements. For devices, RACF profile names include the following information:

   **sysid**  This is the system identifier, which is defined on the SYSNAME keyword in the IEASYSxx member of SYS1.PARMLIB.

   **Note:** The system identifier is necessary only if different devices with the same device class, unit name, and device address can be attached to multiple systems and they have different security requirements. In most cases, you should specify an asterisk (*) for this qualifier.

   **device-class**  This can be one of the following UCB device classes:

   - **TP**  Teleprocessing or communications devices
   - **UR**  Unit record devices
   - **GRAPHIC**  Graphic devices. These device classes are consistent with the class names used on the DISPLAY U operator command.

   **unit-name**  This is a generic name (such as 3800) that identifies the device or devices.

For more details, see [z/OS Security Server RACF Security Administrator’s Guide](#).
Catalog considerations

One IODF can contain configuration data mirroring multiple processor or logical partition system images, but being a VSAM data set, it can be cataloged in only one catalog. Therefore, if you wish to share an IODF data set among multiple systems and each system is using a separate master catalog, you must define (in the master catalog of each system) an alias that relates to the user catalog on the DASD that is shared among the systems. Define aliases and the user catalog before using HCD to define IODF data sets. Figure 133 shows the recommended catalog structure.

![Diagram of recommended IODF catalog structure](image)

**Note:** It is useful to catalog the IODF in a user catalog which resides on the same volume as the IODF. That way if the volume fails and must be restored, the catalog/IODF connection is always preserved across the restore. The catalog is used to reference the IODF during HCD definition activities and during dynamic I/O reconfiguration, not during IPL.

**Data sets cataloged with an esoteric device group name**

When using HCD, data sets that were previously cataloged with an esoteric device group name (for example, SYSDA) by use of the DEFINE NONVSAM, or IMPORT CONNECT command of the Integrated Catalog Facility, or the IEHPROGM utility, can cause unpredictable results if such a data set is accessed through the catalog. The reason is that the catalog entry contains the EDT-index pointing to the esoteric. The order of the esoteric in the EDT is no longer determined by the order in which the esoterics are defined, because HCD arranges the esoterics alphabetically.

To avoid this problem, you can do one of the following:
• Specify a token for the esoterics.
The esoteric token will be used by allocation to find the appropriate esoteric for a data set that has been cataloged using the esoteric. You no longer have to maintain a chronological order and may delete and add esoterics without getting access problems for data sets that are cataloged using esoterics. Tokens for system built esoterics (for example, SYSALLDA) are generated by allocation and always have the same value (for SYSALLDA 9999 decimal, for example). You cannot control the token for system built esoterics. To circumvent the problem, define a new user esoteric with a token that corresponds to the EDT index in the catalog entry and that contains the same device list as SYSALLDA.

To get to the EDT index:
– Use the LISTCAT command, or,
– If you have your MVSCP deck, count the UNITNAME statements for esoterics up to the statement that defines the esoteric name to get to the number for the token.

• Re-catalog the data sets with a generic device type name (for example, 3380), before using HCD to migrate IOCP/MVSCP data.

To determine if you have any data sets that have been cataloged with an esoteric, use the scan utility that is provided in the SYS1.SAMPLIB member IEFESOJL. This utility scans a catalog and lists the data sets that were cataloged with esoteric device group names. The prologue of this SAMPLIB member contains information on the modifications you have to make to the JCL to run the job in your installation.

SMS-related considerations

In a system managed by the storage management subsystem (SMS) you need to choose one of the following alternatives:

• The IODF data set is not managed by SMS. You can then specify the IODF volume serial number when creating an IODF.

• The IODF data set is managed by SMS. The automatic class selection (ACS) routines must be set up to automatically place the IODFs on the IODF volume. In this case SMS ignores the specified volume serial number except to pass it as a symbol to the ACS routines. The ACS routines, especially the storage group ACS routine, can use the volume serial number and the unit name to decide the SMS classes and the storage group.

Note: These considerations are important only for a production IODF that is used for IPL.

You also have to consider that HCD dynamically allocates some data sets (with fixed naming conventions). These data sets are:

• The data set used for the activity log. For more information see "Activity logging and change logging" on page 54. If an ACTLOG data set does not yet exist, HCD dynamically allocates one, using ESOTERIC system defaults (ALLOCxx of SYS1.PARMLIB, respectively the UADS entry). You have to make sure that the entries in your ACS routines do not conflict with the SMS provided defaults. For example, if your ACTLOG data set name is not managed by SMS, whereas your default ESOTERIC defines an SMS managed volume, an allocation error might result. If you want to use a specific volume, specify a volume serial number to allocate a new activity log in the HCD profile (see "Defining an HCD profile" on page 23).
The data sets used by HCD for the migration of IOCP/MVSCP/HCPRIO data (HCDPRINT, HCDASMP, HCDUT1, HCDUT2, HCDUT3). For detailed information refer to "Insufficient data set sizes" on page 305.

The data sets used when building an HCPRIO or IOCP input data set (both named HCDDECK), and the data set used when creating JES3 Initialization Stream Checker data (named HCDJES3).

**ISPF-related considerations**

The usual ISPF facilities are available for the HCD dialog. For example, you can:
- Suppress the display of function key assignments.
- Display panel identifiers.
- Change the position of the command line.

HCD supports the ISPF split-screen facility with the F2=Split and F9=Swap keys. So, if necessary, you can perform other ISPF operations during an HCD session. HCD cannot be used in two parallel ISPF sessions.

Compared to traditional ISPF applications, HCD enables system programmers to control a great number of hardware configuration objects by their related actions.

**VM-related considerations**

HCD allows the definition of VM operating systems and their devices including their VM-specific parameters. This is triggered by the "operating system type - VM" when defining an operating system. Figure 134 shows the panel where you can enter the operating system type.

![Figure 134. Define a VM operating system](image)

When you attach a device to a VM operating system, the Define Device Parameters/Features panel displays the operating system-specific parameters. See Figure 135 on page 346 for an example of attaching a device to a VM operating system.
The VM specific functions of HCD consist of:

- Defining an operating system of type 'VM'.
- Defining devices to a VM operating system.
- Defining VM consoles.
- Migrating an HCPRIO input data set to an IODF. For more details refer to Chapter 11, "How to migrate existing input data sets," on page 261.
- Creating an HCPRIO input data set from a production IODF. See "Build I/O configuration data" on page 215 for a description how to create an HCPRIO input data set based on the definitions in the IODF.
- Issuing VM device reports (including VM console report).

In a mixed environment, running a z/OS system in one partition and z/VM in another partition, any change of the VM definitions (for example, add a device) can be done without a POR for the processor. The Dynamic I/O Reconfiguration function for the hardware can be used to add this device to the Channel Subsystem (CSS). In a second step, a device can be set online dynamically. You do not have to IPL the z/VM system image. For more details refer to z/VM CP Planning and Administration.

If you are running z/OS and z/VM on separate processors, you can configure both systems with HCD in the same IODF. You can export the IODF to the z/VM system and activate it there using z/VM HCD. See z/VM: I/O Configuration for more details.
Chapter 14. How to provide LDAP support for HCD

Overview

This information unit describes:
- the LDAP structure
- the structure of the HCD LDAP backend plug-in
- how to plug the HCD LDAP backend into the IBM Tivoli Directory Server for z/OS
- how to customize the HCD LDAP backend
- how to access the IODF information
- the HCD LDAP backend operational behavior
- how to handle the LDAP requests as transactions

Before reading this section we strongly recommend that you have copies of the following documents available:
- IBM Tivoli Directory Server Administration and Use for z/OS
- IBM Tivoli Directory Server Client Programming for z/OS
- z/OS UNIX System Services Planning

Together with the IBM Tivoli Directory Server for z/OS and the RACF backend SDBM, the HCD LDAP backend can be used to access and update IODF data via the standardized Lightweight Directory Access Protocol (LDAP) based on TCP/IP.

The HCD LDAP backend is optional. The HCD functionality is not limited if the HCD LDAP backend is not used. If you do not want to use the HCD LDAP backend, then you do not need to read this information and you do not need to take any further actions.

All operations on IODFs are performed on behalf of user IDs which have to be explicitly permitted for the HCD LDAP backend. This does not affect your system security because the HCD LDAP backend only supports LDAP clients who are bound to the SDBM backend using a user ID and the appropriate password.

The HCD LDAP backend supports a subset of LDAP search requests and a subset of LDAP add, delete, and modify requests.

The HCD LDAP backend is able to perform sequences of update requests as transactions. The LDAP client has to support LDAP V3 controls in order to use this transaction feature.

Updates to an IODF are performed via HCD. Thus, it is ensured that the HCD validation rules are applied.

Only existing IODFs can be used with the HCD LDAP backend. The HCD LDAP backend cannot be used to create or delete IODFs. It cannot be used, for example, to build a production IODF or perform dynamic activation.
Introduction to LDAP

The following is a brief introduction to the LDAP structure and function. For more information refer to IBM Tivoli Directory Server Administration and Use for z/OS.


LDAP is a protocol which makes directory information accessible. A directory can be considered to be a type of yellow pages. New entries can be added, existing entries can be altered or deleted, and it is possible to search for all matching entries using wildcards.

The LDAP directory is represented hierarchically in a so-called Directory Information Tree (DIT). The nodes of this tree are called entries. Every entry is an instance of an object class. An object class is a name which is associated with a collection of attributes.

Every attribute is either mandatory (required) or optional and either single valued or multi valued. Required attributes must have one or more values, optional attributes can have zero or more values. One required single valued attribute of every object class is objectclass: This attribute specifies the object class of which the entry is an instance.

Each entry has a relative distinguished name (RDN) which is specified when the entry is created. The RDN must identify the entry uniquely among its siblings (not necessarily in the whole tree); it consists of one or more attribute-value pairs. The distinguished name (DN) of an entry is the sequence of the RDNs starting from the entry itself and ending with the RDN of the root entry. The DN identifies a node uniquely within the whole DIT.

An example for a DN of a partition within a DIT for HCD IODFs would be the following:

```
hcdPartitionId=PART00,hcdProcessorConfigId=PROC00,
hcdIodfId=SYS1.IODF00,cn=HCD
```

This is illustrated in the following figure.
IBM Tivoli Directory Server for z/OS owns a DIT which can be accessed by a client via the LDAP protocol. In the case of the IBM Tivoli Directory Server for z/OS, the task of storing the DIT is delegated to several so-called backends or plug-ins. Each plug-in holds a certain subtree portion and is responsible for carrying out the client's requests on this portion. The DN of the root entry of a plug-in is called a suffix. When the IBM Tivoli Directory Server for z/OS receives a request concerning an entry it extracts the suffix from the DN of that entry in order to determine which plug-in is responsible for the request.

HCD supports the IBM Tivoli Directory Server for z/OS by providing the HCD LDAP backend plug-in which makes IODFs accessible via the LDAP protocol.

**HCD LDAP backend: Structure and mode of operation**

The HCD LDAP backend is plugged into the IBM Tivoli Directory Server for z/OS. It is configured using the IBM Tivoli Directory Server for z/OS configuration file (typically called ds.conf).

The HCD LDAP backend is similar to the RACF backend SDBM. As with SDBM, the main function of the HCD LDAP backend is to mediate between the IBM Tivoli Directory Server for z/OS and an external component, in this case HCD. HCD retains control over the IODFs; update requests are validated, processed, and the results stored by HCD in the appropriate IODF. Since it is HCD that processes the requests, updates through the IBM Tivoli Directory Server for z/OS preserve the integrity of the IODFs.

Thus, the HCD portion of the DIT must reflect the data structure of HCD exactly. For this reason, rather strict rules (as compared to the DB2 backend TDBM) have to be observed when requesting an update of IODF data through the IBM Tivoli Directory Server for z/OS.
Access control to the HCD LDAP backend is based on RACF permissions for user IDs, not (as is the usual practice) on LDAP Access Control Lists (ACLs). The HCD LDAP backend performs all services on behalf of a user ID. It accepts a service request only on condition that the associated user ID has previously been bound to (authenticated by) SDBM. If this condition is fulfilled, the HCD LDAP backend switches to this user ID and tries to perform the request using only the RACF access rights granted to the user ID in question. In this way, access to IODFs through the LDAP interface and through the ISPF interface are both controlled by the same security mechanism. Note that this will have some consequences for the configuration of the IBM Tivoli Directory Server for z/OS.

The HCD LDAP backend uses several instances of HCD to perform operations on IODFs. Each of these instances serves exactly one request at a time on behalf of a user ID. This strategy provides an easy method of handling the validation of modified configuration data and serialization of client requests. The HCD instances are managed according to the following principles:

1. After starting up, the HCD LDAP backend opens a (configurable) minimum number of address spaces each of which contains exactly one HCD instance for handling requests. The minimum number of address spaces is controlled by the configuration file parameter `MinHcdInstances` (see "Configuration file parameters" on page 356).

2. When the HCD LDAP backend receives a legitimate request on behalf of a user ID, it assigns this request to an HCD instance. This instance is then tied to the user in question, that is, all subsequent requests from this user will be routed to this same HCD instance.

3. If the number of available instances is not sufficient, the HCD LDAP backend will open a new instance provided that a (configurable) maximum number of instances is not exceeded. The maximum number of address spaces is controlled by the configuration file parameter `MaxHcdInstances` (see "Configuration file parameters" on page 356).

4. In order that instances tied to a user can be switched to another user after having been idle for a certain time, two (configurable) timeout values can be defined:
   - The lower value specifies the time after which an instance can be switched to a user who requests a service and has not yet been tied to an HCD instance. The lower timeout value is controlled by the configuration file parameter `AllowSwitchTime` (see "Configuration file parameters" on page 356).
   - The higher value specifies the time after which the connection between an HCD instance and a user is dissolved in any case. The higher timeout value is controlled by the configuration file parameter `ForceSwitchTime` (see "Configuration file parameters" on page 356).

   The lower value provides additional flexibility: As long as there is no need to switch to a new user, the current connection can be maintained until the second timeout is reached.

A special feature of the HCD LDAP backend is that it supports transactions. A transaction is a sequence of requests which is only executed as a whole. If one of the individual requests fails, the whole transaction is not carried out. This provides additional protection against inconsistency of data. Note, however, that transactions are only supported in conjunction with LDAP V3, not with LDAP V2.
The HCD LDAP backend performs its services on behalf of user IDs and uses only the RACF access rights of these user IDs to determine the legitimacy of a request. This assumes that the plug-in runs under a user ID which is entitled to switch to the user ID of the respective bind request. Since the plug-in takes as its user ID that of the IBM Tivoli Directory Server for z/OS, the HCD LDAP backend can only be plugged into the IBM Tivoli Directory Server for z/OS, if the latter runs as a started task under a user ID which is permitted to switch to another user ID. The HCD LDAP backend uses the pthread_security_np() service for performing this switch (thread-level security model). For more details on this switch, see the section “Preparing security for servers” of the z/OS UNIX System Services Planning book.

The further setup depends on which security level you choose for the IBM Tivoli Directory Server for z/OS. There are two options:

- UNIX level security
- z/OS UNIX level security

With UNIX level security, the IBM Tivoli Directory Server for z/OS must run under the superuser. The superuser has on this security level total authority over the system; in particular, he is automatically entitled to assume the identity of any other user. With z/OS UNIX level security on the other hand, the right to switch user IDs must be explicitly granted, even to the superuser.

z/OS UNIX level security is more secure than UNIX level security, and we recommend that you choose this option. However, you must be aware that this is a global decision which may have consequences for every server on your system. For this reason, the steps required for both options are described below in detail. Warnings are issued whenever a step has repercussions for your system configuration.

For both options you must issue a number of RACF commands. In the example commands shown in the following descriptions, typical assumptions about the system configuration have been made. As these do not necessarily conform with your particular system configuration, you may need to modify the commands as required.

If you have already been using the IBM Tivoli Directory Server for z/OS and are adding the HCD LDAP backend, check whether your setup conforms to the requirements of the HCD LDAP backend. If it does not (for instance, you are not running the IBM Tivoli Directory Server for z/OS as a started task), then you must change your setup.

Performing the setup

It is recommended to divide the setup process into three parts. The first part consists of setting up the IBM Tivoli Directory Server for z/OS so that it is able to run with the HCD LDAP backend. The second part consists of setting up the HCD LDAP backend. The third part consists of integrating the HCD schema into the IBM Tivoli Directory Server for z/OS.

Setting up the IBM Tivoli Directory Server for z/OS

This section lists the prerequisites that the IBM Tivoli Directory Server for z/OS must comply with, so that it can run with the HCD LDAP backend.
Prerequisites for IBM Tivoli Directory Server for z/OS setup:

- Set up the IBM Tivoli Directory Server for z/OS for running as a started task.
- Establish a separate user ID that runs the IBM Tivoli Directory Server for z/OS.
- Set up the IBM Tivoli Directory Server for z/OS for running in single-server mode without replication.
- Configure the IBM Tivoli Directory Server for z/OS with the SDBM backend. The HCD LDAP backend requires the IBM Tivoli Directory Server for z/OS to run with this RACF backend. Therefore, SDBM must be included in the configuration file and all prerequisites for SDBM must be fulfilled.

Additional Prerequisites for z/OS UNIX level security:

Up to here, you have completed to set up the IBM Tivoli Directory Server for z/OS with UNIX level security. If you want to have z/OS UNIX level security, your environment must comply to the following prerequisites:

- Define the RACF FACILITY profile BPX.SERVER. Refer to IBM Tivoli Directory Server Administration and Use for z/OS and z/OS UNIX System Services Planning for details what this means for the setup of the IBM Tivoli Directory Server for z/OS.

  **Note:**
  Defining a profile named BPX.SERVER in the RACF class FACILITY switches system security from UNIX level security to z/OS UNIX level security. Other applications may be affected by this switch.

- Define libraries to program control. Under z/OS UNIX level security, every program that is loaded into a server address space must be marked as controlled (see section "Defining Modules to Program Control" of the z/OS UNIX System Services Planning book).

  **Note:**
  Changing profiles in RACF class PROGRAM can cause severe system problems if not done in a manner suitable for the system. If you are unsure ask your RACF administrator.

Setting up the HCD LDAP backend

This section describes how to set up the HCD LDAP backend. There is a sample REXX procedure CBDSLCUS in SYS1.SAMPLIB that contains the RACF commands listed in this section for the z/OS UNIX level security.

To set up the HCD LDAP backend, perform the following steps. Steps 1 on page 352 through 3 on page 354 are only required for z/OS UNIX level security. If you have chosen UNIX level security, continue with step 4 on page 354.

1. **Authorize the HCD LDAP backend to act on behalf of other user IDs.** With UNIX level security, the IBM Tivoli Directory Server for z/OS runs under a superuser (with UID 0) who is permitted to act on behalf of any user ID. Under z/OS UNIX level security, the HCD LDAP backend (which receives no password from the IBM Tivoli Directory Server for z/OS) can only perform a service for a client user ID when it has been explicitly authorized to act on behalf of that user ID (see the section "Defining Servers to Process Users without Password" of z/OS UNIX System Services Planning).
In order to authorize the user ID of the server to act on behalf of another user ID, you must do the following:

a. Define a surrogate profile for the prospective client by issuing the following RACF commands (the second command updates the in-storage copy of the SURROGAT profiles):

\[
\text{RDEFINE SURROGAT BPX.SRV.UserID UACC(NONE)} \\
\text{SETROPTS RACLIST(SURROGAT) REFRESH}
\]

b. Authorize the user ID of the IBM Tivoli Directory Server for z/OS for this profile by the following commands (the second command updates the in-storage copy of the SURROGAT profiles):

\[
\text{PERMIT BPX.SRV.UserID CLASS(SURROGAT) ID(LDAPSrv) ACCESS(READ)} \\
\text{SETROPTS RACLIST(SURROGAT) REFRESH}
\]

These example commands are based on the following assumptions (which may not hold for your system!):

a. The RACF class SURROGAT has been activated.

b. There is no profile in that class with the name BPX.SRV.UserID, where \( \text{UserID} \) is the user ID of the prospective client.

c. The user ID of the IBM Tivoli Directory Server for z/OS is LDAPsrv.

2. Define libraries to program control. Under z/OS UNIX level security, every program that is loaded into a server address space must be marked as controlled (see section "Defining Modules to Program Control" of the z/OS Linux System Services Planning book).

When using the HCD LDAP backend, the libraries containing the following load modules must be defined to program control:

<table>
<thead>
<tr>
<th>Load Modules</th>
<th>Typical Library</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCD LDAP backend</td>
<td>SYS1.LINKLIB on SYSRES</td>
</tr>
<tr>
<td>HCD</td>
<td>SYS1.LINKLIB on SYSRES</td>
</tr>
<tr>
<td>UIMs</td>
<td>SYS1.NUCLEUS on SYSRES</td>
</tr>
<tr>
<td>C++ RTL</td>
<td>CEE.SCEERUN on SYSRES</td>
</tr>
<tr>
<td>IBM Tivoli Directory Server for z/OS and SDBM backend</td>
<td>SYS1.SIEALNKE and SYS1.LPALIB on SYSRES</td>
</tr>
</tbody>
</table>

Note: If you use load modules from other libraries you have to define these libraries to program control as well.

To define these libraries to program control, issue the following RACF commands:

\[
\text{RDEFINE PROGRAM ** UACC(READ)} \\
\text{RALTER PROGRAM ** UACC(READ) ADDMEM('SYS1.LINKLIB'/'******'/NOPADCHK)} \\
\text{RALTER PROGRAM ** UACC(READ) ADDMEM('SYS1.NUCLEUS'/'******'/NOPADCHK)} \\
\text{RALTER PROGRAM ** UACC(READ) ADDMEM('CEE.SCEERUN'/'******'/NOPADCHK)} \\
\text{RALTER PROGRAM ** UACC(READ) ADDMEM('SYS1.SIEALNKE'/'******'/NOPADCHK)} \\
\text{RALTER PROGRAM ** UACC(READ) ADDMEM('SYS1.LPALIB'/'******'/NOPADCHK)} \\
\text{SETROPTS WHEN(PROGRAM) REFRESH}
\]

The first command defines a profile named ** to the class PROGRAM. The other commands, except the last, define the libraries containing the load modules to program control. The last command refreshes the in-storage copy of the PROGRAM profiles.

The example commands are based on the following assumptions (which may not hold for your system!):

a. The RACF class PROGRAM has been activated.
b. GENERIC is enabled for the RACF class PROGRAM.

c. There is no profile in that class with the name **.

d. The load modules needed reside in their typical libraries as listed above.

3. When using the HCD LDAP backend, the libraries containing the following load modules must be APF authorized:

<table>
<thead>
<tr>
<th>Load Modules</th>
<th>Typical Library</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCD LDAP backend</td>
<td>SYS1.LINKLIB on SYSRES</td>
</tr>
<tr>
<td>C++ RTL</td>
<td>CEE.SCEERUN on SYSRES</td>
</tr>
<tr>
<td>IBM Tivoli Directory Server for z/OS and SDBM backend</td>
<td>SYS1.SIEALNKE and SYS1.LPALIB on SYSRES</td>
</tr>
</tbody>
</table>

The following steps apply to both security levels.

4. **Tailor the started task procedure.** This includes:

- The HCD instances that have been started by the HCD LDAP backend have the same region size as the IBM Tivoli Directory Server for z/OS started task. So, you may need to adjust the region size of the IBM Tivoli Directory Server for z/OS started task according to the region size suitable for the HCD instances.

- You have to ensure that the IBM Tivoli Directory Server for z/OS and the HCD LDAP backend are able to find the load modules in Table 10 on page 353 by using the z/OS search order. If the libraries containing these load modules are not searched by z/OS on your system, you must insert a STEPLIB DD, which contains the missing libraries, into the started task procedure.

5. **Tailor the IBM Tivoli Directory Server for z/OS configuration file.** You must include the definition of the HCD LDAP backend in the configuration file ds.conf. A sample of how to define the HCD LDAP backend as IBM Tivoli Directory Server for z/OS plug-in in the server configuration file is delivered with HCD and is installed in /usr/lpp/hcd/examples/ds.conf. For this purpose you must do the following (the examples are taken from the sample file):

A plug-in configuration statement must be added into the GLOBAL section of the configuration file ds.conf of the IBM Tivoli Directory Server for z/OS to define the HCD LDAP backend as a plug-in:

```
plugin clientOperation CBDMPLPG hcd_plginit "<parameters>"
```

The parameters that are recognized by the HCD LDAP backend are described in "Configuration file parameters" on page 356. Here is an example of how to replace the <parameters> by acceptable keyword/value pairs (enclosed in double quotes). Note that the HCD suffix cn=HCD must be passed as parameter of the plug-in statement.

```
*suffix cn=HCD
MinHcdInstances 1
MaxHcdInstances 3
AllowSwitchTime 30
ForceSwitchTime 600
TransactionRollbackTime 3600
Trace off
Profile off
TraceDsnSuffix HCD.TRACE
ProfileDsnSuffix HCD.PROFILE
TransformAttributeValues off"
```

**Note:** For the processing of the IBM Tivoli Directory Server for z/OS configuration file, the following general rules apply:
6. **Run the HCD LDAP backend.** To verify that your setup is working issue an LDAP request against the HCD LDAP backend. You can use the LDAP operation utilities to do this. For this purpose, enter a command according to the following template:

```
ldapsearch -h ldaphost -p ldapport -D binddn -w passwd -s base
    -b "hcdIodfId=IodfName,suffix" "objectclass=*"
```

This command performs a search on the specified IODF on behalf of the user ID specified by `binddn`. `binddn` must be a DN from within the SDBM name space representing a user ID, and `passwd` the appropriate password. `IodfName` must be the name of an existing IODF data set. `suffix` would be `cn=HCD` if you have kept the default value specified in the sample configuration file `ds.conf`. If the request returns a plausible result, the HCD LDAP backend is working correctly.

**Integrating the LDAP schema for HCD**

HCD is shipped with a predefined schema file representing schema definitions which the IBM Tivoli Directory Server for z/OS needs to evaluate incoming HCD requests issued via the LDAP interface. You must integrate this file into the IBM Tivoli Directory Server for z/OS after this server has been successfully installed and set up. It is recommended that this integration step is performed by the person who is responsible for the IBM Tivoli Directory Server for z/OS (usually the system administrator). The name of the HCD schema file is `schema.hcd.ldif` and is located in the `/usr/lpp/hcd/etc` directory.

Use the `ldapmodify` command to load the schema, for example:

```
ldapmodify -h ldaphost -pldapport -D adminDN -w passwd
    -f /usr/lpp/hcd/etc/schema.hcd.ldif
```

See [IBM Tivoli Directory Server Client Programming for z/OS](#) for more information about `ldapmodify`.

**Customizing the HCD LDAP backend**

The HCD LDAP backend can be customized in the following ways:

- Parameters which are recognized by the HCD LDAP backend can be set in the IBM Tivoli Directory Server for z/OS’s configuration file.
- LDAP debug levels can be set for the IBM Tivoli Directory Server for z/OS and apply to the HCD LDAP backend also.
- HCD LDAP backend can be run in English or Japanese.
**Configuration file parameters**

Configuration file parameters that apply to the HCD LDAP backend are set in the database section of the HCD LDAP backend in the IBM Tivoli Directory Server for z/OS configuration file ds.conf. Any parameter which is not known by the IBM Tivoli Directory Server for z/OS itself is handed over to the HCD LDAP backend. The following list shows the supported parameters:

**MinHcdInstances int**

Specifies the minimum number of HCD instances started during startup of the HCD LDAP backend.

Default: 1
Range: 1-10
Constraints: Must be less than or equal to the numerical value of MaxHcdInstances

For example, MinHcdInstances 1 causes the HCD LDAP backend to start exactly one initial HCD instance.

**MaxHcdInstances int**

Specifies the maximum number of HCD instances which can be used simultaneously by the HCD LDAP backend. The HCD LDAP backend will start new HCD instances dynamically if they are needed and if the maximum number has not already been reached.

Default: 3
Range: 1-10
Constraints: Must be greater than or equal to the numerical value of MinHcdInstances

For example, MaxHcdInstances 3 will allow the HCD LDAP backend to use a maximum of three HCD instances simultaneously.

**AllowSwitchTime int**

Specifies the idle time in seconds after which the user ID of an HCD instance may be changed.

Default: 30
Range: 1-120
Constraints: Must be less than the numerical value of ForceSwitchTime.

For example, AllowSwitchTime 30 will ensure that an HCD instance must be idle for at least thirty seconds before the HCD LDAP backend is allowed to switch the HCD instance to a different user ID.

**ForceSwitchTime int**

Specifies the idle time in seconds after which the user ID of an HCD instance is unconditionally reset. If a new request on behalf of this user ID arrives, this or any other free HCD instance has to be switched to that user ID.

Default: 600
Range: 60-3600
Constraints: Must be greater than the numerical value of AllowSwitchTime.

Exception: If the value specified for ForceSwitchTime is 0, a forced reset of the user ID will not be performed.
For example, `ForceSwitchTime 600` will ensure that the user ID of an HCD instance is reset by the HCD LDAP backend after an idle time of 600 seconds (10 minutes) of the HCD instance.

**TransactionRollbackTime** *int*

Specifies the idle time (in seconds) of an LDAP client after which a transaction, running on behalf of this LDAP client is rolled back automatically.

**Default:** 3600

**Range:** 10-3600

For example, `TransactionRollbackTime 3600` will ensure that the HCD LDAP backend automatically rolls back a transaction which is still open on behalf of an LDAP client which has been idle for 3600 seconds (one hour).

**Trace on/off**

Determines whether the HCD instance running on behalf of the HCD LDAP backend generates an HCD trace (see “HCD trace facility” on page 467). Each HCD instance generates its own trace.

**Default:** off

**TraceDsnSuffix** *name*

Specifies the suffix of the data set names where the traces of the HCD instances are to be written to. The effective name of the trace data set of one particular HCD instance is determined by concatenating the user ID associated with the HCD instance with the suffix.

**Default:** HCD.TRACE

**Constraints:** Only valid data set names with a maximum of 35 characters can be used. Note that the suffix must not be empty.

For example, assume you have set the `TraceDsnSuffix` to LDAP.HCD.TRACE and that you have specified `Trace on`. If an HCD instance now performs an LDAP request on behalf of user ID TEST and is, therefore, switched to this user ID, it will open the trace data set with name 'TEST.LDAP.HCD.TRACE' and DISP=OLD. This data set is then used for tracing the operations of the HCD instance.

**Profile on/off**

Determines whether the HCD instances running on behalf of the HCD LDAP backend will use an HCD profile on startup or user ID switch (see “Defining an HCD profile” on page 23). Each HCD instance uses its own profile, depending on the user ID which the HCD instance is currently related to.

**Default:** off

**ProfileDsnSuffix** *name*

Specifies the suffix of the profile data set name used by the HCD instances. The effective name of the profile data set of one particular HCD instance is determined by concatenating the user ID associated with the HCD instance with the suffix.

**Default:** HCD.PROFILE

**Constraints:** Only valid data set names with a maximum of 35 characters can be used. Note that the suffix must not be empty.

For example, assume you have set `ProfileDsnSuffix` to LDAP.HCD.PROFILE and you have specified `Profile on`. If an HCD
instance now performs an LDAP request on behalf of user ID TEST
and is thus switched to this user ID, it will open the profile data
set with name 'TEST.LDAP.HCD.PROFILE' and DISP=SHR. The
contents of the data set will be read and the HCD instance will be
set up accordingly.

**TransformAttributeValues on/off**

Specifies whether the values of the attributes hcdIodfDescription
and hcdDescription are interpreted as being in IBM-939 or not. In
general, these values are interpreted as being in IBM-037. It is
important to have the correct setting because the IBM Tivoli
Directory Server for z/OS communicates with LDAP clients using
UTF-8 representation and the HCD LDAP backend has to convert
attribute values appropriately.

Default: off

**LDAP debug level**

For a description of the debug levels see “LDAP problem determination” on page 462.

**Message translation**

The HCD LDAP backend supports English and Japanese messages. To obtain
English messages with the character representation IBM-1047, choose one of the
following values for the environment variable LANG:

- En_US
- En_US.IBM-1047

To obtain Japanese messages with the character representation IBM-939, choose one
of the following values for the environment variable LANG:

- Ja_Jp
- Ja_Jp.IBM-939

For the values of LANG supported by the IBM Tivoli Directory Server for z/OS and
for more information on how to specify the value of LANG, see the IBM Tivoli
Directory Server Administration and Use for z/OS.

Note: The settings of the environment variable LANG do not control the language of
the HCD messages which HCD LDAP backend returns to LDAP clients upon their
requests. In fact, all HCD messages returned to LDAP clients will always be in
English.

**Accessing IODF information**

The HCD LDAP backend provides access to IODF information via the LDAP
protocol. You can, for instance, add or delete devices and control units, or modify
operation system settings. It is also possible to automate complex updates, for
instance a whole I/O configuration, by communicating with the IBM Tivoli
Directory Server for z/OS through an application (see IBM Tivoli Directory Server
Client Programming for z/OS).

**The IODF directory information tree**

In order to make HCD IODF information accessible via the LDAP protocol, the
HCD LDAP backend realizes a one-to-one mapping between the IODF data
structure and the structure of the LDAP directory information tree (DIT). It is this one-to-one mapping that makes the update of IODFs through the IBM Tivoli Directory Server for z/OS possible.

The resulting HCD portion of the DIT has some special constraints as compared to an all purpose backend like TDBM:

- One important difference between the HCD portion and TDBM concerns the relation between object class and position in the (sub)tree. In TDBM, every object class can occur on every position. The HCD subtree, on the other hand, must correspond to the structure of the IODF. Accordingly, the object class of every entry managed by the HCD LDAP backend uniquely determines the object class of its parent entry.

  Note, however, that this parent relation cannot be understood as in object oriented programming: Thus the object class hcdPartition has as its parent the object class hcdProcessorConfig, but hcdPartition is certainly not derived from hcdProcessorConfig.

- In addition, there are a number of HCD specific dependencies between entries that reside on different branches of the HCD subtree. In some cases, the HCD LDAP backend takes these dependencies into account by automatically adding or deleting certain entries when certain other entries are added or deleted. In other cases you must ensure yourself that your request conforms with these interrelationships. For details, see “Restrictions for search and update requests” on page 361 and “Appendix F. IODF data model” on page 505.

- The schema of the HCD portion of the DIT is fixed. It cannot be extended or modified.

The object classes required for the IODF DIT and their interrelationships (parent relation and branch crossing interrelationships) are described in “Appendix F. IODF data model” on page 505. This appendix also contains descriptions of all the attributes that are needed for these object classes.

Performing LDAP requests on IODFs

In order to request a service from the HCD LDAP backend you must always perform two steps:

1. **Authenticate yourself to the RACF Backend.** For authentication, you have to specify a so-called bind DN and a password. The authentication (called binding in LDAP terminology) is successful when the bind DN identifies an entry of the backend to which the suffix of the bind DN is assigned, and when the specified password is in fact associated with this entry. The HCD LDAP backend does not perform this identity check itself, but uses the SDBM backend for that purpose. Thus, you must specify a user ID in the bind DN, and the bind DN must have the following structure:

   "racfid=user_ID,profileType=user,sysplex=Your_Sysplex,..."

   Here sysplex=Your_Sysplex is the first element of the suffix. Note that group is not allowed as the value of the profileType attribute.

   If the binding to the RACF Backend fails, the HCD LDAP backend will not support the subsequent requests.

2. **Access an IODF.** If you want to access an IODF with the data set name IodfDsn, the DN in your LDAP request must have the following form:

   "...,hcdIodfId=IodfDsn,suffix_of_HCD_LDAP_Backend"
Note that only IODFs that can be reached by the IBM Tivoli Directory Server for z/OS are accessible with the HCD LDAP backend.

If you have issued your request as described, the HCD LDAP backend will take the following actions:

- The user ID is extracted from the initial bind request.
- The IODF data set name is extracted from the LDAP request.
- A prepared address space is switched to the user ID, and HCD is started.
- HCD loads the specified IODF and performs the requested operations.
- The results are sent back to you.

Operational behavior

The operational behavior of the HCD LDAP backend is basically the same as that for the RACF backend with some small differences. Information about the RACF Backend can be obtained from the IBM Tivoli Directory Server Administration and Use for z/OS.

The following gives an overview of the functional behavior of the HCD LDAP backend:

1. The HCD LDAP backend does not participate in extended group membership searching on a client request.
2. It is possible to run several HCD LDAP backends on one IBM Tivoli Directory Server for z/OS simultaneously.
3. The root of a subtree (see Figure 136 on page 349) is denoted by a suffix in the configuration file. You can specify only one suffix per HCD LDAP backend. Suffix names must be unique if you are running multiple HCD LDAP backends.
4. The HCD LDAP backend does not support Access Control Lists (ACLs) which are normally used to protect information stored in an LDAP directory from unauthorized access. The reason for this is that the DIT portions managed by the HCD LDAP backend are based on IODF data sets for which all access control is performed by RACF.
5. The HCD LDAP backend does not support the following LDAP request types and will answer these requests with the return code "Unwilling to Perform":
   - Bind
   - ModifyDN (also called ModifyRDN, or ModRdn)
   - Compare
   - Abandon
   - Extended Request
6. The following LDAP request types are supported by the HCD LDAP backend:
   - Add
   - Delete
   - Modify
   - Search

The following table shows how the HCD LDAP backend behaves during these LDAP operations:

<table>
<thead>
<tr>
<th>Target DN</th>
<th>Search</th>
<th>Add</th>
<th>Delete</th>
<th>Modify</th>
</tr>
</thead>
</table>
Target DN | Search | Add | Delete | Modify
---|---|---|---|---
hcdIodfId=IodfDatasetName,suffix | Perform the appropriate search request. See "Search" | Error: Inappropriate Matching | Error: Inappropriate Matching | Perform the appropriate modify request. See "Modify" on page 363
...hcdIodfId=IodfDatasetName,suffix | Perform the appropriate search request. See "Search" | Perform the appropriate add request. See "Add" on page 362 | Perform the appropriate delete request. See "Delete" on page 363 | Perform the appropriate modify request. See "Modify" on page 363

7. Multi-server or replication is not supported by the HCD LDAP backend

Restrictions for search and update requests

This section describes the restrictions which the HCD LDAP backend imposes on the search, add, delete and modify capabilities of LDAP. Many of these restrictions derive from the fact that the structure of HCD portion of the DIT is much more rigidly controlled than, for instance, the TDBM subtree.

Note: Within a single request, references to an attribute name must either always be with the alias name or always with the full attribute name. A mix is not accepted.

In the following subsections, suffix stands for the suffix of the HCD LDAP backend.

Search

Searching is restricted as follows:

- Only search bases ending with hcdIodfId=Iodf_dataset_name,suffix are supported. This implies that only one IODF can be searched at a time.
- The only search filters that are supported by the HCD LDAP backend are objectclass=* and objectclass=**name, where name has to be the name of an object class that is defined for the HCD LDAP backend.
- Time or size limits are not supported.
- Controls are not supported.
- It is not possible to restrict the attributes of the matching entries that will be displayed. Every attribute that has at least one value will be shown in the search results.

Examples:

Following are two examples for retrieving information from an existing IODF with the command line search utility of LDAP:

The command

```
ldapsearch -D "racfid=TEST,profiletype=user,sysplex=sysplex1" -w "passwd"
-s base -b "hcdIodfId=TEST.IODF00.WORK,suffix" "objectclass=hcdIodf"
```

retrieves the top entry of object class hcdIodf belonging to the IODF named TEST.IODF00.WORK on behalf of user ID TEST. The result may look as follows:

```
hcdIodfId=TEST.IODF00.IODF_WORK
objectClass=hcdIodf
hcdIodfId=TEST.IODF00.IODF_WORK
```

hcdIodfType=W
hcdIodfDescription=Testing purposes
hcdBlocksAllocated=20
hcdBlocksUsed=2
hcdCreationDate=1999-10-04
hcdLastUpdateDate=1999-12-16
hcdLastUpdateTime=09:25:50
1 matches

The same result could be obtained with the search filter "objectclass=*".

The command

```
ldapsearch -D "racfid=TEST,profiletype=user,sysplex=sysplex1" -w "passwd"
-s one -b "hcdIodfId=TEST.IODF00.WORK,,suffix" "objectclass=hcdDevice"
```

retrieves all entries of object class hcdDevice belonging to the IODF named TEST.IODF00.WORK, again on behalf of user ID TEST. One of the retrieved entries may look as follows:

```
hcdDeviceNumber=000D,hcdIodfId=TEST.IODF00.WORK,,suffix
objectClass=hcdDevice
hcdDeviceNumber=600D
hcdUnit=2540P
hcdModel=1
hcdDescription=Virt. Puncher
```

**Note:** Attribute names in the search results may be in lower case only, depending on the set up of the IBM Tivoli Directory Server for z/OS for example, `hcdiodfid` instead of `hcdIodfId`. Also, there is no specific order of the attribute/value pairs in the returned result.

**Add**

Adding an entry is restricted as follows:

- Entries can only be added below `hcdIodfId=...,suffix`, that is, add is not supported on DN `hcdIodfId=...,suffix` or DN `suffix`.
- Since every object class of the HCD LDAP backend except `hcdIodf` has a uniquely determined parent class, ensure that the object class of the new entry and that of the entry to which the new entry is appended are related as child and parent. Exactly one value must be specified for the `objectclass` attribute. See "Appendix F. IODF data model" on page 505 for parent-child relationships between object classes.
- If the RDN of the entry to be added is `attribute=value`, value must be specified as a value of `attribute` inside the entry.
- The attributes which are contained in the entry’s RDN are determined by the object class of an entry. See "Appendix F. IODF data model" on page 505.
- There must be no entry in the DIT with the same DN as the entry to be added.
- If an add request fails because of a missing parent, the HCD LDAP backend does not update the matched DN field of the result.
- Check "Appendix F. IODF data model" on page 505 to see which object classes can be added.
- Adding an entry may cause other entries to be created automatically using default values. See "Appendix F. IODF data model" on page 505.
- Two controls are supported for the LDAP add request. See "Transactions" on page 365 for details.

**Example:**
A new entry of the object class hcdControlUnit of type 3990 with a control unit number of 0100 can be appended to the entry hcdIodfId=TEST.IODF00.WORK,suffix as follows.

First create a data set member named TEST.LDIF(ADDCU100) with the content:

```
dn:hcdControlUnitNumber=0100,hcdIodfId=TEST.IODF00.WORK,suffix
changetype:add
objectclass:hcdControlUnit
hcdControlUnitNumber:0100
hcdUnit:3990
```

Then call the LDAP command line utility `ldapadd` with the following parameters:

```
ldapadd -D "racfid=TEST,profiletype=user,sysplex=sysplex1" -w "passwd"
-f //TEST.LDIF(ADDCU100)
```

The entry will be added on behalf of the user ID TEST.

You can then verify that the entry was created correctly by issuing:

```
ldapsearch -D "racfid=TEST,profiletype=user,sysplex=sysplex1" -w "passwd"
-s base -b "hcdControlUnitNumber=0100,hcdIodfId=TEST.IODF00.WORK,suffix"
"objectclass=*"
```

The search result should look like:

```
hcdcontrolunitnumber=0100,hcdIodfId=TEST.IODF00.WORK,suffix
objectclass=hcdControlUnit
hcdcontrolunitnumber=0100
hcdunit=3990
1 matches
```

## Delete

Deleting an entry is restricted as follows:

- Only entries below DN hcdIodfId=...,suffix can be deleted. Delete on DN hcdIodfId=..., suffix or DN suffix is not supported.
- Check ["Appendix F. IODF data model" on page 505](#) to see which object classes can be deleted.
- Deleting one entry may cause other entries to be deleted automatically. See ["Appendix F. IODF data model" on page 505](#).
- Two controls are supported for the LDAP delete request. See ["Transactions" on page 365](#) for details.

**Example:**

To delete the entry added in the example shown in ["Add" on page 362](#) you can call the LDAP command line utility `ldapdelete` with the following parameters:

```
ldapdelete -D "racfid=TEST,profiletype=user,sysplex=sysplex1" -w "passwd"
"hcdControlUnitNumber=0100,hcdIodfId=TEST.IODF00.WORK,suffix"
```

The entry will be deleted on behalf of the user ID TEST.

## Modify

Modifying an entry is restricted as follows:

- Only the entry DN hcdIodfId=...,suffix and below can be modified. Modification of DN suffix is not supported.
- Check ["Appendix F. IODF data model" on page 505](#) to see which object classes can be modified.
• The HCD LDAP backend only supports the delete and replace subcommands of modify. The add subcommand is NOT supported.
• The value of the object class attribute cannot be deleted or replaced.
• The value(s) of the attributes which are contained in the entry’s RDN cannot be deleted or replaced.
• One modify request to a single entry can contain a sequence of delete and replace subcommands. This sequence can be considered as atomic: Either the whole sequence is performed or nothing is performed.
• One attribute can only be referenced once in the whole modify request. It can only be deleted once, replaced once, and only either be deleted or replaced.
• Modify delete only supports the deletion of all values of an attribute. For this reason, you must not specify values in the modify delete request. If a value is specified, the whole modify request is rejected by the HCD LDAP backend.
• Attributes described as mandatory in an object class must not be deleted
• Modify replace replaces all existing values of the given attribute with the new values listed, creating the attribute if it did not already exist. A replace with no value will delete the entire attribute if it exists, and is ignored if the attribute did not exist.
• All values must conform with the type specified in the attribute definition.
• Modifying an entry may cause other entries to be modified automatically. See “Appendix F. IODF data model” on page 505.
• Two controls are supported for the LDAP modify request. See “Transactions” on page 363 for details.

Example:

The entry created in “Add” on page 362 can be modified by adding the attribute hcdDescription as follows.

First create a data set member named TEST.LDIF(REPCU100) with the content
dn:hcdControlUnitNumber=0100,hcdIodfId=TEST.IODF00.WORK,suffix
changetype:modify
replace:x
hcdDescription=New description

Then call the LDAP command line utility ldapmodify with the following parameters:
ldapmodify -D "racfid=TEST,profiletype=user,syplex=sysplex1" -w "passwd"
-f '//TEST.LDIF(REPCU100)"

If the modify request completes successfully, the entry will look like:
hcdControlUnitNumber=0100,hcdIodfId=TEST.IODF00.WORK,suffix
objectClass=hcdControlUnit
hcdControlUnitNumber=0100
hcdUnit=3990
hcdDescription=New description

This hcdDescription can now be deleted again with the delete subrequest of modify. To do this, first create a data set member named TEST.LDIF(DELCU100) with the content
dn:hcdControlUnitNumber=0100,hcdIodfId=TEST.IODF00.WORK,suffix
changetype:modify
delete:hcdDescription
-
Then issue the following command:

```
ldapmodify -D "racfid=TEST,profiletype=user,sysplex=sysplex1" -w "passwd"
-f //'TEST.LDIF(DELCU100)"
```

### Transactions

Transactions is a concept in which individual LDAP requests are handled collectively as a single unit such that if one of the requests within the transaction should fail, then the whole transaction with all its requests is not carried out. In this way, you do not have a situation where some requests are performed and others are not. Transactions can be helpful if you want to perform complex tasks which consist of LDAP request sequences. For example, if you want to define a coupling facility (which would require adding two entries of object class hcdChannelPath) you can perform the necessary LDAP requests within a transaction.

In general, arbitrary sequences of at least two update requests (i.e. add, delete, modify) can be performed as a transaction. A transaction cannot consist of a single update request as a single request by definition behaves like a transaction.

### Prerequisites and method of functioning

The LDAP client marks a request for the HCD LDAP backend as being part of a transaction by specifying an LDAP V3 control for the request. For this reason, transactions in the HCD LDAP backend can only be used with LDAP clients supporting LDAP V3 controls.

LDAP V3 controls represent additional information which can be included with a request to, or to a response from the IBM Tivoli Directory Server for z/OS.

In order to use transactions offered by the HCD LDAP backend, the following two LDAP V3 controls, both of which are supported by the HCD LDAP backend, are required. Both controls must always be used with a valid value which determines how the HCD LDAP backend will react to the control.

Examples of how to use these controls are shown on page 367

1. **hcdTransactionControl**

   - **Name:** hcdTransactionControl
   - **Description:** This control is used on the first and last request of a transaction, to indicate either the start (via value NEW) and finish (via value COMMIT or ROLLBACK) of a transaction request sequence.
   - **Assigned Object Identifier:** 1.3.18.0.2.10.3
   - **Target of Control:** Server
   - **Control Criticality:** Critical
   - **Values:** Value is exactly one char 0 terminated string in UTF-8 encoding representing exactly one of the strings (words):
     - NEW
     - COMMIT
     - ROLLBACK

   These strings are not case sensitive. The following byte values (shown as two digit hexadecimal numbers) represent these strings:
2. **hcdTransactionId**:

   **Name:** hcdTransactionId

   **Description:** This control is used on all requests of a transaction except the first request. The value of this control indicates which particular transaction a request belongs to.

   If an LDAP client initiates a new transaction with an appropriate request, the HCD LDAP backend answers the request with an LDAP response containing control hcdTransactionId. The LDAP client has to extract this control's value from the response and must specify the value for hcdTransactionId on all following requests belonging to the transaction.

   **Assigned Object Identifier:** 1.3.18.0.2.10.4

   **Target of Control:** Server

   **Control Criticality:** Critical

   **Values:** Value is exactly one char 0 terminated string in UTF-8 encoding representing a non-negative, non-zero long int value in decimal format which is the transaction ID. Only values previously received from the HCD LDAP backend are allowed - all others are rejected.

   Example: Assuming a transaction has an ID of 238. Then, the following byte values (shown as two digit hexadecimal numbers) represent the appropriate hcdTransactionId control value:

   \[32 33 38 00\]

   The 10 decimal digits have the following character representation in UTF-8 (shown as two digit hexadecimal numbers):

   \[0 = 30, 1 = 31, 2 = 32, 3 = 33, \ldots, 9 = 39.\]

   In order to perform a transaction containing a sequence of LDAP requests \(R_1, R_2, \ldots, R_n\) the LDAP client has to do the following:

   1. Send the LDAP requests \(R_1, R_2, \ldots, R_n\) of the transaction one after another to the IBM Tivoli Directory Server for z/OS and wait for a response to each request before sending the next.

   2. To initiate a new transaction (containing the requests \(R_1, R_2, \ldots, R_n\)) extend the first request \(R_1\) with control hcdTransactionControl and specify NEW as value for this control. Control hcdTransactionId must not be used on the first request \(R_1\). If the HCD LDAP backend is able to open the new transaction, it will respond by returning the control hcdTransactionControl and the control hcdTransactionId with a transaction ID as value. In the case of failure neither controls will be returned.

   3. Send all subsequent requests \(R_2, \ldots, R_n\) of the current transaction, with the control hcdTransactionId containing the valid transaction ID issued by the HCD LDAP backend as a response to the first request \(R_1\).

   4. Commit (or if necessary rollback) the whole transaction using the control hcdTransactionControl and the value COMMIT (or ROLLBACK). This must be added to the last request \(R_n\) of the transaction. If, however, any request in between fails, you can use the hcdTransactionControl to initiate an immediate ROLLBACK. The control hcdTransactionId with the appropriate value must, of course, also be specified.
Note: Only update requests (i.e. add, delete and modify) can be part of a transaction.

All operations belonging to a transaction must act on the same IODF and must have been issued by the same LDAP client with the same LDAP handle bound to the same user ID.

The following section shows how to initiate a transaction, to add further LDAP requests to a transaction and to close a transaction.

How to initiate, extend and close a transaction

To use the transaction facility, you have to:

- Set up and run the IBM Tivoli Directory Server for z/OS and the HCD LDAP backend as described previously in this information unit.
- Provide an LDAP V3 client program which uses the appropriate controls of the HCD LDAP backend.

Refer to [IBM Tivoli Directory Server Administration and Use for z/OS](https://www.ibm.com/support/docview.ws?rs=0&context=doc&contextarea=product&contentId=infoa000000000016g) [IBM Tivoli Directory Server Client Programming for z/OS](https://www.ibm.com/support/docview.ws?rs=0&context=doc&contextarea=product&contentId=infoa000000000016c) and the IBM redbook [Understanding LDAP](https://www.ibm.com/support/docview.ws?rs=0&context=doc&contextarea=product&contentId=infoa000000000016d) for examples of LDAP client programs.

The following shows some examples of how to use the previously introduced controls on LDAP requests to take the following actions:

- Initiate a transaction
- Submit further transaction requests
- End a transaction

All the following examples written in C, are provided for the LDAP add request (here, we use the LDAP request `ldap_add_ext()` from the LDAP client API in C).

Please note that you must choose the version of the LDAP client API function which allows you to specify server controls. See the [IBM Tivoli Directory Server Client Programming for z/OS](https://www.ibm.com/support/docview.ws?rs=0&context=doc&contextarea=product&contentId=infoa000000000016c) for more information on the functions themselves, as well as on the parameters which have to be passed for particular requests.

Example 1: Initiate a new transaction

A transaction is initiated using the hcdTransactionControl with NEW as value. This control can be defined the following way:

```c
static LDAPControl hcdTransactionControl_new =
{
    "1.3.18.0.2.10.3", /* -- hcdTransactionControl -- */
    { 3, "\x4E\x45\x57\x00" }, /* -- NEW ------------- */
    LDAP_OPT_ON /* -- critical --------------- */
};
```

Note, as mentioned before, the value NEW (and also the values COMMIT and ROLLBACK, shown in example 3) have to be specified using UTF-8 encoding.

All controls to be passed to the IBM Tivoli Directory Server for z/OS are stored in an array. In this case, only one control is in the NULL-terminated array.

This array is now be passed to the function which sends the appropriate request to the IBM Tivoli Directory Server for z/OS. For example, `ldap_add_ext()` is used to request an add operation as follows:
Here, the control `hcdTC_new` is used where the `value=NEW` was specified in UTF-8.

If this request was successful and if a new transaction was started, the HCD LDAP backend sends back the control `hcdTransactionId` which contains the transaction ID. Such control may look similar to the following (see also Example 2):

```
static LDAPControl hcdTransactionId = {
   "1.3.18.0.2.10.4", /* -- hcdTransactionId ------ */
   { 1, "\x31\x00" }, /* -- TXN Id ---------------- */
   LDAP_OPT_ON /* -- critical -------------- */
};
static LDAPControl *hcdTC_Id[2] = { &hcdTransactionId, NULL };
```

In the above example, for the control `hcdTransactionId` the value indicates a transaction ID of 1.

Note, you should never generate a value for this control on your own. Instead, call the LDAP client API functions `ldap_result()` and `ldap_parse_result()` on the response of the first request to obtain the valid transaction ID.

After having issued `ldap_add_ext()`, calling `ldap_parse_result()` parses the results which were previously obtained by `ldap_result()`. Here, among other parameters, the LDAP control containing the transaction ID is provided.

The following code example demonstrates how to use both LDAP requests:

```
rc = ldap_result(ld, msgidp, 0, NULL, &LDAP_TXN_Msg);
rc = ldap_parse_result(ld, LDAP_TXN_Msg, &errcodep, &matcheddnp,
   &errmsgp, &referralsp, &servctrlsp, freeit);
```

The interesting parameter as far as controls are concerned is `servctrlsp`. This pointer locates an array such as `hcdTC_new`. The appropriate values of the control have to be copied into the `hcdTransactionId` control which can be done in the following way (assuming that `servctrlsp[0]` contains the `hcdTransactionId` control and the control `servctrlsp[0]` is not freed):

```
hcdTransactionId.ldctl_value.bv_val = (servctrlsp[0])->ldctl_value.bv_val;
hcdTransactionId.ldctl_value.bv_len = (servctrlsp[0])->ldctl_value.bv_len;
```

After these values have been copied, the correct transaction ID can be provided for further LDAP requests which are part of this transaction. This is shown in Example 2 below.

**Example 2: Submit further LDAP requests of the transaction.**

After a transaction was successfully initiated, further LDAP requests can be added to the transaction. Here, the user must provide an `hcdTransactionId` control where `value` contains the correct transaction ID. As stated before, this control is provided by the HCD LDAP backend and must be used for further requests belonging to this specific transaction.

If we assume that for the transaction we just initiated, the transaction ID number is 1, then, the control which was provided by the HCD LDAP backend would be as follows:

```
static LDAPControl hcdTransactionId = {
   "1.3.18.0.2.10.4", /* -- hcdTransactionId ------ */
```
In general, the values 1 and "\x31\x00" have to be replaced by the correct values of the control that is provided by the call of `ldap_parse_result()` (see example 1).

After the correct values for this transaction have been copied into the above control, further LDAP requests can be added to the transaction using this control. The call of such an LDAP request is just the same as in example 1 with the only exception that the controls have been exchanged as follows:

```c
rc = ldap_add_ext(ld, dn, pmods, hcdTC_Id, NULL, &msgidp);
```

As the transaction ID does not change for a specific transaction, all further requests belonging to this transaction must use the same `hcdTransactionId` control.

### Example 3: End a transaction

Here, two different cases must be considered. As stated before, a transaction can be committed for execution, or it can be aborted by the user using a rollback request.

It is recommended that you explicitly rollback a transaction if a single LDAP request returns a bad return code. The reason for this is that, transactions are intended to represent logical units of requests which belong together. In principal, both the actions - commit and rollback - are the same from an implementation point of view. The only difference is, which control is to be specified for the LDAP request that finishes the transaction. If the user wants to commit a transaction, the following control must be defined:

```c
static LDAPControl hcdTransactionControl_commit =
{
    "1.3.18.0.2.10.3", /* -- hcdTransactionControl -- */
    { 6, "\x43\x4F\x4D\x4D\x49\x54\x00" }, /* - COMMIT -- */
    LDAP_OPT_ON /* -- critical --------------- */
};
```

If the user want to rollback a transaction, the following control must be defined:

```c
static LDAPControl hcdTransaction_rollback =
{
    "1.3.18.0.2.10.3", /* -- hcdTransactionControl -- */
    { 8, "\x52\x4F\x4C\x4C\x42\x41\x43\x4B\x00" }, /* ROLLBACK */
    LDAP_OPT_ON /* -- critical --------------- */
};
```

Using these controls, the final request will, in the case of commit, now be:

```c
rc = ldap_add_ext(ld, dn, pmods, hcdTC_commit, NULL, &msgidp);
```

and in the case of a rollback:

```c
rc = ldap_add_ext(ld, dn, pmods, hcdTC_rollback, NULL, &msgidp);
```

If you look at the previous definition of `hcdTC_commit` and `hcdTC_rollback`, both parameters are control arrays that contain the commit or rollback control itself, and in addition, also contain the control `hcdTC_Id` for submitting the next request. Hence, one control indicates the end of the transaction, and the other control
identifies the transaction on the basis of its ID. Once a transaction is closed, no further requests can be added to this control.
Appendix A. How to navigate through the dialog

This appendix illustrates the flow from the options on the HCD Primary Task Selection panel and the various actions that can be taken from each option. The panels that appear for option 1 provide an action bar at the top and a context menu for each object. Figure 137 on page 372 shows the general action bar valid for the action list panels. Some action list panels offer special action bar choices that are not shown in the figure (for example the Show/Hide action bar choice on the Device List panel invoked from the Operating System List panel). Figure 141 on page 373 to Figure 145 on page 374 shows the options on the context menu and the navigation possibilities to other panels. Almost all of the options on the context menu can be directly selected by entering the action code next to the item in the list to be selected. Available action codes are shown in parentheses below the “Options Available” heading.

Figure 137. HCD Primary Task Selection panel

![HCD Primary Task Selection panel]

Figure 138. HCD - Edit profile options - Option 0

![HCD - Edit profile options - Option 0]
Figure 139. HCD - Define, Modify, or View Configuration Data - Option 1

<table>
<thead>
<tr>
<th>Task Menu</th>
<th>Action Bar</th>
<th>Options Available</th>
</tr>
</thead>
</table>
| 1. Define, Modify, or View Config. Data | N/A | 1. Operating system configurations  
consoles  
system-defined generics  
EDTs  
esoterics  
user-modified generics  
2. Switches  
ports  
switch configurations  
port matrix  
3. Processors  
channel subsystems  
partitions  
channel paths  
4. Control units  
5. I/O devices |

Figure 140. HCD - Generic Action Bar Options

<table>
<thead>
<tr>
<th>Task Menu</th>
<th>Action Bar</th>
<th>Options Available</th>
</tr>
</thead>
</table>
| 1. Define, Modify, or View Config. Data | Action List Panel | Operating systems  
EDTs ...  
Esoterics ...  
Generics ...  
Consoles ...  
Switches  
Ports ...  
Switch Configurations ...  
Port Matrix ...  
Processors  
Partitions ...  
CHPIDs ...  
Control units  
Devices  
Migrate configuration data  
Leave HCD |
| Action List Panel | Filter | Set Filter  
Clear Filter |
| Action List Panel | Backup | with prompt ...  
without prompt |
| Action List Panel | Query | List supported processors  
List supported switches  
List supported control units  
List supported devices  
List installed UIMs  
View message log  
View activity log  
View IODF information |
| Action List Panel | Help | Help for help  
Extended help  
Keys help  
Instruction help |
Figure 146. HCD - Utility Functions - Options 2 - 7
Appendix B. Configuration reports

This appendix shows examples of the configuration reports that can be produced by HCD:

- "Textual configuration reports"
- "Graphical configuration reports" on page 435
- "IODF compare reports" on page 436

Textual configuration reports

This section shows examples for textual configuration reports. "Print configuration reports" on page 235 lists the different report types available and describes how to produce them.

Channel subsystem reports

The following channel subsystem reports are available:

- "Processor Summary Report"
- "Channel Subsystem Summary Report" on page 378
- "Partition Report" on page 378
- "IOCDS Report" on page 379
- "Channel Path Summary Report" on page 380
- "Channel Path Detail Report" on page 381
- "CF Channel Path Connectivity Report" on page 383
- "Control Unit Summary Report" on page 384
- "Control Unit Detail Report" on page 384
- "Device Summary Report" on page 387
- "Device Detail Report" on page 387

Processor Summary Report

<table>
<thead>
<tr>
<th>PROCESSOR</th>
<th>TYPE</th>
<th>MODEL</th>
<th>CONFIG.</th>
<th>SERIAL</th>
<th>DESCRIPTION</th>
<th>NETWORK NAME</th>
<th>CPC NAME</th>
<th>LSYSTEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERV01WW</td>
<td>S38</td>
<td>LPAR</td>
<td>2094</td>
<td>1212122094</td>
<td>mix system</td>
<td>ERV01</td>
<td>ERV01</td>
<td>ZZZZZZZZ</td>
</tr>
<tr>
<td>ERV0201</td>
<td>ZX7</td>
<td>LPAR</td>
<td>9672</td>
<td>4434349672</td>
<td>old system 1</td>
<td>IBM390PS</td>
<td>ERV0201</td>
<td>IBM390PS</td>
</tr>
<tr>
<td>ERV0202</td>
<td>ZX7</td>
<td>LPAR</td>
<td>9672</td>
<td>1111119672</td>
<td>old system 2</td>
<td>IBM390PS</td>
<td>ERV0202</td>
<td>IBM390PS</td>
</tr>
<tr>
<td>ERV05A1</td>
<td>S28</td>
<td>LPAR</td>
<td>2094</td>
<td>9912992094</td>
<td>stand alone</td>
<td>IBM390PS</td>
<td>ERV05A1</td>
<td>IBM390PS</td>
</tr>
<tr>
<td>ERV05A2</td>
<td>S54</td>
<td>LPAR</td>
<td>2094</td>
<td>9929912094</td>
<td>stand alone</td>
<td>IBM390PS</td>
<td>ERV05A2</td>
<td>IBM390PS</td>
</tr>
<tr>
<td>NET01</td>
<td>R07</td>
<td>LPAR</td>
<td>2096</td>
<td>332232096</td>
<td>Network System</td>
<td>ZZZZZZZZ</td>
<td>NET01</td>
<td></td>
</tr>
</tbody>
</table>

Figure 147. Processor Summary Report
CONFIG. MODE

Indicates the operation mode in which a processor may operate. These modes are:

**BASIC** The processor is not logically partitioned.

**LPAR** The processor is logically partitioned. Several operating systems may run concurrently in different partitions of the processor.

**SNA ADDRESS** The SNA Address consists of Network name and CPC name and associates the CPC and the processor definition in the IODF.

**SUPPORT LEVEL** Shows the ID of the processor support level and an enumeration of the provided functionality.

### Channel Subsystem Summary Report

Only XMP processors will have a channel subsystem report which shows the defined channel subsystems.

<table>
<thead>
<tr>
<th>CSS ID</th>
<th>MAXIMUM</th>
<th>ACTUAL</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1000</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1000</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>800</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>900</td>
<td>32</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** This report and the following channel subsystem reports show the processor token only if the IODF is a production IODF.

### Partition Report

<table>
<thead>
<tr>
<th>NUMBER</th>
<th>USAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>OS/CF</td>
<td>OS/CF Partition number 2 CSS0</td>
</tr>
<tr>
<td>3</td>
<td>OS</td>
<td>OS Partition number 3 CSS0</td>
</tr>
<tr>
<td>4</td>
<td>CF/OS</td>
<td>reserved Partition number 4 CSS0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NUMBER</th>
<th>USAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>OS</td>
<td>Linux</td>
</tr>
<tr>
<td>A</td>
<td>OS</td>
<td>Linux</td>
</tr>
<tr>
<td>6</td>
<td>OS</td>
<td>Linux</td>
</tr>
<tr>
<td>8</td>
<td>OS</td>
<td>Linux</td>
</tr>
<tr>
<td>9</td>
<td>OS</td>
<td>Linux</td>
</tr>
</tbody>
</table>

**Figure 148. Channel Subsystem Summary Report**

**Figure 149. Partition Report**

**NUMBER** Is the partition number (MIF ID). This information is printed only for EMIF capable processors.

**USAGE** Specifies the usage type of a partition: CF indicates a partition supporting coupling facility. OS indicates a partition running an operating system. CF/OS indicates a partition supporting coupling facility or running an operating system.
IOCDS Report

IOCDS data are retrieved from the support element when a SNA address is defined. Otherwise, the IOCDS data are retrieved from the IODF. An IOCDS status line at the end of the report indicates the source of the IOCDS data.

<table>
<thead>
<tr>
<th>IOCDS NAME</th>
<th>FORMAT</th>
<th>STATUS</th>
<th>IOCDS/HSA</th>
<th>IOCDS/Proc</th>
<th>Protect</th>
<th>DATE</th>
<th>TIME</th>
<th>IOCDS Configuration Token Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>A0 316ACFS</td>
<td>LPAR</td>
<td>POR</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes-POR 2004-11-12 06:14 CFS 16:27:52 04-11-11 SY54 100F71</td>
</tr>
<tr>
<td>A1 063ACFS</td>
<td>LPAR</td>
<td>Alternate</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>2004-03-03 16:49 CFS 16:23:49 04-03-03 SY54 100F72</td>
</tr>
<tr>
<td>A2 091ACFS</td>
<td>LPAR</td>
<td>Alternate</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>2004-04-01 00:01 CFS 16:08:02 04-03-31 SY54 100F73</td>
</tr>
<tr>
<td>A3 296ACFS</td>
<td>LPAR</td>
<td>Alternate</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>2004-10-22 11:45 CFS 14:59:05 04-10-21 SY54 100F71</td>
</tr>
</tbody>
</table>

Figure 150. IOCDS Report

- **NAME**: Represents the user-defined name of the IOCDS (derived from the MSG1 parameter)
- **FORMAT**: IOCDS format (BASIC or LPAR)
- **STATUS**: Indicates the status of the IOCDS: Alternate, POR, Invalid (see “Build S/390 microprocessor IOCDSs” on page 203)
- **Token Match - IOCDS/HSA**: Indicates whether the IOCDS token matches the current HSA token
- **Token Match - IOCDS/Proc**: Indicates whether the IOCDS token matches the current processor token in the IODF
- **Write Protect**: Indicates whether the IOCDS is write-protected (Yes) or not (No), or currently write-protected because it is the POR IOCDS (Yes-POR).
- **Last Update DATE/TIME**: Time stamp of IOCDS creation time
- **IOCDS Configuration Token Information**: is the configuration token information stored in the support element and shows the relationship between the IOCDS and the production IODF from which it was created.
Channel Path Summary Report

If applicable, for spanned channels, there is a separate sub-report after the partition table of a processor which shows the connection of spanned channel paths to channel subsystems.

<table>
<thead>
<tr>
<th>PARTITION</th>
<th>NUMBER</th>
<th>NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>GECSS01X</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>GECSS02X</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>GECSS03X</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>GECSS04X</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>GECSS05X</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>GECSS06X</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>GECSS07X</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>GECSS08X</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>GECSS09X</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>GECSSF0X</td>
</tr>
</tbody>
</table>

Figure 151. Channel Path Summary Report

PCHID AID/P
For external channels: designates the physical channel identifier (PCHID) or, if applicable, designates the HCA adapter ID (AID) and the HCA port (P).

DIS QP
Indicates whether queue prioritization is disabled.

MFS
Designates the maximum frame size in KB.

MNGD
Indicates whether the channel path is managed.

I/O CLUSTER
The I/O cluster name for managed channel paths.

DYN. SWITCH
Designates the switch holding the dynamic connection.

SWITCH ID
Designates the switch the channel is physically plugged in (entry switch).

PORT
Designates the entry port on the entry switch.

MODE
Operation mode of the channel path.
**Channel Path Detail Report**

The Channel Path Detail Report lists the channel paths defined per processor with their attributes and attachment information.

In addition, the switch connections on the path between the CHPID and the control unit are shown if they can be determined by HCD. The entry switch and port of the CHPID are always shown.

For an entry switch of a CHPID which is defined as a dynamic switch, the control unit port is shown if it is compatible with the link address defined for the CHPID. For an entry switch of the CHPID which is defined as a dedicated switch, the control unit port or the ports connecting the switches are only shown if switch configurations are defined which allow HCD determining a valid path between CHPID and control unit.

In case of chained switches, the first print line for a channel path shows the switch the channel path is physically plugged in. The second print line shows the switch the control unit is connected to. In addition, the first print line shows the ID of the switch with the dynamic connection.

Each attached control unit of a specific channel path is separated by a line.

The first print line for an attached control unit shows the first unit address range defined for the control unit. If there are more unit address ranges defined for a control unit, these are shown in the following print lines. Together with the unit address range(s) of the control unit the attached devices of the control unit are shown grouped according to device types and consecutive numbers and unit addresses. The report shows the starting device number and range of the device group. The unit address describes the address of the first device in the range.

For FICON switches, the dynamic switch ID is empty. The control unit port ID is shown as a two-byte port address (the port ID prefixed by the switch address) when used in a cascaded switch environment, or as a one-byte port address otherwise. For cascaded FICON switches, only the channel path port ID and the control unit port ID are shown, but no connection between the switches.

**Note:** For a Coupling Facility control unit, all CF devices attached to this control unit are listed, not only those devices that are defined for the connected coupling facility channel path described in the row.
### Figure 152. Channel Path Detail Report

<table>
<thead>
<tr>
<th>PROCESSOR ID</th>
<th>EVA CSS ID</th>
<th>TYPE</th>
<th>MODEL</th>
<th>CONFIGURATION MODE</th>
<th>LPAR</th>
<th>TIME: 12:57</th>
<th>DATE: 2003-01-15</th>
<th>PAGE E-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 05F</td>
<td>CNC</td>
<td>NO</td>
<td>SHR</td>
<td>00 04 06 00 0000 2105</td>
<td>0</td>
<td>00 00 0000,15</td>
<td>00 01</td>
<td>3390A 3390B</td>
</tr>
<tr>
<td>00 04 00 00 0003 3490</td>
<td>00 0F 0100,2 00 3490</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00 04 FE 00 0B00 9032-5</td>
<td>00 00 0B00 00 9032-5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01 13A</td>
<td>CNC</td>
<td>NO</td>
<td>DED</td>
<td>00 05 00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 2D3</td>
<td>CNC</td>
<td>NO</td>
<td>DED</td>
<td>00 0C 0E 00 0005 2105</td>
<td>00 00</td>
<td>00 0700,16 00 3380B</td>
<td></td>
<td></td>
</tr>
<tr>
<td>03 2E7</td>
<td>IQD</td>
<td>NO</td>
<td>SPAN</td>
<td>00 OA 100</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>04 3A8</td>
<td>CBP</td>
<td>NO</td>
<td>DED</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SWITCH ID**
Designates the entry switch the channel path is physically plugged in. For chained ESCON switches or cascaded FICON switches, a second line is shown with the ID of the switch to which the control unit is connected.

**SWITCH PR PN**
Designates the entry port of the entry switch. In case of an ESCON chained switch, it designates the entry port of the chained switch.

**SWITCH CU PN**
Designates the port the control unit is connected to. In case of an ESCON chained switch, it designates the port of the entry switch to which the chained switch is connected. For a cascaded FICON switch, the port ID is prefixed by the switch address.

**SWITCH DYN ID**
For an ESCON environment, it designates the switch holding the dynamic connection.
**CF Channel Path Connectivity Report**

For XMP processors, there will be a single CF Channel Path Connectivity Report. The CHPID numbers are prefixed by the channel subsystem ID. For example, 1.0C denotes CHPID 0C from CSS 1. If a spanned channel path is used, the CHPIDs are reported from all channel subsystems together with the accessed partitions in the channel subsystems.

<table>
<thead>
<tr>
<th>CHPID TYPE</th>
<th>MODE</th>
<th>O</th>
<th>LIST</th>
<th>CHPID TYPE</th>
<th>MODE</th>
<th>O</th>
<th>LIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>ICP</td>
<td>SPAN</td>
<td>N</td>
<td>TRX1</td>
<td>T29</td>
<td>2094-S18</td>
<td>1.01</td>
</tr>
<tr>
<td>3.00</td>
<td>TRX2</td>
<td>T29CFA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.01</td>
<td>ICP</td>
<td>SPAN</td>
<td>N</td>
<td>TRX1</td>
<td>T29</td>
<td>2094-S18</td>
<td>1.00</td>
</tr>
<tr>
<td>3.00</td>
<td>TRX2</td>
<td>T29CFA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.02</td>
<td>ICP</td>
<td>SPAN</td>
<td>N</td>
<td>TRX1</td>
<td>T29</td>
<td>2094-S18</td>
<td>1.03</td>
</tr>
<tr>
<td>3.02</td>
<td>TRX2</td>
<td>T29CFA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.03</td>
<td>ICP</td>
<td>SPAN</td>
<td>N</td>
<td>TRX1</td>
<td>T29</td>
<td>2094-S18</td>
<td>1.02</td>
</tr>
<tr>
<td>3.03</td>
<td>TRX2</td>
<td>T29CFA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 153. CF Channel Path Connectivity Report**

- **SOURCE/DESTINATION CHPID** - The identifier of the source/destination channel path.
- **SOURCE/DESTINATION TYPE** - Is the type of the source/destination channel path.
- **SOURCE/DESTINATION MODE** - Is the operation mode of the source/destination channel path.
- **SOURCE O** - CHPID is identified as occupied
- **SOURCE/DESTINATION ACCESS LIST** - Shows those partitions the source/destination channel path has in its access list.
- **SOURCE/DESTINATION CANDIDATE LIST** - Shows those partitions the source/destination channel path has in its candidate list.
- **DESTINATION PROCESSOR ID** - Is the name of the processor the destination channel path is defined for.
- **DESTINATION TYPE-MODEL** - Is the type-model of the destination processor.
- **SOURCE CNTRL UNIT** - Is the number of the CF control unit used for the source CF channel path connection.
- **SOURCE DEVICE NUM,RANGE** - Are the numbers of the CF devices and ranges of device groups defined for the source CF channel path connections via the source CF control unit.
- **DESTINATION CNTRL UNIT** - Is the number of the CF control unit used for the destination CF channel path connection.
- **DESTINATION DEVICE NUM,RANGE** - Are the numbers of the CF devices and ranges of device groups used for the destination CF channel path connections via the destination CF control unit.
- **CNTL TYPE** - Indicates the type of the connecting control unit(s).
## Control Unit Summary Report

<table>
<thead>
<tr>
<th>CONTROL UNIT NUMBER</th>
<th>TYPE-MODEL</th>
<th>SERIAL NUMBER</th>
<th>DESCRIPTION</th>
<th>CONNECTED SWITCH ID. PORT NUMBER</th>
<th>LOGICAL PATHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0003 OSC</td>
<td></td>
<td></td>
<td>CU for OSC Channels</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0070 OSN</td>
<td></td>
<td></td>
<td>for all native MVS (also NETMVS)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0120 OSA</td>
<td></td>
<td></td>
<td>TAPES 0480 CT1 BL6</td>
<td>0A.CB</td>
<td></td>
</tr>
<tr>
<td>0480 FCP</td>
<td></td>
<td>11333</td>
<td>Open FCP 16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0900 1750</td>
<td>DS6K1 (0900-0940)</td>
<td>80</td>
<td>17.54 17.58 17.5C 17.60</td>
<td>72</td>
<td>1A.CB</td>
</tr>
<tr>
<td>0950 1750</td>
<td>DS6K1 (0950-0998)</td>
<td>80</td>
<td>17.54 17.58 17.5C 17.60</td>
<td>72</td>
<td></td>
</tr>
<tr>
<td>0820 2105</td>
<td>ESS22 B20 CUJ (3390-1/27) YSE</td>
<td>152</td>
<td>17.45 17.49</td>
<td>34</td>
<td>0</td>
</tr>
<tr>
<td>08C0 2105</td>
<td>ESS22 BCO CU4 (3390-nmix) VSE</td>
<td>144</td>
<td>17.45 17.49</td>
<td>34</td>
<td>0</td>
</tr>
<tr>
<td>0C50 2105</td>
<td>ESS22 CSS CU5 (3390-nmix) YSE</td>
<td>176</td>
<td>17.45 17.49</td>
<td>34</td>
<td>0</td>
</tr>
<tr>
<td>0D80 2107</td>
<td>DS8K0 CUO(D80-DCF) 3390-MIX</td>
<td>80</td>
<td>17.44 17.48 17.4C 17.50</td>
<td>424</td>
<td>0</td>
</tr>
<tr>
<td>0E20 2107</td>
<td>DS8K0 CUZ(E20-E3F) 3390-S4</td>
<td>32</td>
<td>17.44 17.48 17.4C 17.50</td>
<td>424</td>
<td>0</td>
</tr>
<tr>
<td>0E40 2107</td>
<td>DS8K0 CUZ(E40-5EF) 3390-S4</td>
<td>32</td>
<td>17.44 17.48 17.4C 17.50</td>
<td>424</td>
<td>0</td>
</tr>
<tr>
<td>0E60 2105</td>
<td>ESS22 E60 CUO (3390-9) VSE</td>
<td>96</td>
<td>17.45 17.49</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0E80 2105</td>
<td>ESS22 ECO CU4 (3390-1/27) YSE</td>
<td>152</td>
<td>17.45 17.49</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1010 2032</td>
<td>McData E6064 z/OS</td>
<td>10.FE</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1011 2032</td>
<td>McData E6140 z/OS</td>
<td>11.FE</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1028 FCTC</td>
<td>H053E3F</td>
<td>FCTC H05LP04 to H05LP11 + TX1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1028 FCTC</td>
<td>H053E3F2</td>
<td>FCTC H05LP30 to H05LP31</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1128 FCTC</td>
<td>H053E3F</td>
<td>FCTC H05LP04 to H05LP11</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1240 OSA</td>
<td></td>
<td>OSA (1240-125E,5F)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

### Figure 154. Control Unit Summary Report

<table>
<thead>
<tr>
<th>CONNECTED SWITCH ID. PORT NUMBER</th>
<th>Describes where the CU is physically connected to (switch and port)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOGICAL PATHS</td>
<td>Displays the number of defined logical paths per control unit.</td>
</tr>
</tbody>
</table>

## Control Unit Detail Report

The Control Unit Detail Report lists all control units defined in the currently accessed IODF, with their attributes and attachment information.

For each control unit all processors are shown, where the control unit is attached to. For each of those processors, all control units, that are part of the "logical control unit" the currently printed control unit belongs to, are printed. Also, the currently shown control unit is listed again to show information about switch connection, channel path attachment, and attached devices. The information about each logical control unit the currently printed control unit belongs to, is separated by a dotted line.

The switch connection information is shown in the Control Unit Detail Report even if no switch configuration is defined. For example:

If a control unit is connected to a switch, but no complete path is defined through the switch (that means, no valid path is defined through the switch, which allows a dynamic connection) the complete switch connection information with SWITCH ID, CU PN, and PR PN can be determined.

In case of chained switches, the first information in a print line is shown for the switch that the channel path is connected to. Note that this is different from the Channel Path Detail report.
To get information about the switch with the dynamic connection (in case of chained ESCON switches), the Channel Path Detail Report has to be produced.

For each control unit belonging to a logical control unit, the channel path it is attached to, is printed together with the link address (if applicable).

For FICON cascade switching, the link address will be shown as a two-byte number. If a path from the control unit is via cascaded FICON switches, the link between the FICON switches is not shown. Instead, the control unit port of the switch connected to the control unit and the CHPID port of the switch connected to the channel path are shown. If no port connections are defined, switch data is extracted from the dynamic switch and the link address. Port IDs are shown as one-byte port addresses.

The ‘DEVICE’ column shows the devices which are attached to the control unit printed under column ‘CU IN LCU’. If the control unit is attached to more than one channel path, the information about the attached devices is printed together with the last printed channel path. The devices attached to a control unit are grouped according to consecutive number. The report shows the starting device number and range of the device group.

Columns ‘LOG. PATHS PER CU’ and ‘LOGICAL PATHS PER CU PORT’ display the number of defined logical paths.
### Figure 155. Control Unit Detail Report

<table>
<thead>
<tr>
<th>CONTROL UNIT NUMBER</th>
<th>PROCESSOR.CSS ID</th>
<th>CONTROL UNIT DETAIL REPORT</th>
<th>TIME: 16:12</th>
<th>LOG. PATHS PER CU</th>
<th>CU IN LCU</th>
<th>CU PN</th>
<th>IOCL ADD</th>
<th>SWITCH ID</th>
<th>CU PN</th>
<th>PR PN</th>
<th>CHAINED/CASC (SWITCH ID, CU PN, PR PN)</th>
<th>CHPID . LINK ADDR</th>
<th>-- DEVICE -- NUMBER,RANGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0950 1750</td>
<td>PROCO3.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PROCO3.2</td>
<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>0820 2105</td>
<td>PROCO3.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PROCO3.1</td>
<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>PROCO3.2</td>
<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>PROCO3.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0B20 2105</td>
<td>PROCO3.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PROCO3.1</td>
<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>PROCO3.2</td>
<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>PROCO3.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0B00 2105</td>
<td>PROCO3.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PROCO3.1</td>
<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>PROCO3.2</td>
<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>PROCO3.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0C50 2105</td>
<td>PROCO3.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PROCO3.1</td>
<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>PROCO3.2</td>
<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>PROCO3.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

- **PROCESSOR.CSS ID** Designates the processors, and in case of an XMP processor, the channel subsystem to which the CU is attached.
- **LOG. PATHS PER CU** Specifies the number of defined logical paths for a control unit per channel subsystem.
- **CU IN LCU** Designates which CUs belong to the logical CU.
- **IOCL** Designates I/O concurrency level (same as SHARED in IOCP). Specifies the level of concurrency of I/O requests that the parallel channel path allows for the control unit (CU).
  - 1 One I/O request at a time. (SHARED=Y)
  - 2 Multiple I/O requests at a time. (SHARED=N)
- **CU-ADD** Designates the CU address.
- **SWITCH ID** Designates the switch the CU is connected to.
- **CU PN** Designates the port the CU is physically connected to.
- **PR PN** Designates the entry port of the channel (except in the case of chained switches).
- **CHAINED/CASC (SWITCH ID, CU PN, PR PN)** Designates a possible second switch the CHPID is connected to.
- **CHPID . LINK ADDR** Designates the channel path and the one- or two-byte link address to which the control unit is connected. 
  
  - (*) in this column indicates that the control unit is connected to 
    
  - managed channel paths.
- **DEVICE NUMBER,RANGE** Specifies information about the devices that are attached to the processors by the control unit.
LOGICAL PATHS PER CU PORT

Specifies the number of defined logical paths for all control units connected to a specific port.

Device Summary Report

The Device Summary Report gives you an overview of the devices defined in the currently accessed IODF and their attaching control units. The devices are grouped according to the same characteristics. The report shows the starting device number and the range of the group.

For multi-exposure devices, the base devices are grouped separately from the non-base exposure devices.

--- DEVICE SUMMARY REPORT ---

<table>
<thead>
<tr>
<th>NUMBR</th>
<th>RANGE</th>
<th>TYPE</th>
<th>MODEL</th>
<th>ATTACHING CONTROL UNITS</th>
<th>SERIAL NUMBER</th>
<th>DESCRIPTION</th>
<th>VOLSER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0080,16</td>
<td>3174</td>
<td>0080</td>
<td>Merian SNA Connection</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0090,2</td>
<td>R56K</td>
<td>0110</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0092,2</td>
<td>R56K</td>
<td>0112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0094,2</td>
<td>R56K</td>
<td>0114</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0096,2</td>
<td>R56K</td>
<td>0116</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0110,2</td>
<td>R56K</td>
<td>0110</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0112,2</td>
<td>R56K</td>
<td>0112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0114,2</td>
<td>R56K</td>
<td>0114</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0116,2</td>
<td>R56K</td>
<td>0116</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01CD</td>
<td>3270-X</td>
<td>01CD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01C1,31</td>
<td>3270-X</td>
<td>01C0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0200,64</td>
<td>3390</td>
<td>200A 200B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0300,32</td>
<td>9345</td>
<td>300A 300B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0320,32</td>
<td>9345</td>
<td>320A 320B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0340,32</td>
<td>9345</td>
<td>340A 340B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0360,32</td>
<td>9345</td>
<td>360A 360B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 156. Device Summary Report

Device Detail Report

The Device Detail Report lists all devices defined in the currently accessed IODF, with their attributes and attachment information. Each device is shown with the processors to which it is attached. The device - processor attachment attributes are also listed. If applicable, the subchannel set where the device is located, is also shown in column SS. If not applicable, this column shows a blank.

For each processor the device is attached to, the CUs for the attachment to the processor, as well as the channel path(s) the CU is attached to, are also listed.

The starting device number and the range of subsequent device numbers are shown in one row. A range value of 1 is omitted.

The Device Detail Report shows a partition matrix which indicates whether a logical partition of a corresponding processor has access to the device either via the channel path access list or the channel path candidate list, and whether a partition is excluded or included via the device candidate list. Devices with a null-device candidate list are excluded from the report.

At the end of the report, for SMP processors, the totals for the following items are listed for each processor:

- CHPIDS
- PHYSICAL CONTROL UNITS
- SUBCHANNELS
- LOGICAL CONTROL UNITS
For the CHPID total, the report lists separate values for the total shared and the total unshared. For the physical CU total, HCD reports the total shared (those attaching to shared channel paths) and the total unshared.

For the subchannel and logical CU totals, the report lists separate values for the shared, unshared, and additional unshared counts that are, respectively, assigned and unassigned to a logical partition. The shared count is the total number assigned to shared channel paths. The generated for LPAR unshared count is the total that would have been generated for a basic IOCDS. The additional unshared count contains the unshared values that were generated for the LPAR IOCDS.

The TOTAL value is the total that would have been contained in the generated IOCDS. The HSA TOTAL is the total that will exist in the HSA after POR. At the completion of POR, the HSA may contain more subchannels and logical CUs than does the IOCDS.

For XMP processors, the following totals are reported:

- CHPIDS
- PHYSICAL CONTROL UNITS
- DEVICES
- LOGICAL CONTROL UNITS

The column CSS TOTAL lists the number of CHPIDs, physical and logical control units and the maximum number of devices that are currently defined for that channel subsystem.

The column IOCDS TOTAL lists the number of CHPIDs, physical and logical control units and the maximum number of devices without definitions caused by over-defined CHPIDs.

Columns HSA TOTAL and HSA LIMIT are not applicable for XMP processors.

The column USER LIMIT lists the maximum number of devices defined by the user for that channel subsystem.

The column SUPPORTED LIMIT lists the maximum number of CHPIDs, physical and logical control units and the maximum number of devices that are supported for the processor for that channel subsystem.
Switch reports

The following switch reports are available:

- “Switch Summary Report” on page 390
- “Switch Detail Report” on page 390
- “Switch Configuration Summary Report” on page 390
- “Switch Configuration Detail Report” on page 391
Switch Summary Report

<table>
<thead>
<tr>
<th>SWITCH ID</th>
<th>TYPE</th>
<th>ADDR</th>
<th>NUMBER</th>
<th>DESCRIPTION</th>
<th>CU NUMBER</th>
<th>DEVICE NUMBER</th>
<th>INSTALLED PORTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>2032</td>
<td>41</td>
<td>55-9999</td>
<td>switch address 41 fabric a</td>
<td>0098</td>
<td>0098</td>
<td>33</td>
</tr>
<tr>
<td>98</td>
<td>9033</td>
<td></td>
<td>55-9999</td>
<td>First switch</td>
<td>0098</td>
<td>0098</td>
<td>9</td>
</tr>
<tr>
<td>99</td>
<td>9032</td>
<td></td>
<td>55-9999</td>
<td>Second switch</td>
<td>0099</td>
<td>0099</td>
<td>29</td>
</tr>
</tbody>
</table>

Figure 158. Switch Summary Report

ADDR Shows the switch address, if available.
CU NUMBER Shows all switch control units attached to the switch CU port of the switch.
DEVICE NUMBER Shows all switch devices defined for a switch.

Switch Detail Report

If the switch is connected to an XMP processor, the processor ID is qualified with the channel subsystem ID. A spanned channel path is suffixed with an asterisk (*).

<table>
<thead>
<tr>
<th>SWITCH ID</th>
<th>TYPE</th>
<th>PORT NAME</th>
<th>CONNECTION UNIT</th>
<th>UNIT ID</th>
<th>UNIT TYPE</th>
<th>OCCUPIED</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>9032</td>
<td>AE</td>
<td>CU 053E</td>
<td>3490</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>9032</td>
<td>AF</td>
<td>SW 99 PORT FB 9033</td>
<td>NO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>9032</td>
<td>B0</td>
<td>CU 0400</td>
<td>3190</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>9032</td>
<td>B1</td>
<td>PROCA_CP18</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>9032</td>
<td>B2</td>
<td>PROCA_CP19</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>9032</td>
<td>B3</td>
<td>CU 0900</td>
<td>3990</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>9032</td>
<td>B4</td>
<td>PR G29.1 CHPID 12 2084-C24</td>
<td>NO</td>
<td>CT 1060 SCTC</td>
<td></td>
</tr>
</tbody>
</table>

Figure 159. Switch Detail Report

CONNECTION UNIT Specifies the type of the unit the port is connected to.
PR = Processor, CU = Control Unit, SW = Switch
OCCUPIED Indicates a port connection external to the IODF.

Switch Configuration Summary Report

<table>
<thead>
<tr>
<th>SWITCH ID</th>
<th>TYPE</th>
<th>CONFIG ID</th>
<th>DEFAULT CONNECTION</th>
<th>CONFIGURATION DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>98</td>
<td>9033</td>
<td>BASIC</td>
<td>ALLOW</td>
<td>night-shift configuration</td>
</tr>
<tr>
<td>99</td>
<td>9032</td>
<td>BASIC2</td>
<td>ALLOW</td>
<td></td>
</tr>
</tbody>
</table>

Figure 160. Switch Configuration Summary Report

DEFAULT CONNECTION Indicates what state a potential dynamic connection may have (allow or prohibit).
Switch Configuration Detail Report
The Switch Configuration Detail Report lists all supported ports of a switch with their dynamic connection attributes.

<table>
<thead>
<tr>
<th>SWITCH ID</th>
<th>SWITCH TYPE</th>
<th>CONFIGURATION ID</th>
<th>ALLOWED PORTS</th>
<th>BLOCKED PORTS</th>
<th>DEDICATED PORTS</th>
<th>ALLOWED CONNECTIONS</th>
<th>PROHIBITED CONNECTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
<td>9032-5</td>
<td>LW0010000</td>
<td>25</td>
<td>0</td>
<td>10</td>
<td>05-0A,0C-1F</td>
<td>04-05-07-1F</td>
</tr>
<tr>
<td>09</td>
<td>PR EVA.0</td>
<td>CHP 00 2084</td>
<td>NO</td>
<td>05-0A,0C-1F</td>
<td>04-06-1F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>PR EVA.0</td>
<td>CHP 01 2084</td>
<td>NO</td>
<td>04-06-1F</td>
<td>04-05-07-1F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>CU 0000</td>
<td>2105</td>
<td>NO</td>
<td>04-06-1F</td>
<td>04-05-07-1F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>PR EVA.0</td>
<td>CHP 02 2084</td>
<td>NO</td>
<td>04-06-1F</td>
<td>04-05-07-1F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>SW 01</td>
<td>P0 04 09032</td>
<td>NO</td>
<td>04-06-1F</td>
<td>04-05-07-1F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>CU 0003</td>
<td>3490</td>
<td>NO</td>
<td>04-06-1F</td>
<td>04-05-07-1F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>CU 0005</td>
<td>2105</td>
<td>NO</td>
<td>04-06-1F</td>
<td>04-05-07-1F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>SW 01</td>
<td>P0 05 04032</td>
<td>NO</td>
<td>04-06-1F</td>
<td>04-05-07-1F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>SW 01</td>
<td>P0 06 04032</td>
<td>NO</td>
<td>04-06-1F</td>
<td>04-05-07-1F</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 161. Switch Configuration Detail Report

Operating System reports
The following operating system reports are available:

- “Operating System Summary Report”
- “MVS Device Report”
- “MVS Device Detail Report” on page 393
- “Eligible Device Table Report” on page 393
- “NIP Console Report” on page 394
- “VM Device Report” on page 395
- “VM Device Detail Report” on page 395
- “VM Console Report” on page 396

Operating System Summary Report

| OPERATING SYSTEM SUMMARY REPORT | TIME: 13:37 DATE: 1997-03-02 PAGE 0- | 1 |
|---------------------------------|----------------------------------------|
| OPERATING SYSTEM ID | OPERATING SYSTEM TYPE | DESCRIPTION |
| OPYSYS01              | MVS                      | MVS operating system |
| OPYSYS02              | VM                       | VM operating system |

Figure 162. Operating System Summary Report

MVS Device Report
The MVS Device Report gives an overview of the devices defined to an MVS-type operating system in the currently accessed IODF.

The devices are grouped according to same characteristics. The report shows the starting device number and the range of a group.
### MVS Device Report

**Operating System Configuration ID:** OS000001

<table>
<thead>
<tr>
<th>DEV#, RANGE</th>
<th>TYPE-MODEL</th>
<th>SS</th>
<th>BASE</th>
<th>UCB-TYPE</th>
<th>ERP-NAME</th>
<th>DDT-NAME</th>
<th>MLT-NAME</th>
<th>OPT</th>
<th>UIM-NAME</th>
<th>ATI</th>
<th>AL</th>
<th>SH</th>
<th>SW</th>
<th>MX</th>
<th>MI</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000,16</td>
<td>3390A</td>
<td>0</td>
<td>0</td>
<td>3010200F</td>
<td>IECVDERP</td>
<td>IGDDOTA1</td>
<td>IEMLT02</td>
<td></td>
<td>CBDUS002</td>
<td>00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0010,16</td>
<td>3390A</td>
<td>1</td>
<td>0</td>
<td>3010200F</td>
<td>IECVDERP</td>
<td>IGDDOTA1</td>
<td>IEMLT02</td>
<td></td>
<td>CBDUS002</td>
<td>00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0020,16</td>
<td>3380A</td>
<td>0</td>
<td>0</td>
<td>3010200E</td>
<td>IECVDERP</td>
<td>IGDDT01</td>
<td>IEMLT02</td>
<td></td>
<td>CBDUS002</td>
<td>00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0020,16</td>
<td>3380A</td>
<td>1</td>
<td>0</td>
<td>3010200E</td>
<td>IECVDERP</td>
<td>IGDDT01</td>
<td>IEMLT02</td>
<td></td>
<td>CBDUS002</td>
<td>00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Key**

- **DEV#, RANGE:** Device number, count of devices (decimal)
- **TYPE-MODEL:** Device name
- **SS:** Subchannel set ID
- **BASE:** Base device number for multiple exposure devices
- **UCB-TYPE:** UCB type bytes
- **ERP-NAME:** Error recovery program
- **DDT-NAME:** Device descriptor table
- **MLT-NAME:** Module list table
- **OPT:** Optional MLT indicator
- **UIM-NAME:** Unit information module supporting the device
- **ATI:** Attention table index (UCBATI)
- **AL:** Alternate control unit (UCBALTCU)
- **SH:** Shared up option (UCBSHRUP)
- **SW:** Device can be swapped by DDR (UCBSWAPF)
- **MX:** Device has multiple exposures (UCBMTPXP)
- **MI:** MIH processing should be bypassed (UCBMIHBP)
- **O:** MLT is optional
- **Y:** Device supports this feature
- **blank:** Device does not support this feature

<table>
<thead>
<tr>
<th>Class Name</th>
<th>Class Type</th>
<th>Device Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAPE</td>
<td>80</td>
<td>0</td>
</tr>
<tr>
<td>Communication Devices</td>
<td>40</td>
<td>0</td>
</tr>
<tr>
<td>C-T-C</td>
<td>41</td>
<td>0</td>
</tr>
<tr>
<td>DASD</td>
<td>20</td>
<td>64</td>
</tr>
<tr>
<td>Graphics</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>Unit Record</td>
<td>08</td>
<td>0</td>
</tr>
<tr>
<td>Character Readers</td>
<td>04</td>
<td>0</td>
</tr>
<tr>
<td>Total number of I/O devices defined by this I/O configuration</td>
<td></td>
<td>64</td>
</tr>
</tbody>
</table>

**Figure 163. MVS Device Report**
MVS Device Detail Report

PARAMETER
Shows the parameter values specified for the devices. If you do not specify "Yes" or "No" for devices that support the dynamic capability, the DYNAMIC parameter will not be displayed.

FEATURE
Shows the features given to these devices.

Eligible Device Table Report

Figure 164. MVS Device Detail Report

Figure 165. Eligible Device Table Report
NAME TYPE describes the type of the device groups contained in the EDT:

- GENERIC generic device type group
- ESOTERIC esoteric device group
- G/Generic system generated generic device type group
- G/ESOTERIC system generated esoteric device group

VIO Eligible for virtual I/O, designates temporary data sets that exist in paging storage only.
PREF Preference value, indicates the order the system should follow when attempting allocation.
AFFINITY INDEX This index is used by the system allocation programs to determine which devices have
affinity to each other. Devices have affinity if either of the following statements are true:

- The devices have the same affinity index.
- The affinity index for one of the devices is a subset of the other devices’ affinity index.

An affinity index is a subset of another if both of the following statements are true:

- Neither index has a value of X’FFFF’.
- One or more bits in one index are set to a binary "one" and one or more corresponding
  bits in the other index are set to a binary "one".

ALLOCATION DEVICE TYPE UCB device table for allocation entry.
ASSOCIATED GENERICS Indicates the relation of a device type to generics.
DEVICE NUMBER LISTS Lists the devices that are included in the group. The devices are grouped according to
subsequent device numbers. The report shows the range of the device group.

Note: Device ranges in a subchannel set with a subchannel set ID > 0 are displayed in a
5-digit notation with the leading digit indicating the subchannel set ID. For example, a
device range 1000-103F located in subchannel set 1 is shown as 11000-1103F. A device range
2000-203F in subchannel set 0 is shown as 2000-203F.

NIP Console Report

<table>
<thead>
<tr>
<th>DEVICE #</th>
<th>TYPE-MODEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>3278-3</td>
</tr>
</tbody>
</table>

Figure 166. NIP Console Report
VM Device Report

The VM Device Report gives an overview of the devices defined to a VM operating system in the currently accessed IODF.

The devices are grouped according to the same characteristics. The report shows the starting device number and the range of a group.

---

**Figure 167. VM Device Report**

**VM Device Detail Report**

---

**Figure 168. VM Device Detail Report**

**PARAMETER**
Shows the parameter values specified for the devices.

**FEATURE**
Shows the features given to these devices.
Diagnostic messages

The following example shows you messages that might be returned with the report. The diagnostic messages are sorted by severity. For each connection, HCD displays only one message, even if the connection includes several errors. You first have to correct the first error before the next message is displayed. HCD displays the messages according to the priority as described in "Displaying..."
The I/O Path report shows the physically sensed I/O paths (with physical types) of the active system compared with the logical definitions of the paths (also the object types) of a specific IODF.

- If the sensed I/O path reports a switch, the verification assumes that this is a dynamic switch. It checks whether the defined I/O path in the IODF contains a dynamic switch, and whether the link address corresponds to the output port of the sensed data. If the IODF contains defined switch data, it is verified whether the defined data correspond to the sensed data. If dedicated switches are defined, the I/O Path report always shows differences in the D column.

- When you compare between the sensed and defined data, not all fields are used to determine if there is a difference between the two sides (either an *, or Θ is shown in the column D of the report). The data that is used are the CHPID, the...
control unit number, the device number, and the switch information of the
defined dynamic switches and the sensed dynamic switches.

The I/O path verification checks if the actual system contains the same paths as in
the defined I/O configuration. Differences are indicated in the I/O Path report in
column D with the following characters:

1. The * indicates that differences are found between the sensed and the
defined I/O path. Either
   • only sensed data is available, or
   • only defined data in the IODF is available, or
   • the sensed and defined switch data differ.

2. The C indicates that the defined and the sensed I/O path are the same, but
   the defined I/O path is defined to the CSS only.

3. An @ is a combination of * and C, and indicates that differences are found
   between the sensed and the defined I/O path, and that the I/O path is
   defined to the CSS only.

4. The 0 indicates that the defined I/O path in the IODF is defined to the
   operating system only.
I/O PATH REPORT

TIME: 14:06  DATE: 2005-01-13  PAGE V-1

SYSTEM:  SYSPLEX:  PROCESSOR: G29  CSS ID: 0  PARTITION: TRX1  OS CONFIG: XXY

<--------------------- Sensed data ----------------------> <-------------------------- IODF data ----------------------------->

00 OSA 0230 9676 0230,2 00 OSA 0230 OSA 0230,2 OSA
00 OSA 0230 9676 023F 9676-2 00 OSA 0230 OSA 023F OSAD
04 CFS FFF7 9672 9672-CFS 04 CFS FFF7 CFS FFE8,2 CFS
05 CFS FFF8 9672-FE 9672-CFS 05 CFS FFF8 CFS FFE8,2 CFS
10 CTC_S EE AC C3 4010 9672-R72 4010,4 10 CTC_S EE AC C3 4010 9672-R72 4010,4
10 CTC_S EE AC C3 4020 9672-R72 4020,4 10 CTC_S EE AC C3 4020 9672-R72 4020,4
10 CTC_S EE AC CB 4030 9672-R72 4030,4 10 CTC_S EE AC CB 4030 9672-R72 4030,4
11 CNC_S DD AE FE 0001 9032-3 11 CNC_S DD AE FE 0001 9032-3
11 CNC_S DD AE DB 2000 9394-3 11 CNC_S DD AE DB 2000 9394-3
12 CNC 13 CNC 13 CNC
13 CNC OFFL 12 CNC OFFL 13 CNC
14 CNC_S ED AA DB 0830 3490-A20 0830,16 14 CNC_S ED AA DB 0830 3490-A20 0830,16
14 CNC_S ED AA EA 0850 3490-C2A 0850,2 14 CNC_S ED AA EA 0850 3490-C2A 0850,2
15 CNC_S 10 01 FA 2540 2105-E20 2540,37 2105 15 CNC_S 10 01 FA 2540 2105-E20 2540,37
15 CNC_S 10 01 FB 3440 2105-F20 3440,32 2105 15 CNC_S 10 01 FB 3440 2105-F20 3440,32
15 CNC_S 10 01 FA 3E80 2105-E20 3E80,13 2105 15 CNC_S 10 01 FA 3E80 2105-E20 3E80,13
18 CNC_S ES AA FE 000A 9032-0 000A 9032-0 18 CNC_S ES AA FE 000A 9032-0 000A 9032-0
19 CNC_S ES AA EA 0640 3490-3 0640,8 3490 19 CNC_S ES AA EA 0640 3490-3 0640,8 3490
1C CNC_S EF00 EF00,2 EF00,2 1C CNC_S EF00 EF00,2 EF00,2
20 OSA 0210 9676 0214,11 20 OSA 0210 9676 0214,11
20 OSA 0210 9676 9676-2 20 OSA 0210 9676 9676-2
2C CNC_S ED AC DB 5100 5100,4 2C CNC_S ED AC DB 5100 5100,4

Figure 172. Example and Legend of an I/O Path Report (Part 1 of 2)
<table>
<thead>
<tr>
<th>KEY</th>
<th>KEY DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SENSED DATA</td>
<td>I/O configuration data retrieved by sensing I/O</td>
</tr>
<tr>
<td>CHP ID</td>
<td>Sensed channel path ID</td>
</tr>
<tr>
<td>CHP TYPE</td>
<td>Sensed channel path type</td>
</tr>
<tr>
<td>DYN.SWITCH PI</td>
<td>Sensed dyn. switch input port (channel path side)</td>
</tr>
<tr>
<td>DYN.SWITCH SW</td>
<td>Sensed dynamic switch ID</td>
</tr>
<tr>
<td>DYN.SWITCH PO</td>
<td>Sensed dyn. switch output port (control unit side)</td>
</tr>
<tr>
<td>CONTROL UNIT NUMB -</td>
<td>Sensed control unit number</td>
</tr>
<tr>
<td>CONTROL UNIT TYPE -</td>
<td>Sensed control unit type</td>
</tr>
<tr>
<td>DEVICE NUMBER -</td>
<td>Sensed device number and range</td>
</tr>
<tr>
<td>DEVICE TYPE -</td>
<td>Sensed device type</td>
</tr>
<tr>
<td>DEVICE O -</td>
<td>Sensed device(s) offline indicator</td>
</tr>
<tr>
<td>Y</td>
<td>Offline</td>
</tr>
<tr>
<td>BLANK</td>
<td>Not offline</td>
</tr>
<tr>
<td>PATH STAT</td>
<td>If blank, sensed I/O path is online</td>
</tr>
<tr>
<td>OFFL</td>
<td>Sensed I/O path is offline</td>
</tr>
<tr>
<td>UNKN</td>
<td>Status of sensed I/O path is unknown</td>
</tr>
<tr>
<td>IODF DATA</td>
<td>I/O configuration definitions in IODF</td>
</tr>
<tr>
<td>CHP ID</td>
<td>Defined channel path ID</td>
</tr>
<tr>
<td>CHP TYPE</td>
<td>Defined channel path type</td>
</tr>
<tr>
<td>SWITCH PI</td>
<td>Defined switch input port (channel path side)</td>
</tr>
<tr>
<td>SWITCH SW</td>
<td>Defined switch ID</td>
</tr>
<tr>
<td>SWITCH PO</td>
<td>Defined switch output port (control unit side)</td>
</tr>
<tr>
<td>CHAINED PI</td>
<td>Defined chained switch input port</td>
</tr>
<tr>
<td>CHAINED SW</td>
<td>Defined chained switch ID</td>
</tr>
<tr>
<td>CHAINED PO</td>
<td>Defined chained switch output port</td>
</tr>
<tr>
<td>DYNSW SW</td>
<td>Defined dynamic switch for channel path</td>
</tr>
<tr>
<td>DYNSW LA</td>
<td>Defined dynamic link address for control unit</td>
</tr>
<tr>
<td>CONTROL UNIT NUMB</td>
<td>Defined control unit number</td>
</tr>
<tr>
<td>CONTROL UNIT TYPE</td>
<td>Defined control unit type</td>
</tr>
<tr>
<td>CONTROL UNIT L</td>
<td>Defined control unit logical address</td>
</tr>
<tr>
<td>DEVICE NUMBER</td>
<td>Defined device number and range</td>
</tr>
<tr>
<td>DEVICE TYPE</td>
<td>Defined device type</td>
</tr>
<tr>
<td>DEVICE S</td>
<td>Defined device subchannel set ID</td>
</tr>
<tr>
<td>D</td>
<td>If C, only defined to channel subsystem</td>
</tr>
<tr>
<td>O</td>
<td>Only defined to OS configuration</td>
</tr>
<tr>
<td>*</td>
<td>Differences were found between the 2 sides</td>
</tr>
<tr>
<td>@</td>
<td>Indicates C and * from above</td>
</tr>
</tbody>
</table>

**Figure 172. Example and Legend of an I/O Path Report (Part 2 of 2)**

**DYN.SWITCH**
Contains either the switch information for the dynamic switch in the path or the dedicated switch if there is no dynamic switch.

**PATH STAT**
Represents the status of the I/O path between the CHPID and the devices that it is connected to.

**OFFL**
Represents the situations where there is no I/O path to be found or when the I/O path is offline.

**UNKN**
Represents the situations where the I/O path is currently in a pending state or when the MVS-type environment is running as a guest on a VM system and the path status cannot be obtained from the system.
**Supported Hardware Report**

The Supported Hardware Report shows the actual status of the hardware supported in your installation. It shows the following supported hardware:

- Processors
- Control units
- Devices including:
  - Device characteristics
  - Control unit attachments
- MVS devices including:
  - Device capabilities
  - Parameters (with selection values) / features
- VM devices including
  - device capabilities
  - parameters/features

The following figure shows you an example of a supported hardware report. The example may differ from the report you get on your system, because the data depend on the installed processor support modules and UIMs.
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>MOD</th>
<th>SUPLEVID</th>
<th>IOCP</th>
<th>SUPPORTED HARDWARE - PROCESSES</th>
<th>IOCPP</th>
<th>SUPPORTED CHPID TYPES</th>
<th>DCM</th>
<th>CAS</th>
<th>W1</th>
<th>RI</th>
<th>DP</th>
<th>CHPID</th>
<th>CU</th>
<th>LCU</th>
<th>SUBCH</th>
<th>LPAR</th>
<th>CSS</th>
<th>SCHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2003-1C5</td>
<td>177</td>
<td>H970513</td>
<td>IZP</td>
<td>2003,1</td>
<td></td>
<td>BL, BY, CTC, CTC, CVC, CBY, OSA</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>128</td>
<td>6144</td>
<td>3072</td>
<td>24576</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2003-1C6</td>
<td>167</td>
<td>H960703</td>
<td>IZP</td>
<td>2003,1</td>
<td></td>
<td>BL, BY, CTC, CTC, CVC, CBY, OSA</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>128</td>
<td>6144</td>
<td>3072</td>
<td>24576</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2003-1C7</td>
<td>177</td>
<td>H970513</td>
<td>IZP</td>
<td>2003,1</td>
<td></td>
<td>BL, BY, CTC, CTC, CVC, CBY, OSA</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>128</td>
<td>6144</td>
<td>3072</td>
<td>24576</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2003-1C8</td>
<td>177</td>
<td>H970513</td>
<td>IZP</td>
<td>2003,1</td>
<td></td>
<td>BL, BY, CTC, CTC, CVC, CBY, OSA</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>128</td>
<td>6144</td>
<td>3072</td>
<td>24576</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2003-1C9</td>
<td>177</td>
<td>H970513</td>
<td>IZP</td>
<td>2003,1</td>
<td></td>
<td>BL, BY, CTC, CTC, CVC, CBY, OSA</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>128</td>
<td>6144</td>
<td>3072</td>
<td>24576</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2003-1C0</td>
<td>177</td>
<td>H970513</td>
<td>IZP</td>
<td>2003,1</td>
<td></td>
<td>BL, BY, CTC, CTC, CVC, CBY, OSA</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>128</td>
<td>6144</td>
<td>3072</td>
<td>24576</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 1 of 29)
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2064-1C5</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>ICP, IOD, FCP</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-1C6</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-1C6</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-1C7</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-1C7</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-1C8</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-1C8</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-1C9</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-1C9</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-100</td>
<td>201</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,2</td>
<td>N N Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-101</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>N N Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-101</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>N N Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-102</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-102</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-103</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-103</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-104</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-104</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-105</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-105</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-106</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>Y Y Y N</td>
<td>256</td>
<td>8192 4096 64512</td>
<td>15</td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 2 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>MOD</th>
<th>SUPLEVEL</th>
<th>IOCP</th>
<th>SYSTEM</th>
<th>SUPPORTED HARDWARE - PROCESSORS</th>
<th>SUPPORTED CHPID TYPES</th>
<th>TIME:</th>
<th>DATE:</th>
<th>PAGE X-</th>
</tr>
</thead>
<tbody>
<tr>
<td>2064-108</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td>2008-02-15</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>2064-108</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-109</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-109</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-110</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-110</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-111</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-111</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-112</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-112</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-113</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-113</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-114</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-114</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-115</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-115</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-116</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-116</td>
<td>202</td>
<td>H010931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-2C1</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-2C1</td>
<td>202</td>
<td>H020415</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-2C2</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-2C2</td>
<td>202</td>
<td>H020415</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-2C3</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y N Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2064-2C3</td>
<td>202</td>
<td>H020415</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFS,CBY,OSA, FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP, ICP, ICP, ICP</td>
<td>Y Y Y Y N 2 5 6 8192 4096</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 3 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>MOD</th>
<th>SUPLEV ID</th>
<th>ICP</th>
<th>SYSTEM</th>
<th>SUPPORTED HARDWARE - PROCESSORS</th>
<th>TIME: 12:45</th>
<th>DATE: 2008-02-15</th>
<th>PAGE X- 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>2064-2C3</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C3</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C4</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y N Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C4</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C5</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C5</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C6</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y N Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C6</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C7</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y N Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C7</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C8</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y N Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C8</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C9</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y N Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-2C9</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-210</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y N Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-210</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-211</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y N Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-211</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-212</td>
<td>200</td>
<td>HO00931</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y N Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
<tr>
<td>2064-212</td>
<td>202</td>
<td>HO02015</td>
<td>IYP</td>
<td>2064,1</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,CBY,OSA,FC,FCV,CBR,CBS,OSD,OSE,CFP,CBP,ICP,100,FCP</td>
<td>Y Y Y Y N</td>
<td>256 1892 4096 64512</td>
<td>15</td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 4 of 29)
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2064-214</td>
<td>202</td>
<td>H020415</td>
<td>IYP</td>
<td>2064</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2064-215</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2064-215</td>
<td>202</td>
<td>H020415</td>
<td>IYP</td>
<td>2064</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y Y Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2064-216</td>
<td>200</td>
<td>H000931</td>
<td>IYP</td>
<td>2064</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y Y Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0A1</td>
<td>206</td>
<td>H020331</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0A1</td>
<td>208</td>
<td>H020731</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0A2</td>
<td>206</td>
<td>H020331</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0B1</td>
<td>206</td>
<td>H020331</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0B1</td>
<td>208</td>
<td>H020731</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0C1</td>
<td>206</td>
<td>H020331</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0C1</td>
<td>208</td>
<td>H020731</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0D1</td>
<td>206</td>
<td>H020331</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0D1</td>
<td>208</td>
<td>H020731</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0EC</td>
<td>207</td>
<td>H020331</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0EC</td>
<td>206</td>
<td>H020331</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0EC</td>
<td>208</td>
<td>H020731</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0EC</td>
<td>206</td>
<td>H020331</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2066-0EC</td>
<td>208</td>
<td>H020731</td>
<td>IYP</td>
<td>2066</td>
<td>BL, BY, CTC, CVC, CFR, CBS, OSA, FC, FCV, CB, CBS, OSO, DSE, OSE, CPF, CBP, ICP, IOP, FCP</td>
<td>Y N Y Y N</td>
<td>256 8192 4096 64512</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 5 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>MOD</th>
<th>SPMLEV</th>
<th>IOCP</th>
<th>SUPPORTED HARDWARE - PROCESSORS</th>
<th>TIME: 12:45</th>
<th>DATE: 2008-02-15</th>
<th>PAGE X-6</th>
</tr>
</thead>
<tbody>
<tr>
<td>2084-B16</td>
<td>210</td>
<td>H030530</td>
<td>ICP</td>
<td>2084,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP</td>
<td>Y Y Y Y Y N</td>
<td>512</td>
<td>8192 4096 644512 30 2</td>
</tr>
<tr>
<td>2084-B16</td>
<td>211</td>
<td>H040331</td>
<td>ICP</td>
<td>2084,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 644512 30 4</td>
<td></td>
</tr>
<tr>
<td>2084-B24</td>
<td>210</td>
<td>H030530</td>
<td>ICP</td>
<td>2084,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC</td>
<td>Y Y Y Y Y N</td>
<td>512</td>
<td>8192 4096 644512 30 2</td>
</tr>
<tr>
<td>2084-B24</td>
<td>211</td>
<td>H040331</td>
<td>ICP</td>
<td>2084,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 644512 30 4</td>
<td></td>
</tr>
<tr>
<td>2084-D32</td>
<td>210</td>
<td>H030530</td>
<td>ICP</td>
<td>2084,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC</td>
<td>Y Y Y Y Y N</td>
<td>512</td>
<td>8192 4096 644512 30 2</td>
</tr>
<tr>
<td>2084-D32</td>
<td>211</td>
<td>H040331</td>
<td>ICP</td>
<td>2084,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 644512 30 4</td>
<td></td>
</tr>
<tr>
<td>2086-A04</td>
<td>212</td>
<td>H040331</td>
<td>ICP</td>
<td>2086,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC</td>
<td>Y Y Y Y Y N</td>
<td>512</td>
<td>8192 4096 644512 30 2</td>
</tr>
<tr>
<td>2094-S08</td>
<td>213</td>
<td>H050331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 40 2</td>
<td></td>
</tr>
<tr>
<td>2094-S08</td>
<td>216</td>
<td>H070331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S18</td>
<td>213</td>
<td>H050331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S18</td>
<td>216</td>
<td>H070331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S28</td>
<td>213</td>
<td>H050331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S28</td>
<td>216</td>
<td>H070331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S38</td>
<td>213</td>
<td>H050331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S38</td>
<td>216</td>
<td>H070331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S64</td>
<td>213</td>
<td>H050331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S64</td>
<td>216</td>
<td>H070331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S64</td>
<td>219</td>
<td>H070331</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S64</td>
<td>220</td>
<td>H080130</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S64</td>
<td>221</td>
<td>H080131</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S64</td>
<td>222</td>
<td>H080130</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S64</td>
<td>223</td>
<td>H080131</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
<tr>
<td>2094-S64</td>
<td>224</td>
<td>H080130</td>
<td>ICP</td>
<td>2094,1 CNC, CTC, CVC, CBY, FC, FCV, CSR, CSR, OSE, OCF, CBF, ICP, IQD, FCP, OSC, OSN</td>
<td>Y Y Y Y Y 1024</td>
<td>8192 4096 65280 60 4 2</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 173. Supported Hardware Report (Part 6 of 29)**
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>MOD</th>
<th>SUPLEVEL</th>
<th>IOCPE</th>
<th>SYSTEM</th>
<th>SUPPORTED HARDWARE - PROCESSORS</th>
<th>SUPPORTED CHPID TYPES</th>
<th>TIME:</th>
<th>DATE:</th>
<th>PAGE X-</th>
</tr>
</thead>
<tbody>
<tr>
<td>7060-H30</td>
<td>189</td>
<td>H990930</td>
<td>IZP</td>
<td>7060,1</td>
<td>BL, BY, CTC, CVC, CBY, DSD, ICR, ICS, EID</td>
<td>N N N N 256 4096 36864 15</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>7</td>
</tr>
<tr>
<td>7060-H50</td>
<td>189</td>
<td>H990930</td>
<td>IZP</td>
<td>7060,1</td>
<td>BL, BY, CTC, CVC, CBY, DSD, ICR, ICS, EID</td>
<td>N N N N 256 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7060-H55</td>
<td>187</td>
<td>H010131</td>
<td>IZP</td>
<td>7060,2</td>
<td>BL, BY, CTC, CVC, CBY, DSD, ICR, ICS, OSE</td>
<td>N N N N 256 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7060-H70</td>
<td>189</td>
<td>H990930</td>
<td>IZP</td>
<td>7060,1</td>
<td>BL, BY, CTC, CVC, CBY, DSD, ICR, ICS, EID</td>
<td>N N N N 256 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7060-H75</td>
<td>187</td>
<td>H010131</td>
<td>IZP</td>
<td>7060,2</td>
<td>BL, BY, CTC, CVC, CBY, DSD, ICR, ICS, OSE</td>
<td>N N N N 256 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 7 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>MOD</th>
<th>SUPLEVID</th>
<th>IOC</th>
<th>SYSTEM</th>
<th>SUPPORTED HARDWARE - PROCESSORS</th>
<th>SUPPORTED CHIPID TYPES</th>
<th>TIME:</th>
<th>DATE:</th>
<th>PAGE:</th>
<th>X:</th>
<th>Y:</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>9672-R25</td>
<td>169</td>
<td>H970610</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 6444 3072 25476 15</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R26</td>
<td>180</td>
<td>H980730</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 8192 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R26</td>
<td>184</td>
<td>H991130</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 8192 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R31</td>
<td>160</td>
<td>H950303</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 56 4096 2048 16384 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R31</td>
<td>163</td>
<td>H940623</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 56 4096 2048 16384 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R32</td>
<td>161</td>
<td>H950425</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 128 6444 3072 25476 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R33</td>
<td>166</td>
<td>H960703</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 6444 3072 25476 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R34</td>
<td>169</td>
<td>H970610</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 8192 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R36</td>
<td>180</td>
<td>H980730</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 8192 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R36</td>
<td>184</td>
<td>H991130</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 8192 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R41</td>
<td>160</td>
<td>H950303</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 56 4096 2048 16384 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R41</td>
<td>163</td>
<td>H940623</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 56 4096 2048 16384 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R42</td>
<td>161</td>
<td>H950425</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 128 6444 3072 25476 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R44</td>
<td>166</td>
<td>H960703</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 6444 3072 25476 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R45</td>
<td>169</td>
<td>H970610</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 6444 3072 25476 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-R46</td>
<td>180</td>
<td>H980730</td>
<td>I2P</td>
<td>9672</td>
<td>BL, BY, CTC, CVC, CTR, CBA, OSA</td>
<td>N Y Y N 256 8192 4096 36864 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 8 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>MOD</th>
<th>SUPLEVEL</th>
<th>IOCPSYSTEM</th>
<th>SUPPORTED HARDWARE - PROCESSORS</th>
<th>DCM</th>
<th>CAS</th>
<th>W1</th>
<th>RI</th>
<th>DP</th>
<th>CHPID</th>
<th>CU</th>
<th>LCU</th>
<th>SUBCH</th>
<th>LPAR</th>
<th>CSS</th>
<th>SCHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>9672-X7</td>
<td>182</td>
<td>H990531</td>
<td>IZP 9672,6</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,OSA, FCV,CFB,CBS,ICR,ICS,OSD,OSE</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>256</td>
<td>8192</td>
<td>4096</td>
<td>36864</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-X7</td>
<td>186</td>
<td>H991130</td>
<td>IZP 9672,6</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,OSA, FCV,CFB,CBS,ICR,ICS,OSD,OSE</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>256</td>
<td>8192</td>
<td>4096</td>
<td>36864</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-X7</td>
<td>182</td>
<td>H990531</td>
<td>IZP 9672,6</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,OSA, FCV,CFB,CBS,ICR,ICS,OSD,OSE</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>256</td>
<td>8192</td>
<td>4096</td>
<td>36864</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-X7</td>
<td>186</td>
<td>H991130</td>
<td>IZP 9672,6</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,OSA, FCV,CFB,CBS,ICR,ICS,OSD,OSE</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>256</td>
<td>8192</td>
<td>4096</td>
<td>36864</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-X7</td>
<td>182</td>
<td>H990531</td>
<td>IZP 9672,6</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,OSA, FCV,CFB,CBS,ICR,ICS,OSD,OSE</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>256</td>
<td>8192</td>
<td>4096</td>
<td>36864</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9672-X7</td>
<td>186</td>
<td>H991130</td>
<td>IZP 9672,6</td>
<td>BL,BY,CNC,CTC,CVC,CFR,CFB,OSA, FCV,CFB,CBS,ICR,ICS,OSD,OSE</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>256</td>
<td>8192</td>
<td>4096</td>
<td>36864</td>
<td>15</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 9 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>MOD</th>
<th>SUBLVLID</th>
<th>IOC</th>
<th>SUPPORTED HARDWARE - PROCESSORS</th>
<th>SUPPORTED CHPID TYPES</th>
<th>TIME:</th>
<th>DATE:</th>
<th>PAGE:</th>
</tr>
</thead>
<tbody>
<tr>
<td>9672-Y66</td>
<td>184</td>
<td>H991130</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-Y76</td>
<td>180</td>
<td>H980730</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-Y86</td>
<td>184</td>
<td>H991130</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-Y96</td>
<td>180</td>
<td>H980730</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-ZX7</td>
<td>182</td>
<td>H990531</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-ZY7</td>
<td>182</td>
<td>H990531</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-ZZ7</td>
<td>182</td>
<td>H990531</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-Z17</td>
<td>182</td>
<td>H990531</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-Z27</td>
<td>182</td>
<td>H990531</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-Z37</td>
<td>182</td>
<td>H990531</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-Z47</td>
<td>182</td>
<td>H990531</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9672-Z57</td>
<td>182</td>
<td>H990531</td>
<td>I2P</td>
<td>9672,6</td>
<td>BL, BY, CTC, CVC, CFR, CBS, CYB, OSA</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
<tr>
<td>9674-C01</td>
<td>171</td>
<td>H940318</td>
<td>I2P</td>
<td>9674,1</td>
<td>FC, CFR</td>
<td>12:45</td>
<td>2008-02-15</td>
<td>10</td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 10 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>MOD</th>
<th>SUPLEVEL</th>
<th>IOCP</th>
<th>SYSTEM</th>
<th>SUPPORTED CHPID TYPES</th>
<th>DOM</th>
<th>CAS</th>
<th>WI</th>
<th>RI</th>
<th>DP</th>
<th>CHPID</th>
<th>CU</th>
<th>LCU</th>
<th>SUBCH</th>
<th>LPAR</th>
<th>CSS</th>
<th>SCHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>9674-C03</td>
<td>173</td>
<td>H950427</td>
<td>IZP</td>
<td>9674,3</td>
<td>CFR</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>32</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9674-C04</td>
<td>174</td>
<td>H960703</td>
<td>IZP</td>
<td>9674,4</td>
<td>CFR</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>32</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9674-C05</td>
<td>175</td>
<td>H970610</td>
<td>IZP</td>
<td>9674,4</td>
<td>CFR</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>32</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**KEY**

---

**DESCRIPTION**

- **TYPE-MODEL**: SUPPORTED PROCESSOR TYPE
- **MOD**: INDEX OF PROCESSOR SUPPORT MODULE
- **SUPLEVEL**: SUPPORT LEVEL ID
- **IOCP**: PREFIX OF IOC PROGRAM FOR THIS SUPPORT LEVEL
- **SYSTEM**: IOC SYSTEM VALUE
- **SUPPORTED CHPID TYPES**: LIST OF CHANNEL PATH TYPES WHICH ARE SUPPORTED BY THIS CONTROL UNIT
- **DOM**: DYNAMIC CHPID MANAGEMENT SUPPORT
- **CAS**: SUPPORT OF FICON 2-SWITCH CASCADED SWITCHING
- **WI**: WRITE IOCDS REGARDLESS OF CPC TYPE
- **RI**: RECEIVE IOCDS WRITTEN REGARDLESS OF CPC TYPE
- **DP**: SUPPORT OF DYNAMIC PARTITIONS
- **CHPID**: MAXIMUM NUMBER OF SUPPORTED CHPIDS
- **CU**: MAXIMUM NUMBER OF ATTACHABLE CONTROL UNITS
- **LCU**: MAXIMUM NUMBER OF LOGICAL CONTROL UNITS (PER CHANNEL SUBSYSTEM)
- **SUBCH**: MAXIMUM NUMBER OF SUBCHANNELS (PER CHANNEL SUBSYSTEM)
- **LPAR**: MAXIMUM NUMBER OF LOGICAL PARTITIONS
- **CSS**: MAXIMUM NUMBER OF CHANNEL SUBSYSTEMS (XMP PROCESSOR ONLY)
- **Y**: PROCESSOR HAS THE CAPABILITY
- **N**: CAPABILITY IS NOT AVAILABLE

*Figure 173. Supported Hardware Report (Part 11 of 29)*
Appendix B. Configuration reports

Figure 173. Supported Hardware Report (Part 12 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UIM</th>
<th>PROTCL</th>
<th>DP ID</th>
<th>SUPPORTED CHIPID TYPES</th>
<th>ATTACHABLE DEVICES</th>
</tr>
</thead>
<tbody>
<tr>
<td>3424</td>
<td>005 N S</td>
<td>2</td>
<td>BL,CVC,EIO</td>
<td>3424</td>
<td></td>
</tr>
<tr>
<td>3430</td>
<td>005 N D</td>
<td>1</td>
<td>BL,CVC,EIO</td>
<td>3430</td>
<td></td>
</tr>
<tr>
<td>3480</td>
<td>005 Y,D,S,54 S</td>
<td>2</td>
<td>BL,CNC,CVC,FCY,EIO</td>
<td>3480</td>
<td></td>
</tr>
<tr>
<td>3490</td>
<td>005 Y D,S,54 S</td>
<td>2</td>
<td>BL,CNC,CVC,ISO,FCY,EIO</td>
<td>3490</td>
<td></td>
</tr>
<tr>
<td>3490-C1A</td>
<td>005 Y D,S,54 S</td>
<td>2</td>
<td>BL,CNC,CVC,FCY,EIO</td>
<td>3490</td>
<td></td>
</tr>
<tr>
<td>3490-C1D</td>
<td>005 Y D,S,54 S</td>
<td>2</td>
<td>BL,CNC,CVC,ISO,FCY,EIO</td>
<td>3490</td>
<td></td>
</tr>
<tr>
<td>3490-C1I</td>
<td>005 Y D,S,54 S</td>
<td>2</td>
<td>BL,CNC,CVC,ISO,FCY,EIO</td>
<td>3490</td>
<td></td>
</tr>
<tr>
<td>3490-C2A</td>
<td>005 Y D,S,54 S</td>
<td>2</td>
<td>BL,CNC,CVC,FCY,EIO</td>
<td>3490</td>
<td></td>
</tr>
<tr>
<td>3490-C2D</td>
<td>005 Y D,S,54 S</td>
<td>2</td>
<td>BL,CNC,CVC,FCY,EIO</td>
<td>3490</td>
<td></td>
</tr>
<tr>
<td>3505</td>
<td>012 N D</td>
<td>2</td>
<td>BL,CVC,CBY,EIO</td>
<td>3505,3525</td>
<td></td>
</tr>
<tr>
<td>3540</td>
<td>032 N D</td>
<td>2</td>
<td>BL,CVC,CBY,EIO</td>
<td>3540</td>
<td></td>
</tr>
<tr>
<td>3590</td>
<td>005 Y D,S,54 S</td>
<td>2</td>
<td>BL,CNC,CVC,FCY,EIO</td>
<td>3590</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 173. Supported Hardware Report (Part 13 of 29)**
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UIM</th>
<th>U</th>
<th>PROTCL</th>
<th>DP</th>
<th>IO</th>
<th>SUPPORTED CHPID TYPES</th>
<th>ATTACHABLE DEVICES</th>
</tr>
</thead>
<tbody>
<tr>
<td>4245</td>
<td>012</td>
<td>N</td>
<td>D</td>
<td>D2</td>
<td>2</td>
<td>BL,BY,CVC,CBY,EIO</td>
<td>4245</td>
</tr>
<tr>
<td>4248</td>
<td>012</td>
<td>N</td>
<td>D</td>
<td>D2</td>
<td>2</td>
<td>BL,BY,CVC,CBY,EIO</td>
<td>3262,4248,6262</td>
</tr>
<tr>
<td>4370</td>
<td>022</td>
<td>N</td>
<td>D,S</td>
<td>S2</td>
<td>2</td>
<td>BL,BY,CNC,CVC,CBY,FCV,EIO</td>
<td>AFPI-0,4370</td>
</tr>
<tr>
<td>5088-1</td>
<td>291</td>
<td>N</td>
<td>D,S</td>
<td>D2</td>
<td>2</td>
<td>BL,CVC,EIO</td>
<td>HFGD,3251,3279-28,3279-38,5080,5081,6090,6091</td>
</tr>
<tr>
<td>5088-2</td>
<td>291</td>
<td>N</td>
<td>D,S</td>
<td>D2</td>
<td>2</td>
<td>BL,CVC,EIO</td>
<td>HFGD,3251,3279-28,3279-38,5080,5081,6090,6091</td>
</tr>
<tr>
<td>6098</td>
<td>291</td>
<td>N</td>
<td>S,S4</td>
<td>S2</td>
<td>2</td>
<td>BL,CVC,EIO</td>
<td>HFGD,3251,3279-28,3279-38,5080,5081,6090,6091</td>
</tr>
<tr>
<td>6120</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D2</td>
<td>2</td>
<td>IOC,EIO</td>
<td>3101,3151,3174,3179,3180,3190,3191,3192,3193,3194,3262,3268,3270,3277,3278,3279,3286,3287,3289,3290,3812,3816,4224,4245,4250,5210</td>
</tr>
<tr>
<td>6135</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6139</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAILAN</td>
</tr>
<tr>
<td>6140</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6212</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6243</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6244</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6245</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6251</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6252</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6253</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6254</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6255</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6256</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6257</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6258</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6262</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6310</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>6311</td>
<td>308</td>
<td>N</td>
<td>D</td>
<td>D1</td>
<td>1</td>
<td>IOC,EIO</td>
<td>ICAELAN</td>
</tr>
<tr>
<td>7171</td>
<td>027</td>
<td>N</td>
<td>D</td>
<td>D2</td>
<td>2</td>
<td>BL,BY,CVC,EIO</td>
<td>4245</td>
</tr>
<tr>
<td>7177-3</td>
<td>023</td>
<td>N</td>
<td>D</td>
<td>D2</td>
<td>2</td>
<td>BY,CVC</td>
<td>3101,3151,3174,3179,3180,3190,3191,3192,3193,3194,3262,3268,3270,3277,3278,3279,3286,3287,3289,3290,3812,3816,4224,4245,4250,5210</td>
</tr>
<tr>
<td>8232</td>
<td>014</td>
<td>N</td>
<td>D,S,S4</td>
<td>S2</td>
<td>2</td>
<td>BL,CVC,EIO</td>
<td>8232</td>
</tr>
<tr>
<td>9032</td>
<td>051</td>
<td>Y</td>
<td>2</td>
<td>CNC</td>
<td></td>
<td>SWCH,9032</td>
<td>9032</td>
</tr>
<tr>
<td>9032-3</td>
<td>051</td>
<td>Y</td>
<td>2</td>
<td>CNC</td>
<td></td>
<td>SWCH,9032-3</td>
<td>9032</td>
</tr>
<tr>
<td>9032-5</td>
<td>051</td>
<td>Y</td>
<td>2</td>
<td>CNC,FCV</td>
<td></td>
<td>SWCH,9032-5</td>
<td>9032</td>
</tr>
<tr>
<td>9033</td>
<td>051</td>
<td>Y</td>
<td>2</td>
<td>CNC</td>
<td></td>
<td>SWCH,9033</td>
<td>9033</td>
</tr>
<tr>
<td>9340</td>
<td>002</td>
<td>Y</td>
<td>S,S4</td>
<td>S2</td>
<td>2</td>
<td>BL,CVC,CVC,FCV,EIO</td>
<td>9345</td>
</tr>
<tr>
<td>9341</td>
<td>002</td>
<td>Y</td>
<td>S,S4</td>
<td>S2</td>
<td>2</td>
<td>BL,CVC,CVC,FCV,EIO</td>
<td>9345</td>
</tr>
<tr>
<td>9343</td>
<td>002</td>
<td>Y</td>
<td>S,S4</td>
<td>S2</td>
<td>2</td>
<td>BL,CVC,CVC,FCV,EIO</td>
<td>9345</td>
</tr>
<tr>
<td>9343-1</td>
<td>002</td>
<td>Y</td>
<td>S,S4</td>
<td>S2</td>
<td>2</td>
<td>BL,CVC,CVC,FCV,EIO</td>
<td>9345</td>
</tr>
</tbody>
</table>

**KEY**
- **TYPE-MODEL** - SUPPORTED CONTROL UNIT TYPE
- **UIM** - INDEX OF UNIT INFORMATION MODULE SUPPORTING THIS CONTROL UNIT TYPE
- **U** - If Y, indicates that the unit address range must start with 00 when the control unit is connected to an ESCON Channel Path
- **DP** - DEFAULT PROTOCOL FOR PARALLEL CONTROL UNIT
- **IO** - DEFAULT I/O CONCURRENCY LEVEL FOR PARALLEL CONTROL UNIT
- **PROTCL** - PROTOCOL SUPPORTED BY THE CONTROL UNIT TYPE
- **SUPPORTED CHPID TYPES** - LIST OF CHANNEL PATH TYPES WHICH ARE SUPPORTED BY THIS CONTROL UNIT TYPE
- **ATTACHABLE DEVICES** - LIST OF ATTACHABLE DEVICE TYPES

**Figure 173. Supported Hardware Report (Part 14 of 29)**
<table>
<thead>
<tr>
<th>SUPPORTED HARDWARE</th>
<th>CONTROL UNITS (2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MD</td>
<td>CM</td>
</tr>
<tr>
<td>AFP1</td>
<td>1</td>
</tr>
<tr>
<td>CTC</td>
<td>1</td>
</tr>
<tr>
<td>FCTC</td>
<td>1</td>
</tr>
<tr>
<td>IQD</td>
<td>1</td>
</tr>
<tr>
<td>NOCHECK</td>
<td>8</td>
</tr>
<tr>
<td>OSA</td>
<td>1</td>
</tr>
<tr>
<td>OSC</td>
<td>1</td>
</tr>
<tr>
<td>OSN</td>
<td>1</td>
</tr>
<tr>
<td>R56K</td>
<td>1</td>
</tr>
<tr>
<td>R56K-2</td>
<td>N</td>
</tr>
<tr>
<td>SMCH</td>
<td>1</td>
</tr>
<tr>
<td>SWCH</td>
<td>1</td>
</tr>
<tr>
<td>1750</td>
<td>256</td>
</tr>
<tr>
<td>2032</td>
<td>256</td>
</tr>
<tr>
<td>2105</td>
<td>256</td>
</tr>
<tr>
<td>2501</td>
<td>-</td>
</tr>
<tr>
<td>2701</td>
<td>-</td>
</tr>
<tr>
<td>2910</td>
<td>-</td>
</tr>
<tr>
<td>2821</td>
<td>-</td>
</tr>
<tr>
<td>2835-2</td>
<td>N</td>
</tr>
<tr>
<td>2940-2</td>
<td>N</td>
</tr>
<tr>
<td>3088</td>
<td>-</td>
</tr>
<tr>
<td>3160</td>
<td>-</td>
</tr>
<tr>
<td>3170</td>
<td>-</td>
</tr>
<tr>
<td>3172</td>
<td>2</td>
</tr>
<tr>
<td>3174</td>
<td>256</td>
</tr>
<tr>
<td>3203</td>
<td>-</td>
</tr>
<tr>
<td>3258</td>
<td>16</td>
</tr>
<tr>
<td>3262</td>
<td>1</td>
</tr>
<tr>
<td>3272</td>
<td>32</td>
</tr>
<tr>
<td>3274</td>
<td>32</td>
</tr>
<tr>
<td>3300</td>
<td>-</td>
</tr>
<tr>
<td>3380-C22</td>
<td>N</td>
</tr>
<tr>
<td>3422</td>
<td>16</td>
</tr>
<tr>
<td>3423</td>
<td>16</td>
</tr>
<tr>
<td>3424</td>
<td>8</td>
</tr>
<tr>
<td>3430</td>
<td>16</td>
</tr>
<tr>
<td>3480</td>
<td>16</td>
</tr>
<tr>
<td>3490</td>
<td>16</td>
</tr>
<tr>
<td>3490-C1A</td>
<td>N</td>
</tr>
<tr>
<td>3490-C1B</td>
<td>N</td>
</tr>
<tr>
<td>3490-C1C</td>
<td>N</td>
</tr>
<tr>
<td>3490-C2A</td>
<td>N</td>
</tr>
<tr>
<td>3490-C22</td>
<td>N</td>
</tr>
<tr>
<td>3505</td>
<td>-</td>
</tr>
<tr>
<td>3540</td>
<td>2</td>
</tr>
<tr>
<td>3590</td>
<td>16</td>
</tr>
<tr>
<td>3704</td>
<td>-</td>
</tr>
<tr>
<td>3705</td>
<td>-</td>
</tr>
<tr>
<td>3720</td>
<td>-</td>
</tr>
<tr>
<td>3725</td>
<td>-</td>
</tr>
<tr>
<td>3737</td>
<td>-</td>
</tr>
<tr>
<td>3745</td>
<td>-</td>
</tr>
<tr>
<td>3746</td>
<td>-</td>
</tr>
<tr>
<td>3790L</td>
<td>256</td>
</tr>
<tr>
<td>3800-1</td>
<td>Y</td>
</tr>
<tr>
<td>3800-3</td>
<td>N</td>
</tr>
<tr>
<td>3800-6</td>
<td>N</td>
</tr>
<tr>
<td>3800-8</td>
<td>N</td>
</tr>
<tr>
<td>3803</td>
<td>16</td>
</tr>
<tr>
<td>3811</td>
<td>1</td>
</tr>
<tr>
<td>3820</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 15 of 29)
<table>
<thead>
<tr>
<th>TYPE</th>
<th>MODEL</th>
<th>MD</th>
<th>CHFMX</th>
<th>DEVMIN</th>
<th>DEVMAX</th>
<th>UNIT ADDRESSES</th>
<th>TIME: 12:45</th>
<th>DATE: 2008-02-15</th>
<th>PAGE X- 16</th>
</tr>
</thead>
</table>

**Figure 173. Supported Hardware Report (Part 16 of 29)**
<table>
<thead>
<tr>
<th>KEY</th>
<th>KEY DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE-MODEL</td>
<td>SUPPORTED CONTROL UNIT TYPE</td>
</tr>
<tr>
<td>MD</td>
<td>IF Y, MODEL IS DEFAULT</td>
</tr>
<tr>
<td>ATTACHMENT CNT CHPMAX</td>
<td>MAXIMUM NUMBER OF CHANNEL PATHS THAT CAN BE CONNECTED TO C/U</td>
</tr>
<tr>
<td>DEVMIN</td>
<td>MINIMUM NUMBER OF DEVICES THAT MUST BE CONNECTED TO C/U</td>
</tr>
<tr>
<td>DEVMAX</td>
<td>MAXIMUM NUMBER OF DEVICES THAT CAN BE CONNECTED TO C/U</td>
</tr>
<tr>
<td>UNIT ADDRESSES MIN</td>
<td>MINIMUM NUMBER OF UNIT ADDRESSES THAT MUST BE DEFINED TO C/U</td>
</tr>
<tr>
<td>UNIT ADDRESSES MAX</td>
<td>MAXIMUM NUMBER OF UNIT ADDRESSES THAT CAN BE DEFINED TO C/U</td>
</tr>
<tr>
<td>UNIT ADDRESSES REC</td>
<td>RECOMMENDED NUMBER OF UNIT ADDRESSES</td>
</tr>
<tr>
<td>UNIT ADDRESSES RANGE</td>
<td>MAXIMUM NUMBER OF UNIT ADDRESS RANGES THAT CAN BE DEFINED TO C/U</td>
</tr>
<tr>
<td>LOGICAL ADDRESSING LA</td>
<td>IF Y, C/U SUPPORTS LOGICAL ADDRESSING (CUADD)</td>
</tr>
<tr>
<td>MIN</td>
<td>MINIMUM VALUE OF ALLOWED LOGICAL ADDRESS (CUADD)</td>
</tr>
<tr>
<td>MAX</td>
<td>MAXIMUM VALUE OF ALLOWED LOGICAL ADDRESS (CUADD)</td>
</tr>
<tr>
<td>MULTHOST</td>
<td>IF Y, MULTIPLE HOSTS CAN CONNECT TO THE SAME LOGICAL ADDRESS (CUADD)</td>
</tr>
<tr>
<td>MAXCU</td>
<td>MAXIMUM NUMBER OF LOGICAL CONTROL UNITS SUPPORTED (CUADD)</td>
</tr>
<tr>
<td>LOGICAL PATHS MAPATH</td>
<td>MAXIMUM NUMBER OF LOGICAL PATHS SUPPORTED BY C/U</td>
</tr>
<tr>
<td>MINGRP</td>
<td>MINIMUM GROUP ATTACHMENT VALUE FOR LOGICAL PATHS</td>
</tr>
<tr>
<td>SH</td>
<td>IF Y, ONLY SINGLE HOST CAN ATTACH TO CONTROL UNIT AT A TIME</td>
</tr>
<tr>
<td>MXESC</td>
<td>MAXIMUM NUMBER OF LOGICAL PATHS PER ESCON PORT</td>
</tr>
<tr>
<td>MXFIC</td>
<td>MAXIMUM NUMBER OF LOGICAL PATHS PER FICON PORT</td>
</tr>
<tr>
<td>DCM</td>
<td>DYNAMIC CHIPID MANAGEMENT SUPPORT</td>
</tr>
<tr>
<td>Y</td>
<td>CONTROL UNIT HAS THE CAPABILITY</td>
</tr>
<tr>
<td>N</td>
<td>CONTROL UNIT DOES NOT HAVE THE CAPABILITY</td>
</tr>
<tr>
<td>-</td>
<td>UIM DOES NOT DEFINE A VALUE OR VALUE IS NOT APPLICABLE</td>
</tr>
</tbody>
</table>

*Figure 173. Supported Hardware Report (Part 17 of 29)*
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UIM</th>
<th>VM</th>
<th>MIX</th>
<th>GR</th>
<th>RL</th>
<th>RH</th>
<th>RD</th>
<th>ST</th>
<th>ATTACHABLE TO CU</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFPI-0</td>
<td>022</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>BCTC</td>
<td>014</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>BSCI</td>
<td>026</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>BSCI</td>
<td>026</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>BTCB</td>
<td>026</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>CTC</td>
<td>014</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>CTC</td>
<td>270</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>DUMMY</td>
<td>050</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>FBKSCSI</td>
<td>254</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>S6K</td>
<td>258</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>S6K</td>
<td>014</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>SMCH</td>
<td>291</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>SMCH</td>
<td>308</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>SMCH</td>
<td>308</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>SMCH</td>
<td>1030</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>SMCH</td>
<td>1050</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>SMCH</td>
<td>1050</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>SMCH</td>
<td>1150</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>SMCH</td>
<td>1227</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>1228</td>
<td>032</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>1403-1</td>
<td>012</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>1403-2</td>
<td>012</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>1403-3</td>
<td>268</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>1403-5</td>
<td>268</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>1403-7</td>
<td>012</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>203Z</td>
<td>014</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2250-Z</td>
<td>227</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2305-Z</td>
<td>229</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2501-81</td>
<td>012</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2501-82</td>
<td>012</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2540-1</td>
<td>012</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2540-1</td>
<td>012</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2701</td>
<td>022</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2710</td>
<td>278</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2740</td>
<td>024</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2740C</td>
<td>024</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2740X</td>
<td>024</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2741C</td>
<td>025</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2741P</td>
<td>025</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>308B</td>
<td>014</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3151</td>
<td>281</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3160</td>
<td>022</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3161</td>
<td>281</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3162</td>
<td>281</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3163</td>
<td>281</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3164</td>
<td>281</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3170</td>
<td>022</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3172</td>
<td>057</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3174</td>
<td>027</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3178</td>
<td>014</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3179</td>
<td>004</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3180</td>
<td>260</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3180-1</td>
<td>004</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3190</td>
<td>260</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3191</td>
<td>004</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3192</td>
<td>004</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 18 of 29)
Figure 173. Supported Hardware Report (Part 19 of 29)
TYPE-MODEL
_____________
3390S
3420-3
3420-4
3420-5
3420-6
3420-7
3420-8
3422
3423
3424
3430
3471
3472
3480
3481
3482
3483
3490
3505
3525
3540
3590
3704
3705
3720
3725
3737
3745
3746
3767-1
3767-2
3791L
3800-1
3800-3
3800-6
3800-8
3812
3816
3820
3825
3827
3828
3829
3831
3835
3851
3886
3890
3895
3900
3935
3995
3995-SDA
3995-151
3995-153
4000
4224
4245
4248
4250
4370
5080
5081
5210
6090
6091
6262
7171
7770-3
8232
83B3

UIM MVS VM MX
___ ___ __ __
002 Y
N Y
005 Y
Y N
005 Y
Y N
005 Y
Y N
005 Y
Y N
005 Y
Y N
005 Y
Y N
005 Y
Y N
005 Y
N N
005 Y
Y N
005 Y
Y N
004 Y
N N
004 Y
N N
005 Y
Y N
004 Y
N N
004 Y
N N
004 Y
N N
005 Y
Y N
012 Y
Y N
012 Y
Y N
032 Y
Y N
005 Y
Y N
023 Y
Y N
023 Y
Y N
023 Y
Y N
023 Y
Y N
014 Y
Y N
023 Y
Y N
023 Y
Y N
024 Y
N N
024 Y
N N
027 Y
N N
011 Y
Y N
011 Y
Y N
011 Y
Y N
011 Y
Y N
031 Y
Y N
287 N
Y N
022 Y
Y N
022 Y
Y N
022 Y
Y N
022 Y
Y N
022 Y
Y N
022 Y
Y N
022 Y
Y N
271 N
Y N
032 Y
Y N
032 Y
Y N
032 Y
Y N
022 Y
Y N
022 Y
Y N
053 Y
Y N
053 Y
Y N
002 Y
Y N
002 Y
Y N
022 Y
Y N
031 Y
Y N
012 Y
Y N
012 Y
Y N
031 Y
N N
022 Y
Y N
291 N
Y N
291 N
Y N
031 Y
Y N
291 N
Y N
291 N
Y N
268 N
Y N
027 Y
Y N
023 Y
N N
014 Y
Y N
025 Y
N N

GR
__
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
Y
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N
N

RL
__
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1

RH
____
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
1
256
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095
4095

SUPPORTED HARDWARE - DEVICES
TIME: 12:45 DATE: 2008-02-15 PAGE X- 20
RD TM ST ATTACHABLE TO CU
__ __ __ ___________________________________________________________________________________
1 Y Y NOCHECK,1750,2105,2107
1 Y Y NOCHECK,3803
1 Y Y NOCHECK,3803
1 Y Y NOCHECK,3803
1 Y Y NOCHECK,3803
1 Y Y NOCHECK,3803
1 Y Y NOCHECK,3803
1 Y Y NOCHECK,3422
1 Y Y NOCHECK,3423
1 Y Y NOCHECK,3424
1 Y Y NOCHECK,3430
1 Y Y NOCHECK,3174,3274
1 Y Y NOCHECK,3174,3274
1 Y Y NOCHECK,3480
1 Y Y NOCHECK,3174,3274
1 Y Y NOCHECK,3174,3274
1 Y Y NOCHECK,3174,3274
1 Y Y NOCHECK,3490,3490-C1A,3490-C10,3490-C11,3490-C2A,3490-C22
1 Y Y NOCHECK,3505
1 Y Y NOCHECK,3505
1 Y Y NOCHECK,3540
1 Y Y NOCHECK,3590
1 Y N NOCHECK,3704
1 Y N NOCHECK,3705
1 Y N NOCHECK,3720
1 Y N NOCHECK,3725
1 N Y NOCHECK,3737
1 Y N NOCHECK,OSN,3745,3746
1 Y N NOCHECK
1 Y Y NOCHECK,2701,3704,3705,3720,3725,3745,3746
1 Y Y NOCHECK,2701,3704,3705,3720,3725,3745,3746
1 Y N NOCHECK,3174,3272,3274,3704,3705,3720,3725,3745,3746,3791L,7171
1 Y Y NOCHECK,3800-1
1 Y Y NOCHECK,3800-3
1 Y Y NOCHECK,3800-6
1 Y Y NOCHECK,3800-8
1 Y Y NOCHECK,3174,3272,3274,3791L,6120
1 Y Y NOCHECK,3174,3272,3274,3791L,6120
1 Y N NOCHECK,3820
1 Y Y AFP1,NOCHECK,3825
1 Y Y AFP1,NOCHECK,3827
1 Y Y AFP1,NOCHECK,3828
1 Y Y AFP1,NOCHECK,3829
1 Y Y AFP1,NOCHECK,3831
1 Y Y AFP1,NOCHECK,3835
1 Y Y NOCHECK,3851
1 Y Y NOCHECK,3886
1 Y Y NOCHECK,3890
1 Y Y NOCHECK,3895
1 Y Y AFP1,NOCHECK,3900
1 Y Y AFP1,NOCHECK,3935
1 N Y NOCHECK,3995,3995-SDA
16 N Y NOCHECK,3995-SDA
1 Y Y NOCHECK,3995-SDA,3995-151
1 Y Y NOCHECK,3995-SDA,3995-153
1 Y Y AFP1,NOCHECK,4000
1 Y Y NOCHECK,3174,3272,3274,3791L,6120
1 Y Y NOCHECK,4245,6120
1 Y Y NOCHECK,4248
1 Y Y NOCHECK,3174,3272,3274,3791L,6120
1 Y Y AFP1,NOCHECK,4370
1 Y Y NOCHECK,5088-1,5088-2,6098
1 Y Y NOCHECK,5088-1,5088-2,6098
1 Y Y NOCHECK,3174,3272,3274,3791L,6120
1 Y Y NOCHECK,5088-1,5088-2,6098
1 Y Y NOCHECK,5088-1,5088-2,6098
1 Y Y NOCHECK,4248,6262
1 Y N NOCHECK,7171
1 Y N NOCHECK,7770-3
32 N Y NOCHECK,8232
1 Y Y NOCHECK,2701,3704,3705,3720,3725,3745,3746

Figure 173. Supported Hardware Report (Part 20 of 29)

Appendix B. Configuration reports

421


<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UIM</th>
<th>MVS</th>
<th>VM</th>
<th>MX</th>
<th>RL</th>
<th>RH</th>
<th>RD</th>
<th>TM</th>
<th>ST</th>
<th>ATTACHABLE TO CU</th>
</tr>
</thead>
<tbody>
<tr>
<td>9032</td>
<td>051</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,9032</td>
</tr>
<tr>
<td>9032-3</td>
<td>051</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,9032,9032-3</td>
</tr>
<tr>
<td>9032-5</td>
<td>051</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,9032,9032-5</td>
</tr>
<tr>
<td>9033</td>
<td>051</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,9033</td>
</tr>
<tr>
<td>9332-40</td>
<td>258</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,6310</td>
</tr>
<tr>
<td>9332-42</td>
<td>258</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,6310</td>
</tr>
<tr>
<td>9332-60</td>
<td>258</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,6310</td>
</tr>
<tr>
<td>9332-62</td>
<td>258</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,6310</td>
</tr>
<tr>
<td>9335-81</td>
<td>258</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,6310</td>
</tr>
<tr>
<td>9336-10</td>
<td>258</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,6310</td>
</tr>
<tr>
<td>9336-20</td>
<td>258</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,6310</td>
</tr>
<tr>
<td>9345</td>
<td>002</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,9340,9341,9343,9343-1</td>
</tr>
<tr>
<td>9348-1</td>
<td>261</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>NOCHECK,6311</td>
</tr>
</tbody>
</table>

---

**KEY**

- **TYPE-MODEL** - SUPPORTED DEVICE TYPE
- **UIM** - INDEX OF UNIT INFORMATION MODULE SUPPORTING THIS DEVICE TYPE
- **MVS** - DEVICE IS SUPPORTED FOR MVS DEFINITION
- **VM** - DEVICE TYPE IS SUPPORTED FOR VM DEFINITION
- **MX** - DEVICE IS A MULTI-EXPOSURE DEVICE OR A PARALLEL ACCESS VOLUME DEVICE
- **GR** - DEVICE IS A GROUP DEVICE
- **RL** - MINIMUM NUMBER OF DEVICES TO BE DEFINED
- **RH** - MAXIMUM NUMBER OF DEVICES TO BE DEFINED
- **RD** - DEFAULT NUMBER OF DEVICES TO BE DEFINED
- **TM** - DEFAULT TIMEOUT VALUE
- **ST** - DEFAULT STADET VALUE
- **ATTACHABLE TO CU** - LIST OF CONTROL UNIT TYPES TO WHICH DEVICE TYPE IS ATTACHABLE
  - **Y** - DEVICE TYPE HAS THE CAPABILITY
  - **N** - CAPABILITY IS NOT AVAILABLE

---

Figure 173. Supported Hardware Report (Part 21 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UIM</th>
<th>GENERIC</th>
<th>DPREF</th>
<th>DYN</th>
<th>4DIG</th>
<th>UCB</th>
<th>NIP</th>
<th>SUPPORTED HARDWARE - MVS DEVICES</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFP1-0</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>FEATURE, OFFLINE, DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>CTC</td>
<td>014</td>
<td>CTC</td>
<td>8380</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>BSC1</td>
<td>026</td>
<td>AAA9</td>
<td>6900</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>BSC2</td>
<td>026</td>
<td>AAAA</td>
<td>7000</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / AUTONSRS, AUTOCALL, DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>BSC3</td>
<td>026</td>
<td>AAAB</td>
<td>7100</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / AUTOANSR, AUTOCALL, DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>CTC</td>
<td>014</td>
<td>CTC</td>
<td>8400</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE, OFFLINE, DYNAMIC, LOCANY / 370</td>
</tr>
<tr>
<td>DUMMY</td>
<td>050</td>
<td>DUMMY</td>
<td>99991</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>FCTC</td>
<td>014</td>
<td>FCTC</td>
<td>8380</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>IQD</td>
<td>058</td>
<td>IQD</td>
<td>8362</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OA</td>
<td>058</td>
<td>OA</td>
<td>8360</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OAD</td>
<td>058</td>
<td>OAD</td>
<td>8361</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OSN</td>
<td>058</td>
<td>OSN</td>
<td>8363</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>R56K</td>
<td>056</td>
<td>R56K</td>
<td>8389</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>SCTC</td>
<td>014</td>
<td>SCTC</td>
<td>8300</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>SWCH</td>
<td>051</td>
<td>SWCH</td>
<td>10500</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>BSC1</td>
<td>026</td>
<td>AAA9</td>
<td>6900</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>BSC2</td>
<td>026</td>
<td>AAAA</td>
<td>7000</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / AUTONSRS, AUTOCALL, DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>BSC3</td>
<td>026</td>
<td>AAAB</td>
<td>7100</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / AUTOANSR, AUTOCALL, DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>CTC</td>
<td>014</td>
<td>CTC</td>
<td>8400</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE, OFFLINE, DYNAMIC, LOCANY / 370</td>
</tr>
<tr>
<td>DUMMY</td>
<td>050</td>
<td>DUMMY</td>
<td>99991</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>FCTC</td>
<td>014</td>
<td>FCTC</td>
<td>8380</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>IQD</td>
<td>058</td>
<td>IQD</td>
<td>8362</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OA</td>
<td>058</td>
<td>OA</td>
<td>8360</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OAD</td>
<td>058</td>
<td>OAD</td>
<td>8361</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OSN</td>
<td>058</td>
<td>OSN</td>
<td>8363</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>R56K</td>
<td>056</td>
<td>R56K</td>
<td>8389</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>SCTC</td>
<td>014</td>
<td>SCTC</td>
<td>8300</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>SWCH</td>
<td>051</td>
<td>SWCH</td>
<td>10500</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>BSC1</td>
<td>026</td>
<td>AAA9</td>
<td>6900</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>BSC2</td>
<td>026</td>
<td>AAAA</td>
<td>7000</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / AUTONSRS, AUTOCALL, DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>BSC3</td>
<td>026</td>
<td>AAAB</td>
<td>7100</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / AUTOANSR, AUTOCALL, DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>CTC</td>
<td>014</td>
<td>CTC</td>
<td>8400</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE, OFFLINE, DYNAMIC, LOCANY / 370</td>
</tr>
<tr>
<td>DUMMY</td>
<td>050</td>
<td>DUMMY</td>
<td>99991</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>FCTC</td>
<td>014</td>
<td>FCTC</td>
<td>8380</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>IQD</td>
<td>058</td>
<td>IQD</td>
<td>8362</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OA</td>
<td>058</td>
<td>OA</td>
<td>8360</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OAD</td>
<td>058</td>
<td>OAD</td>
<td>8361</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OSN</td>
<td>058</td>
<td>OSN</td>
<td>8363</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>R56K</td>
<td>056</td>
<td>R56K</td>
<td>8389</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>SCTC</td>
<td>014</td>
<td>SCTC</td>
<td>8300</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>SWCH</td>
<td>051</td>
<td>SWCH</td>
<td>10500</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>BSC1</td>
<td>026</td>
<td>AAA9</td>
<td>6900</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>BSC2</td>
<td>026</td>
<td>AAAA</td>
<td>7000</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / AUTONSRS, AUTOCALL, DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>BSC3</td>
<td>026</td>
<td>AAAB</td>
<td>7100</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(BSCA), FEATURE, OFFLINE, (R)TCU(2701,2702,2703) / AUTOANSR, AUTOCALL, DUALCODE, DUALCOMM</td>
</tr>
<tr>
<td>CTC</td>
<td>014</td>
<td>CTC</td>
<td>8400</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE, OFFLINE, DYNAMIC, LOCANY / 370</td>
</tr>
<tr>
<td>DUMMY</td>
<td>050</td>
<td>DUMMY</td>
<td>99991</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>FCTC</td>
<td>014</td>
<td>FCTC</td>
<td>8380</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>IQD</td>
<td>058</td>
<td>IQD</td>
<td>8362</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OA</td>
<td>058</td>
<td>OA</td>
<td>8360</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OAD</td>
<td>058</td>
<td>OAD</td>
<td>8361</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>OSN</td>
<td>058</td>
<td>OSN</td>
<td>8363</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>R56K</td>
<td>056</td>
<td>R56K</td>
<td>8389</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>SCTC</td>
<td>014</td>
<td>SCTC</td>
<td>8300</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
<tr>
<td>SWCH</td>
<td>051</td>
<td>SWCH</td>
<td>10500</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE, DYNAMIC, LOCANY</td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 22 of 29)
Figure 173. Supported Hardware Report (Part 23 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UIM GENERIC</th>
<th>DPREF</th>
<th>DYNS</th>
<th>40DIG</th>
<th>UCB</th>
<th>NIP</th>
<th>SUPPORTED PARAMETERS (VALUES) / FEATURES</th>
</tr>
</thead>
</table>
| 3286-2     | 031 3286-2  | 4400  | N    | N    | N   | N   | FEATURE, OFFLINE / DOCHAR, GRCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCHAR, UKCH
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UIM</th>
<th>GENERIC</th>
<th>DPREF</th>
<th>DYN</th>
<th>4DIG</th>
<th>UCB</th>
<th>NIP</th>
<th>SUPPORTED PARAMETERS (VALUES) / FEATURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>3812</td>
<td>031</td>
<td>3286-2</td>
<td>4400</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>FEATURE,OFFLINE / DOCHAR,FRCHAR,GRCHAR,KCHAR,UKCHAR,PTREAD</td>
</tr>
<tr>
<td>3820</td>
<td>022</td>
<td>3820</td>
<td>1800</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE,DYNAMIC</td>
</tr>
<tr>
<td>3825</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>3827</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>3828</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>3829</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>3830</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>3831</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>3835</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>3886</td>
<td>032</td>
<td>3886</td>
<td>4900</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>OFFLINE</td>
</tr>
<tr>
<td>3890</td>
<td>032</td>
<td>3890</td>
<td>4800</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>OFFLINE</td>
</tr>
<tr>
<td>3895</td>
<td>032</td>
<td>3895</td>
<td>5400</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>OFFLINE</td>
</tr>
<tr>
<td>3900</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>3935</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>3995</td>
<td>053</td>
<td>3995</td>
<td>8600</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>OFFLINE,DYNAMIC,LOCANY</td>
</tr>
<tr>
<td>3995-SDA</td>
<td>053</td>
<td>3995</td>
<td>8600</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>OFFLINE,DYNAMIC,LOCANY</td>
</tr>
<tr>
<td>3995-151</td>
<td>002</td>
<td>3390</td>
<td>280</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC,LOCANY / SHARED,SHAREDUP</td>
</tr>
<tr>
<td>3995-153</td>
<td>002</td>
<td>3390</td>
<td>280</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>FEATURE,OFFLINE,DYNAMIC,LOCANY / SHARED,SHAREDUP</td>
</tr>
<tr>
<td>4000</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>4224</td>
<td>031</td>
<td>3286-2</td>
<td>4400</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>FEATURE,OFFLINE / DOCHAR,FRCHAR,GRCHAR,KCHAR,UKCHAR,PTREAD</td>
</tr>
<tr>
<td>4245</td>
<td>012</td>
<td>4245</td>
<td>1800</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE,DYNAMIC</td>
</tr>
<tr>
<td>4248</td>
<td>012</td>
<td>4245</td>
<td>1800</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE,DYNAMIC</td>
</tr>
<tr>
<td>4250</td>
<td>031</td>
<td>3286-2</td>
<td>4400</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>FEATURE,OFFLINE / DOCHAR,FRCHAR,GRCHAR,KCHAR,UKCHAR,PTREAD</td>
</tr>
<tr>
<td>4370</td>
<td>022</td>
<td>AFP1</td>
<td>1750</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC / BURSTER</td>
</tr>
<tr>
<td>5210</td>
<td>031</td>
<td>3286-2</td>
<td>4400</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>FEATURE,OFFLINE / DOCHAR,FRCHAR,GRCHAR,KCHAR,UKCHAR,PTREAD</td>
</tr>
<tr>
<td>7171</td>
<td>027</td>
<td>3791L</td>
<td>7700</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE,DYNAMIC,OWNER(VTAM,OTHER)</td>
</tr>
<tr>
<td>7770-3</td>
<td>023</td>
<td>AAA6</td>
<td>7200</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>OFFLINE,DYNAMIC,OWNER(VTAM,OTHER)</td>
</tr>
<tr>
<td>8383</td>
<td>025</td>
<td>AAA6</td>
<td>6600</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>(R)ADAPTER(TEL1),OFFLINE,SETADDR(0,1,2,3),(R)TCU(2701,2702,2703)</td>
</tr>
<tr>
<td>9032</td>
<td>051</td>
<td>SWCH</td>
<td>10500</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE,DYNAMIC,LOCANY</td>
</tr>
<tr>
<td>9032-3</td>
<td>051</td>
<td>SWCH</td>
<td>10500</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE,DYNAMIC,LOCANY</td>
</tr>
<tr>
<td>9032-5</td>
<td>051</td>
<td>SWCH</td>
<td>10500</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE,DYNAMIC,LOCANY</td>
</tr>
<tr>
<td>9033</td>
<td>051</td>
<td>SWCH</td>
<td>10500</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>OFFLINE,DYNAMIC,LOCANY</td>
</tr>
<tr>
<td>9345</td>
<td>002</td>
<td>9345</td>
<td>270</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>FEATURE,OFFLINE,DYNAMIC,LOCANY / SHARED,SHAREDUP</td>
</tr>
</tbody>
</table>

KEY

----
TYPE-MODEL - SUPPORTED DEVICE TYPE
UIM - INDEX OF UNIT INFORMATION MODULE SUPPORTING THIS DEVICE TYPE
GENERIC - GENERIC DEVICE TYPE
DPREF - DEFAULT DEVICE PREFERENCE VALUE
DYN - DEVICE SUPPORTS DYNAMIC I/O RECONFIGURATION
4DIG - DEVICE TYPE SUPPORTS DEVICE NUMBERS GREATER THAN 0FFF
UCB - DEVICE TYPE SUPPORTS UCB BEING LOCATED ABOVE 16 MB
NIP - DEVICE TYPE SUPPORTED AS NIP CONSOLE
SUPPORTED PARAMETERS (VALUES) / FEATURES - LIST OF SUPPORTED PARAMETERS (WITH SELECTION VALUES) AND FEATURES
Y - DEVICE TYPE HAS THE CAPABILITY
N - CAPABILITY IS NOT AVAILABLE
(R) - PARAMETER IS REQUIRED
PRIVATE: - START OF PARAMETERS PRIVATE TO UIM

Figure 173. Supported Hardware Report (Part 25 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UI M VM D/T</th>
<th>SUPPORTED HARDWARE - VM DEVICES</th>
<th>SUPPORTED PARAMETERS (VALUES) / FEATURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFP1-0</td>
<td>278 AFP1</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>BCTC</td>
<td>270 CTCA</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>BSC1</td>
<td>282 3705</td>
<td>N (R)ADAPTER(BSCA) BASEADD,OFFLINE,SETADDR(0,1,2,3,4) UIRATE</td>
<td></td>
</tr>
<tr>
<td>BSC2</td>
<td>282 3705</td>
<td>N (R)ADAPTER(BSCA) BASEADD,OFFLINE,SETADDR(0,1,2,3,4) UIRATE</td>
<td></td>
</tr>
<tr>
<td>BSC3</td>
<td>282 3705</td>
<td>N (R)ADAPTER(BSCA) BASEADD,OFFLINE,SETADDR(0,1,2,3,4) UIRATE</td>
<td></td>
</tr>
<tr>
<td>CTC</td>
<td>270 CTCA</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>CTC4</td>
<td>270 CTCA</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>DUMMY</td>
<td>306 DUMMY</td>
<td>N (R)CLASS(DASD,TAPE,TERM,DISPLAY,RDR,PRT,PUN,SWITCH),OFFLINE,UIRATE / DPS,RESERVE</td>
<td></td>
</tr>
<tr>
<td>FBASCSI</td>
<td>254 FBASCSI</td>
<td>N</td>
<td>OFFLINE,UIRATE, (R)ATTR(1750,2105,2107,2145,SCSI), (R)FCPDEV, (R)WWPN, (R)LUN, PREFPATH</td>
</tr>
<tr>
<td>FCP</td>
<td>254 FCP</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>FCTC</td>
<td>270 CTCA</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>HFGD</td>
<td>291 HFGD</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>ICABSCA</td>
<td>308 ICA</td>
<td>N (R)ADAPTER(BSCA), OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>ICADLAN</td>
<td>308 ICA</td>
<td>N (R)ADAPTER(ELAN), OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>ICADLNC</td>
<td>308 ICA</td>
<td>N (R)ADAPTER(HDLC), OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>ICATTEL2</td>
<td>308 ICA</td>
<td>N (R)ADAPTER(TeI).OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>IQD</td>
<td>314 OSA</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>OSA</td>
<td>314 OSA</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>OSAD</td>
<td>314 OSAD</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>OSN</td>
<td>314 OSN</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>RS8K</td>
<td>312 3088</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>SACT</td>
<td>270 CTCA</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>SMCH</td>
<td>307 9032</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>TMX</td>
<td>281 3705</td>
<td>N (R)ADAPTER(TeI).BASEADD,OFFLINE,SETADDR(0,1,2,3,4),UIRATE</td>
<td></td>
</tr>
<tr>
<td>1050</td>
<td>280 3705</td>
<td>N (R)ADAPTER(IBM), BASEADD,OFFLINE,SETADDR(0,1,2,3,4),UIRATE</td>
<td></td>
</tr>
<tr>
<td>1050X</td>
<td>280 3705</td>
<td>N (R)ADAPTER(IBM), BASEADD,OFFLINE,SETADDR(0,1,2,3,4),UIRATE</td>
<td></td>
</tr>
<tr>
<td>1287</td>
<td>288 1287</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>1288</td>
<td>288 1288</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>1403-N1</td>
<td>268 1403</td>
<td>N AFP,CHARS,CLASS,DEST, FOLD, FORM, IMAGE, OFFLINE, SEP, UIRATE / UNVCHSET</td>
<td></td>
</tr>
<tr>
<td>1403-2</td>
<td>268 1403</td>
<td>N AFP,CHARS,CLASS,DEST, FOLD, FORM, IMAGE, OFFLINE, SEP, UIRATE / UNVCHSET</td>
<td></td>
</tr>
<tr>
<td>1403-3</td>
<td>268 1403</td>
<td>N AFP,CHARS,CLASS,DEST, FOLD, FORM, IMAGE, OFFLINE, SEP, UIRATE / UNVCHSET</td>
<td></td>
</tr>
<tr>
<td>1403-5</td>
<td>268 1403</td>
<td>N AFP,CHARS,CLASS,DEST, FOLD, FORM, IMAGE, OFFLINE, SEP, UIRATE / UNVCHSET</td>
<td></td>
</tr>
<tr>
<td>1403-7</td>
<td>268 1403</td>
<td>N AFP,CHARS,CLASS,DEST, FOLD, FORM, IMAGE, OFFLINE, SEP, UIRATE / UNVCHSET</td>
<td></td>
</tr>
<tr>
<td>2032</td>
<td>307 2032</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>2250-3</td>
<td>277 2250</td>
<td>N</td>
<td>OFFLINE,UIRATE</td>
</tr>
<tr>
<td>2305-2</td>
<td>269 2305-2</td>
<td>N OFFLINE,SHARED,UIRATE</td>
<td></td>
</tr>
<tr>
<td>2501-2</td>
<td>266 2501</td>
<td>N OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>2501-2C</td>
<td>266 2501</td>
<td>N OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>2540-1</td>
<td>268 2540P</td>
<td>N OFFLINE,SHARED,UIRATE</td>
<td></td>
</tr>
<tr>
<td>3178</td>
<td>260 3178</td>
<td>Y MODEL(2,3,4,5), OFFLINE, UIRATE / EMUL3270,E3270HLD, OPRDR</td>
<td></td>
</tr>
<tr>
<td>3179</td>
<td>260 3179</td>
<td>Y MODEL(2,3,4,5), OFFLINE, UIRATE / EMUL3270,E3270HLD, OPRDR</td>
<td></td>
</tr>
<tr>
<td>3180-1</td>
<td>260 3180</td>
<td>Y OFFLINE, UIRATE / EMUL3270,E3270HLD, OPRDR</td>
<td></td>
</tr>
<tr>
<td>3190</td>
<td>260 3190</td>
<td>Y OFFLINE, UIRATE / EMUL3270,E3270HLD, OPRDR</td>
<td></td>
</tr>
<tr>
<td>3191</td>
<td>260 3190</td>
<td>Y OFFLINE, UIRATE / EMUL3270,E3270HLD, OPRDR</td>
<td></td>
</tr>
<tr>
<td>3192</td>
<td>260 3190</td>
<td>Y OFFLINE, UIRATE / EMUL3270,E3270HLD, OPRDR</td>
<td></td>
</tr>
<tr>
<td>3192-F</td>
<td>260 3190</td>
<td>Y OFFLINE, UIRATE / EMUL3270,E3270HLD, OPRDR</td>
<td></td>
</tr>
<tr>
<td>3193</td>
<td>260 3190</td>
<td>Y OFFLINE, UIRATE / EMUL3270,E3270HLD, OPRDR</td>
<td></td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 26 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UIM VM D/T</th>
<th>CONS</th>
<th>SUPPORTED PARAMETERS (VALUES) / FEATURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>3194</td>
<td>260 3190</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3203-5</td>
<td>268 3203-5</td>
<td>N</td>
<td>AFP, CHARS, CLASS, DEST, FCB, FOLD, FORM, IMAGE, OFFLINE, UIRATE, LIMIT / UNVCHSET</td>
</tr>
<tr>
<td>3211</td>
<td>268 3211</td>
<td>N</td>
<td>AFP, CHARS, CLASS, DEST, FCB, FOLD, FORM, IMAGE, INDEX, OFFLINE, UIRATE, LIMIT / UNVCHSET</td>
</tr>
<tr>
<td>3215</td>
<td>260 3215</td>
<td>Y</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3216</td>
<td>278 AFP1</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3250</td>
<td>277 3250</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3251</td>
<td>277 3250</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3262-13</td>
<td>287 3287</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3262-3</td>
<td>287 3287</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3262-5</td>
<td>268 3262</td>
<td>N</td>
<td>AFP, CHARS, CLASS, DEST, FCB, FOLD, FORM, IMAGE, OFFLINE, UIRATE, LIMIT / UNVCHSET</td>
</tr>
<tr>
<td>3268-2</td>
<td>287 3287</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3270</td>
<td>260 3270</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3274</td>
<td>283 3705</td>
<td>N</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3277-1</td>
<td>260 3277</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3277-2</td>
<td>260 3277</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3278-1</td>
<td>260 3278</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3278-2</td>
<td>260 3278</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3278-3</td>
<td>260 3278</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3279-2</td>
<td>260 3279</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3279-2A</td>
<td>260 3279</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3279-3</td>
<td>260 3279</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3279-2B</td>
<td>260 3279</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3279-2C</td>
<td>260 3279</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3279-3</td>
<td>260 3279</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3279-3A</td>
<td>260 3279</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3279-3B</td>
<td>260 3279</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3279-3X</td>
<td>260 3279</td>
<td>Y</td>
<td>OFFLINE, UIRATE / EMUL3270, E3270HLD, OPRDR</td>
</tr>
<tr>
<td>3284-1</td>
<td>287 3284</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3284-2</td>
<td>287 3284</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3286-1</td>
<td>287 3286</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3286-2</td>
<td>287 3286</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3287-1</td>
<td>287 3287</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3287-1C</td>
<td>287 3287</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3287-2C</td>
<td>287 3287</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3300</td>
<td>278 AFP1</td>
<td>N</td>
<td>OFFLINE, UIRATE</td>
</tr>
<tr>
<td>3330-1</td>
<td>257 3330-1</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3330-11</td>
<td>257 3330-11</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3330-2</td>
<td>257 3330-2</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3330V</td>
<td>271 3330</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3333-1</td>
<td>257 3330-1</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3333-11</td>
<td>257 3330-11</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3340</td>
<td>257 3340</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3344</td>
<td>257 3340</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3350</td>
<td>257 3350</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3370</td>
<td>258 3370</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3375</td>
<td>258 3375</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3380</td>
<td>258 3380</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3380-C2</td>
<td>258 3380</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3380A</td>
<td>258 3380A</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3380B</td>
<td>258 3380B</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3390</td>
<td>258 3390</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3390A</td>
<td>258 3390A</td>
<td>N</td>
<td>OFFLINE, SHARED, UIRATE</td>
</tr>
<tr>
<td>3420-3</td>
<td>261 3420-3</td>
<td>N</td>
<td>OFFLINE, UIRATE / CONV, DualDens, TRANS, 7-TRACK</td>
</tr>
<tr>
<td>3420-4</td>
<td>261 3420-4</td>
<td>N</td>
<td>OFFLINE, UIRATE / CONV, DualDens, TRANS, 7-TRACK</td>
</tr>
<tr>
<td>3420-5</td>
<td>261 3420-5</td>
<td>N</td>
<td>OFFLINE, UIRATE / CONV, DualDens, TRANS, 7-TRACK</td>
</tr>
<tr>
<td>3420-6</td>
<td>261 3420-6</td>
<td>N</td>
<td>OFFLINE, UIRATE / CONV, DualDens, TRANS, 7-TRACK</td>
</tr>
<tr>
<td>3420-7</td>
<td>261 3420-7</td>
<td>N</td>
<td>OFFLINE, UIRATE / CONV, DualDens, TRANS, 7-TRACK</td>
</tr>
</tbody>
</table>

Figure 173. Supported Hardware Report (Part 27 of 29)
<table>
<thead>
<tr>
<th>TYPE-MODEL</th>
<th>UIM GENERIC</th>
<th>DPREF DYN</th>
<th>OFFLINE,UIRATE</th>
<th>3420-8</th>
<th>261</th>
<th>3420-8</th>
<th>N</th>
<th>OFFLINE,UIRATE / CONV,DUALDENS,TRANS,7-TRACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>3422</td>
<td>261</td>
<td>3422 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3424</td>
<td>261</td>
<td>3424 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3430</td>
<td>261</td>
<td>3430 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3480</td>
<td>261</td>
<td>3480 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3490</td>
<td>261</td>
<td>3490 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3505</td>
<td>266</td>
<td>3505 N</td>
<td>CLASS,OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3525</td>
<td>268</td>
<td>3525 N</td>
<td>CLASS,FORM,OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3540</td>
<td>268</td>
<td>3540 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3590</td>
<td>261</td>
<td>3590 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3704</td>
<td>279</td>
<td>3704 N</td>
<td>(R)ADAPTER(BSCA,IBM1,TELE2,TYPE1),CPNAME,MODEL(A1,A2,A3,A4,1,2,3,4),OFFLINE,SETADDR(0,1,2,3,4),UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3705</td>
<td>279</td>
<td>3705 N</td>
<td>(R)ADAPTER(BSCA,IBM1,TELE2,TYPE4),CPNAME,MODEL(A1,A2,A3,A4,1,2,3,4),OFFLINE,SETADDR(0,1,2,3,4),UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3720</td>
<td>279</td>
<td>3720 N</td>
<td>(R)ADAPTER(TYPE4),CPNAME,RMODEL(A1,A2,A3,A4,1,2,3,4),OFFLINE,SETADDR(0,1,2,3,4),UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3725</td>
<td>279</td>
<td>3725 N</td>
<td>(R)ADAPTER(BSCA,IBM1,TELE2,TYPE4),BASEADD,CPNAME,MODEL(A1,A2,A3,A4,1,2,3,4),OFFLINE,SETADDR(0,1,2,3,4),UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3737</td>
<td>270</td>
<td>CTCA N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3745</td>
<td>279</td>
<td>3745 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3800-1</td>
<td>267</td>
<td>3800 N</td>
<td>AFP,CHARS,CLASS,DEST,DPMSIZE(1,2,3,4,5,6,7,8,9),FCB,FLASH,FORM,IMAGE,OFFLINE,SEP,UIRATE,LIMIT / 4WCGMS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3800-3</td>
<td>267</td>
<td>3800-3 N</td>
<td>AFP,CHARS,CLASS,DEST,DPMSIZE(1,2,3,4,5,6,7,8,9),FCB,FLASH,FORM,IMAGE,OFFLINE,SEP,UIRATE,MARK,LIMIT / 4WCGMS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3800-6</td>
<td>267</td>
<td>3800-6 N</td>
<td>AFP,CHARS,CLASS,DEST,DPMSIZE(1,2,3,4,5,6,7,8,9),FCB,FLASH,FORM,IMAGE,OFFLINE,SEP,UIRATE,MARK,LIMIT / 4WCGMS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3800-8</td>
<td>267</td>
<td>3800-8 N</td>
<td>AFP,CHARS,CLASS,DEST,DPMSIZE(1,2,3,4,5,6,7,8,9),FCB,FLASH,FORM,IMAGE,OFFLINE,SEP,UIRATE,MARK,LIMIT / 4WCGMS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3812</td>
<td>287</td>
<td>3812 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3816</td>
<td>287</td>
<td>3816 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3820</td>
<td>287</td>
<td>3820 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3824</td>
<td>287</td>
<td>3824 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3827</td>
<td>278</td>
<td>3827 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3828</td>
<td>278</td>
<td>3828 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3829</td>
<td>278</td>
<td>3829 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3831</td>
<td>278</td>
<td>3831 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3835</td>
<td>278</td>
<td>3835 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3851</td>
<td>271</td>
<td>3851 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3868</td>
<td>288</td>
<td>3868 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3890</td>
<td>288</td>
<td>3890 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3895</td>
<td>288</td>
<td>3895 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3900</td>
<td>288</td>
<td>3900 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3935</td>
<td>278</td>
<td>3935 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3995</td>
<td>308</td>
<td>3995 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3995-SA</td>
<td>308</td>
<td>3995-SA N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3995-151</td>
<td>258</td>
<td>3995-151 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4000</td>
<td>278</td>
<td>4000 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4224</td>
<td>287</td>
<td>4224 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4245</td>
<td>268</td>
<td>4245 N</td>
<td>AFP,CHARS,CLASS,DEST,FCB,FOLD,FORM,IMAGE,OFFLINE,SEP,UIRATE,MARK,LIMIT / 4WCGMS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4248</td>
<td>268</td>
<td>4248 N</td>
<td>AFP,CHARS,CLASS,DEST,FCB,FOLD,FORM,IMAGE,OFFLINE,SEP,UIRATE,MARK,LIMIT / 4WCGMS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4370</td>
<td>287</td>
<td>4370 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5080</td>
<td>291</td>
<td>5080 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5081</td>
<td>291</td>
<td>5081 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5210</td>
<td>287</td>
<td>5210 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6090</td>
<td>291</td>
<td>6090 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6091</td>
<td>291</td>
<td>6091 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6262</td>
<td>268</td>
<td>6262 N</td>
<td>AFP,CHARS,CLASS,DEST,FCB,FOLD,FORM,IMAGE,OFFLINE,SEP,UIRATE,MARK,LIMIT / 4WCGMS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7171</td>
<td>283</td>
<td>7171 N</td>
<td>OFFLINE,UIRATE / EMUL3270,E3270HLD,OPRDQ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8232</td>
<td>270</td>
<td>8232 N</td>
<td>OFFLINE,UIRATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TYPE-MODEL</td>
<td>UIM GENERIC</td>
<td>DPREF</td>
<td>DYN</td>
<td>4DIG</td>
<td>UCB</td>
<td>NIP</td>
<td>SUPPORTED PARAMETERS (VALUES) / FEATURES</td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>-------</td>
<td>-----</td>
<td>------</td>
<td>-----</td>
<td>-----</td>
<td>------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>9032</td>
<td>307 9032</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>9032-3</td>
<td>307 9032</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>9032-5</td>
<td>307 9032</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>9033</td>
<td>307 9033</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,UIRATE</td>
<td></td>
</tr>
<tr>
<td>9332-40</td>
<td>258 9332</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,SHARED,UIRATE,MDC(OFF,DFLT9FF,DFLTON)</td>
<td></td>
</tr>
<tr>
<td>9332-42</td>
<td>258 9332</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,SHARED,UIRATE,MDC(OFF,DFLT9FF,DFLTON)</td>
<td></td>
</tr>
<tr>
<td>9332-60</td>
<td>258 9332</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,SHARED,UIRATE,MDC(OFF,DFLT9FF,DFLTON)</td>
<td></td>
</tr>
<tr>
<td>9332-62</td>
<td>258 9332</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,SHARED,UIRATE,MDC(OFF,DFLT9FF,DFLTON)</td>
<td></td>
</tr>
<tr>
<td>9335-81</td>
<td>258 9335</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,SHARED,UIRATE,MDC(OFF,DFLT9FF,DFLTON)</td>
<td></td>
</tr>
<tr>
<td>9336-10</td>
<td>258 9336</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,SHARED,UIRATE,MDC(OFF,DFLT9FF,DFLTON)</td>
<td></td>
</tr>
<tr>
<td>9336-20</td>
<td>258 9336</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,SHARED,UIRATE,MDC(OFF,DFLT9FF,DFLTON)</td>
<td></td>
</tr>
<tr>
<td>9345</td>
<td>258 9345</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,SHARED,UIRATE,MDC(OFF,DFLT9FF,DFLTON)</td>
<td></td>
</tr>
<tr>
<td>9348-1</td>
<td>261 9348</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OFFLINE,UIRATE / DUALDENS</td>
<td></td>
</tr>
</tbody>
</table>

**KEY**

---

**TYPE-MODEL** - Supported device type

**UIM** - Index of unit information module supporting this device type

**VM D/T** - VM device type

**CONS** - Device type is supported as VM console

**SUPPORTED PARAMETERS (VALUES) / FEATURES** - List of supported parameters (with selection values) and features

- **Y** - Device type has the capability
- **N** - Capability is not available
- **(R)** - Parameter is required
## I/O Definition Reference

### Physical Configuration Information

### Physical Device Types

#### Physical Device
- **3490-B20**
- **3490-B40**
- Attachable to the following control units:
  - 3490-A10
  - 3490-A20

#### Physical Device (Integrated Tape Subsystem)
- **3490-C10**
- **3490-C11**
- **3490-C22**

#### Physical Device (Integrated Tape Subsystem inside a 3494)
- **3490-C1A**
- **3490-C2A**

3494 and 3495 are tape libraries containing automation, library manager, one or more tape control units, storage cells and tape cartridges. To prepare the IODF, no need to define 3494 and 3495 explicitly. However, need to indicate the tape devices as library tape devices by specifying LIBRARY=YES in the device definition.

### Logical Configuration Rules

#### For Channel Subsystem
- Up to maximum 4 channel paths for A10, 8 for A20, 2 for C1A or C2A.
- Range of 16 unit addresses for control unit. Also accept a minimum of 2 addresses for integrated tape subsystem.

#### For Operating System
- LIBRARY=YES, if devices are installed in a system-managed IBM 3494 or IBM 3495 Tape Library.
- LIBRARY=YES or NO for BTLS managed library drives.
- DYNAMIC=YES, if devices are dynamically reconfigured.
- AUTOSWITCH=YES, if tape drives are dynamically switched between system (MVS/ESA 5.2.0)

### Configuration Example

```
CHPID PATH=(22,27),TYPE=CNC
```

---

*Figure 174. I/O Definition Reference (Part 1 of 5)*
Software Prerequisites
Minimum version and release to operate the device in MVS/ESA environment:
- MVS/SP - JES2 3.1
- MVS/DFP 3.1

Minimum Product Levels
- DFSORT release 11 (release 12 for C1A,C2A)
- EREP 3.4 (3.5 for C1A,C2A)
- DFHSM 2.6
- DFDS 2.5

References
- MVS/ESA & MVS/XA Support for 3490 Magnetic Tape Subsystem. GC28-1141
- IBM 3490E Planning and Migration Guide GC35-0219
- IBM 3490 Planning and Migration Guide GC35-0116

Notes
- For the latest information, contact your local IBM Marketing Representative.

Device number definition values: ADDRESS={device number<,number-of-devices>}

device number: 1 - 4 hexadecimal number in the range 0000 - FFFF.
number-of-devices: Number of sequential device numbers to be assigned to the devices.
Minimum value: 1
Default value: 1
Maximum value: 4095

Channel Subsystem information:

When attached to a parallel interface:
I/O interface time out function default: TIMEOUT=YES
Status verification facility default: STADET=YES

---

Figure 174. I/O Definition Reference (Part 2 of 5)
Unit address: UNITADD=xx
The unit address is a hexadecimal value from 00 to FF which must be specified in the unit address range of the control unit. Default unit address are the last 2 digits of the device number.

MVS configuration information:
-------------------------------
Generic name: 3490 It may be modified dependent on the specified features.
Support of dynamic I/O reconfiguration: Yes
Support of device numbers greater than OFFF: Yes
Support of UCBs above 16 MB storage: Yes

Required parameters: None.
Optional parameters:

OFFLINE Device considered online or offline at IPL
Default value: OFFLINE=NO
Specifies whether MVS is to consider the device online or offline at IPL.
Yes The device is considered offline at IPL.
No The device is considered online at IPL.
(Default)
If MVS needs the device during IPL, specify No.

DYNAMIC Device supports dynamic configuration
Specify yes to indicate that the device is to be eligible for Dynamic I/O Configuration.

LIBRARY Device supports auto tape library
Data type is YES or NO
Pre-selected: No
Specify YES to indicate that the device belongs to an automated tape library.

AUTOSWITCH Device is automatically switchable
Data type is YES or NO
Pre-selected: No
Specify YES to indicate that the device should...
be treated as an automatically switchable device

Supported features:

ALTCRTL       Separate physical control unit path

Specify Yes to indicate that there is a separate physical control unit path to the device.

SHARABLE       Device is sharable between systems

Specify Yes to indicate that the 3803 two-channel switch is used for partitioning and that magnetic tape drives can be shared between two processors.

Do not allocate or unload a shared tape drive.

If specify Yes for SHARABLE, HCD forces a value of Yes for the OFFLINE parameter, even if you specify No for OFFLINE.

COMPACT        Compaction

Specify Yes to indicate that compaction is available for tape devices.

Compaction is a method of compressing and encoding data in order to reduce storage space.
Graphical configuration reports

This section contains one example for an LCU report. "Create or view graphical configuration reports" on page 238 describes how to produce this report.

LCU Report

The LCU report shows all logical control units for the designated processor. Each diagram shows one or more logical control units.
The following figures show examples of IODF compare reports that can be produced by the IODF compare function of HCD. Not all possible reports are shown, and not all examples are shown completely. "HCD compare functions" on page 246 describes how to produce these reports.

For XMP processors, the processorID for the comparison is shown together with the channel subsystem in question, for example, XMP01.1.

With all IODF reports where channel subsystems of XMP processors are involved, you can compare two channel subsystems.

You can also compare an SMP processor to a channel subsystem of an XMP processor. If this kind of comparison is limited by processor only, the SMP processor is compared to channel subsystem 0 of the XMP processor.

### Processor Compare Report

<table>
<thead>
<tr>
<th>PROC</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GOLDENE1</td>
<td>Actual Data</td>
<td>Old Data</td>
<td>with 4 CSSes prima + ry 2094-S38</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Processor Description continued</td>
</tr>
<tr>
<td>GOLDENE2</td>
<td>Added</td>
<td>LPAR 2094-S28</td>
<td>with 4 CSSes secon + dary 2094-S28</td>
</tr>
<tr>
<td></td>
<td>80800002795A209409 + -11-2314:37:54</td>
<td>80800002795A209409 + -11-1316:53:04</td>
<td>Processor Local System Name</td>
</tr>
</tbody>
</table>

Figure 175. Processor Compare Report
Channel Subsystem Compare Report

New IODF name: BOKA.IODFC1.MIG
Old IODF name: BOKA.IODFC1.MIG1

Limited to New Processor Id: XMPBIG
Old Processor Id: XMPBIG
Limited to New CSS Id: 0
Old CSS Id: 0

<table>
<thead>
<tr>
<th>PROC CSS ID</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XMPBIG 0</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td>64512</td>
<td>64510</td>
<td>Maximum Numbers of Devices in Subchannel Set 0</td>
<td></td>
</tr>
<tr>
<td>32180</td>
<td>11000</td>
<td>Maximum Numbers of Devices in Subchannel Set 1</td>
<td></td>
</tr>
</tbody>
</table>

Figure 176. Channel Subsystem Compare Report

Partition Compare Report

New IODF name: SEL.IODF00.COMP200.NEW
Old IODF name: SEL.IODF00.COMP200.OLD

Limited to New Processor Id: PROCCHK
Old Processor Id: PROCCHK

<table>
<thead>
<tr>
<th>PROC PART</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROCCHK.0 PART1</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>same</td>
<td>same</td>
<td></td>
</tr>
<tr>
<td>OS</td>
<td>same</td>
<td></td>
<td></td>
</tr>
<tr>
<td>test_system_1</td>
<td>same</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROCCHK.0 PART2</td>
<td>Added</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>same</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OS</td>
<td>same</td>
<td></td>
<td></td>
</tr>
<tr>
<td>test_system_2</td>
<td>same</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 177. Partition Compare Report

Appendix B. Configuration reports 437
## Channel Path Compare Report

### Table: Channel Path Compare Report

<table>
<thead>
<tr>
<th>PROC</th>
<th>CHPID</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GOLDENE1.0 50</td>
<td>Added</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>555</td>
<td></td>
<td></td>
<td>Physical Channel ID</td>
</tr>
<tr>
<td></td>
<td>OSD</td>
<td></td>
<td></td>
<td>Channel Path Type</td>
</tr>
<tr>
<td></td>
<td>SPAN</td>
<td></td>
<td></td>
<td>Channel Path Operation Mode</td>
</tr>
<tr>
<td></td>
<td>up to 1920 subchan +</td>
<td></td>
<td></td>
<td>continued</td>
</tr>
<tr>
<td></td>
<td>nels</td>
<td></td>
<td></td>
<td>Connects to Dynamic Switch</td>
</tr>
<tr>
<td></td>
<td>undefined</td>
<td></td>
<td></td>
<td>Channel Path Managed Indicator</td>
</tr>
<tr>
<td></td>
<td>not managed</td>
<td></td>
<td></td>
<td>Channel Path I/O Cluster Name</td>
</tr>
<tr>
<td></td>
<td>none</td>
<td></td>
<td></td>
<td>MTU size (in KB)</td>
</tr>
<tr>
<td></td>
<td>n/a</td>
<td></td>
<td></td>
<td>HCA Adapter ID</td>
</tr>
<tr>
<td></td>
<td>n/a</td>
<td></td>
<td></td>
<td>HCA Port</td>
</tr>
<tr>
<td></td>
<td>disabled</td>
<td></td>
<td></td>
<td>Queue Prioritization</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; GECSS0FX</td>
<td></td>
<td></td>
<td>Partition in Access List</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; GECSS01X</td>
<td></td>
<td></td>
<td>Partition in Access List</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; GECSS03X</td>
<td></td>
<td></td>
<td>Partition in Access List</td>
</tr>
<tr>
<td>GOLDENE1.0 51</td>
<td>Added</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>n/a</td>
<td></td>
<td></td>
<td>Physical Channel ID</td>
</tr>
<tr>
<td></td>
<td>IQD</td>
<td></td>
<td></td>
<td>Channel Path Type</td>
</tr>
<tr>
<td></td>
<td>SHR</td>
<td></td>
<td></td>
<td>Channel Path Operation Mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Connects to Dynamic Switch</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Channel Path Managed Indicator</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Channel Path I/O Cluster Name</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MTU size (in KB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HCA Adapter ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HCA Port</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Queue Prioritization</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; GECSS0FX</td>
<td></td>
<td></td>
<td>Partition in Access List</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; GECSS01X</td>
<td></td>
<td></td>
<td>Partition in Access List</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; GECSS09X</td>
<td></td>
<td></td>
<td>Partition in Candidate List</td>
</tr>
</tbody>
</table>

Figure 178. Channel Path Compare Report
### Control Unit Attachment Compare Report

Control Unit Attachment Compare Report  
TIME: 14:21  DATE: 2000-12-08  PAGE D - 4

New IODF name: BOKA.IODF38  
Old IODF name: BOKA.IODF38.TEMP

Limited to New Processor Id: FR38LPAR  Old Processor Id: FR38LPAR

<table>
<thead>
<tr>
<th>PROC</th>
<th>CU</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FR38LPAR</td>
<td>2000</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>serial</td>
<td>same</td>
<td>same</td>
<td>Control Unit Attachment Type</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>S *</td>
<td>same</td>
<td>Protocol</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2 *</td>
<td>same</td>
<td>I/O Concurrency Level</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>same</td>
<td>Control Unit Address</td>
</tr>
<tr>
<td></td>
<td>00,256</td>
<td>00,256</td>
<td>same</td>
<td>Unit Address, Number of addresses</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; 0C.70</td>
<td>&gt;&gt; same</td>
<td>&gt;&gt; same</td>
<td>Connected Channel Path, Destination Link Address</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; 68.9E</td>
<td>&gt;&gt; same</td>
<td>&gt;&gt; same</td>
<td>Number of Connected Managed Channel Paths</td>
</tr>
<tr>
<td>FR38LPAR</td>
<td>2200</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>serial</td>
<td>same</td>
<td>same</td>
<td>Control Unit Attachment Type</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>S *</td>
<td>same</td>
<td>Protocol</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2</td>
<td>same</td>
<td>I/O Concurrency Level</td>
</tr>
<tr>
<td></td>
<td>00,256</td>
<td>00,256</td>
<td>same</td>
<td>Unit Address, Number of addresses</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; 0B.B9</td>
<td>&gt;&gt; same</td>
<td>&gt;&gt; same</td>
<td>Connected Channel Path, Destination Link Address</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; 17.9B</td>
<td>&gt;&gt; same</td>
<td>&gt;&gt; same</td>
<td>Number of Connected Managed Channel Paths</td>
</tr>
</tbody>
</table>

**Figure 179. Control Unit Attachment Compare Report**
**Device Attachment Compare Report**

New IODF name: BOKA.IODF75.WORK.R17  
Old IODF name: BOKA.IODF77.WORK.R17

Limited to New Processor Id: GOLDENE1  
Old Processor Id: GOLDENE1

Limited to New CSS Id: 0  
Old CSS Id: 0

<table>
<thead>
<tr>
<th>PROC</th>
<th>Device, Range</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GOLDENE1.0 0000,8</td>
<td>Actual Data</td>
<td>Old Data</td>
<td>0</td>
<td>Subchannel Set ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>Unit Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
<td>Illegal Status Detection Facility</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>No</td>
<td>Timeout Facility</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>undefined</td>
<td>Preferred Channel Path</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; GECSS0FX</td>
<td>Partition in Explicit Device Candidate list</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; GECSS01X</td>
<td>Partition in Explicit Device Candidate list</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; GECSS03X</td>
<td>Partition in Explicit Device Candidate list</td>
</tr>
<tr>
<td>GOLDENE1.0 0008,8</td>
<td>Actual Data</td>
<td>Old Data</td>
<td>0</td>
<td>Subchannel Set ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>08</td>
<td>Unit Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
<td>Illegal Status Detection Facility</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>No</td>
<td>Timeout Facility</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>undefined</td>
<td>Preferred Channel Path</td>
</tr>
</tbody>
</table>

**Control Unit Compare Report**

New IODF name: REDDE.IODF00.COMP1  
Old IODF name: REDDE.IODF00.COMP2

Limited to New Processor Id: LPARPROC  
Old Processor Id: LPARPROC

Limited to New Partition Id: PART1  
Old Partition Id: PART3

<table>
<thead>
<tr>
<th>CU</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0200</td>
<td>Actual Data</td>
<td>Old Data</td>
<td>399D</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>same</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>same</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; BASPROC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; LPARPROC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; P208A1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; 0200,15</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; 020F</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; 0210,11</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; 01 E4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; 01 E5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt;&gt; access list</td>
</tr>
</tbody>
</table>

---

Figure 180. Device Attachment Compare Report

Figure 181. Control Unit Compare Report
### Device Compare Report

**New IODF name:** REDDE.IODF00.COMP1  **Old IODF name:** REDDE.IODF00.COMP2

**Limited to New Processor Id:** LPARPROC  **Old Processor Id:** LPARPROC

**Limited to New Partition Id:** PART1  **Old Partition Id:** PART3

<table>
<thead>
<tr>
<th>Device, Range</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0100</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td>9032</td>
<td>same</td>
<td>same</td>
<td>Device Type</td>
</tr>
<tr>
<td></td>
<td>same</td>
<td>same</td>
<td>Serial Number</td>
</tr>
<tr>
<td></td>
<td>same</td>
<td>same</td>
<td>Device Description</td>
</tr>
<tr>
<td></td>
<td>same</td>
<td>same</td>
<td>VOLSER</td>
</tr>
<tr>
<td>&gt;&gt; BASPROC</td>
<td>&gt;&gt; new added</td>
<td>Attached to Processor</td>
<td></td>
</tr>
<tr>
<td>&gt;&gt; LPARPROC</td>
<td>&gt;&gt; same</td>
<td>Attached to Processor</td>
<td></td>
</tr>
<tr>
<td>&gt;&gt; P2084.1</td>
<td>&gt;&gt; same</td>
<td>Attached to Processor</td>
<td></td>
</tr>
<tr>
<td>&gt;&gt; 0100</td>
<td>&gt;&gt; same</td>
<td>Connected to Control Unit</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>0200,15</th>
<th>Actual Data</th>
<th>Old Data</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3390</td>
<td>same</td>
<td>same</td>
<td>Device Type</td>
</tr>
<tr>
<td></td>
<td>same</td>
<td>same</td>
<td>Serial Number</td>
</tr>
<tr>
<td></td>
<td>same</td>
<td>same</td>
<td>Device Description</td>
</tr>
<tr>
<td>&gt;&gt; BASPROC</td>
<td>&gt;&gt; new added</td>
<td>Attached to Processor</td>
<td></td>
</tr>
<tr>
<td>&gt;&gt; LPARPROC</td>
<td>&gt;&gt; same</td>
<td>Attached to Processor</td>
<td></td>
</tr>
<tr>
<td>&gt;&gt; 0200</td>
<td>&gt;&gt; same</td>
<td>Connected to Control Unit</td>
<td></td>
</tr>
<tr>
<td>&gt;&gt; 0210</td>
<td>&gt;&gt; same</td>
<td>Connected to Control Unit</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 182. Device Compare Report**
Switch Compare Report

<table>
<thead>
<tr>
<th>SWITCH</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>9032</td>
<td>same</td>
<td>Switch Type</td>
</tr>
<tr>
<td></td>
<td></td>
<td>same</td>
<td>Switch Serial Number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>same</td>
<td>Description</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; 1000 1000</td>
<td>&gt;&gt; same</td>
<td>Switch Control unit, switch device</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; 1010 1010</td>
<td>&gt;&gt; same</td>
<td>Switch Control unit, switch device</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; 1020 1020</td>
<td>&gt;&gt; same</td>
<td>Switch Control unit, switch device</td>
</tr>
</tbody>
</table>

Switch Detail Compare Report

<table>
<thead>
<tr>
<th>SWITCH PORT</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 B0</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>installed</td>
<td>same</td>
<td>Port Installed Flag</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Port Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; 0800</td>
<td>&gt;&gt; same</td>
<td>Attached to Control Unit</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; P2084.1 31</td>
<td>&gt;&gt; same</td>
<td>Attached to Processor, Channel Path</td>
</tr>
<tr>
<td>01 B1</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>installed</td>
<td>same</td>
<td>Port Installed Flag</td>
</tr>
<tr>
<td></td>
<td>CU_400</td>
<td>CU_500</td>
<td>Port Name</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01 B2</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>installed</td>
<td>same</td>
<td>Port Installed Flag</td>
</tr>
<tr>
<td></td>
<td>PROCA_CP18</td>
<td>not occupied</td>
<td>Port Occupied Flag</td>
</tr>
</tbody>
</table>
Switch Configuration Compare Report
The Switch Configuration Compare Report compares the switch configurations contained in the specified IODFs.

<table>
<thead>
<tr>
<th>SWITCH</th>
<th>SWCONFIG</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>BASIC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Actual Data</td>
<td></td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PROHIBIT</td>
<td></td>
<td>ALLOW</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SW Building 01-125</td>
<td></td>
<td>same</td>
<td>Switch Configuration Description</td>
</tr>
</tbody>
</table>

Figure 185. Switch Configuration Compare Report

Switch Configuration Detail Compare Report

<table>
<thead>
<tr>
<th>SWITCH</th>
<th>SWCONFIG PORT</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>BASIC E1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Actual Data</td>
<td></td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unblocked</td>
<td></td>
<td>same</td>
<td>Blocked / Unblocked Connection</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; F1</td>
<td></td>
<td>&gt;&gt; same</td>
<td>Port of Dedicated Connection</td>
</tr>
<tr>
<td>01</td>
<td>BASIC E2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Actual Data</td>
<td></td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unblocked</td>
<td></td>
<td>same</td>
<td>Blocked / Unblocked Connection</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; E7</td>
<td></td>
<td>&gt;&gt; same</td>
<td>Port of Allowed Connection</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; F3</td>
<td></td>
<td>&gt;&gt; same</td>
<td>Port of Allowed Connection</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; E8</td>
<td></td>
<td>&gt;&gt; same</td>
<td>Port of Prohibited Connection</td>
</tr>
<tr>
<td>01</td>
<td>BASIC E3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Actual Data</td>
<td></td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unblocked</td>
<td></td>
<td>same</td>
<td>Blocked / Unblocked Connection</td>
</tr>
<tr>
<td></td>
<td>&gt;&gt; E8</td>
<td></td>
<td>&gt;&gt; same</td>
<td>Port of Prohibited Connection</td>
</tr>
</tbody>
</table>

Figure 186. Switch Configuration Detail Compare Report
# Esoteric Compare Report

<table>
<thead>
<tr>
<th>OS/CONFIG EDIT</th>
<th>ESOTERIC</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS1 01 BOBO</td>
<td></td>
<td>Deleted</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No</td>
<td></td>
<td>Esoteric is VIO Eligible</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt;&gt; 0201,3</td>
<td></td>
<td>Assigned Device, Range</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt;&gt; 0205,3</td>
<td></td>
<td>Assigned Device, Range</td>
</tr>
<tr>
<td>OS1 01 HUGO</td>
<td>Added</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No</td>
<td></td>
<td>Esoteric is VIO Eligible</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt;&gt; 0200,8</td>
<td></td>
<td>Assigned Device, Range</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt;&gt; 0300,64</td>
<td></td>
<td>Assigned Device, Range</td>
</tr>
<tr>
<td>OS1 01 SYSDA</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>same</td>
<td></td>
<td>Esoteric is VIO Eligible</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt;&gt; 0204</td>
<td></td>
<td>Assigned Device, Range</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&gt;&gt; 0300,64</td>
<td></td>
<td>Assigned Device, Range</td>
</tr>
</tbody>
</table>

*Figure 187. Esoteric Compare Report*
**OS Device Compare Report**

**New IODF name:** REDDE.IODF00.COMP1  
**Old IODF name:** REDDE.IODF00.COMP2

**Limited to New Operating System Id:** OS1  
**Limited to Old Operating System Id:** OS1

<table>
<thead>
<tr>
<th>OS</th>
<th>Device, Range</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS1</td>
<td>0100</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>9033</td>
<td>same</td>
<td></td>
<td>Device Type</td>
</tr>
<tr>
<td></td>
<td>SWCH</td>
<td>same</td>
<td></td>
<td>Name of Generic</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>same</td>
<td></td>
<td>Value(s) of Parameter OFFLINE</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>same</td>
<td></td>
<td>Value(s) of Parameter DYNAMIC</td>
</tr>
<tr>
<td>OS1</td>
<td>0200,32</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3390</td>
<td>same</td>
<td></td>
<td>Device Type</td>
</tr>
<tr>
<td></td>
<td>3390</td>
<td>same</td>
<td></td>
<td>Name of Generic</td>
</tr>
<tr>
<td></td>
<td>No *</td>
<td>same</td>
<td></td>
<td>Value(s) of Parameter OFFLINE</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>same</td>
<td></td>
<td>Value(s) of Parameter DYNAMIC</td>
</tr>
<tr>
<td>OS1</td>
<td>01D1,4</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3390A</td>
<td>same</td>
<td></td>
<td>Device Type</td>
</tr>
<tr>
<td></td>
<td>3390</td>
<td>same</td>
<td></td>
<td>Name of Generic</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td></td>
<td>Subchannel Set ID</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>same</td>
<td></td>
<td>Value(s) of Parameter OFFLINE</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>same</td>
<td></td>
<td>Value(s) of Parameter DYNAMIC</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>SHARED</td>
<td>same</td>
<td></td>
<td>Feature</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

* indicates this value as default value (only shown when both sides exist)

**Figure 188. OS Device Compare Report**

**Note:** Device ranges in a subchannel set with a subchannel set ID > 0 are displayed in a 5-digit notation with the leading digit indicating the subchannel set ID. For example, a device range \( 1000,64 \) located in subchannel set 1 is shown as \( 11000,64 \). A device range \( 2000,32 \) in subchannel set 0 is shown as \( 2000,32 \).
### OS Console Compare Report

**New IODF name:** USER.IODF03.WORK  
**Old IODF name:** HCD.IODF01.WORK  
**Limited to New Operating System Id:** OPSYS01  
**Old Operating System Id:** OPSYS01

<table>
<thead>
<tr>
<th>OSCONFIG DEVICE</th>
<th>New IODF</th>
<th>Old IODF</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPSYS01 0001</td>
<td>Actual Data</td>
<td>Old Data</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Order Number</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>OPSYS01 0002</td>
<td>Added</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Order Number</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Devices relate to the limiting LPAR via CHPIDs which have the limiting LPAR in the candidate list only.

### CSS / OS Device Compare Report

**IODF Name:** SEL.IODF00.WORK1  
**Processor Id:** PROC01  
**Css Id:** 0  
**Partition Name:** PART00/.  
**Operating System Configuration Id:** MVS1

<table>
<thead>
<tr>
<th>Device, Range</th>
<th>CSS Device Type</th>
<th>OS Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>A000,80</td>
<td>3390B</td>
<td>same</td>
</tr>
<tr>
<td>A050,176</td>
<td>3390A</td>
<td>same</td>
</tr>
<tr>
<td>A200,80</td>
<td>3390B</td>
<td>same</td>
</tr>
<tr>
<td>E210,00</td>
<td>3390</td>
<td>3800-1</td>
</tr>
<tr>
<td>E220,1000</td>
<td>3390</td>
<td>same</td>
</tr>
</tbody>
</table>

*Devices relate to the limiting LPAR via CHPIDs but the device is excluded from the CSS with an explicit device candidate list.

*Devices relate to the limiting LPAR via CHPIDs which have the limiting LPAR in the candidate list only.
Appendix C. Problem determination for HCD

Overview

The information in this appendix is intended to help you diagnose problems that may arise with definitions that were created by the use of HCD. It explains:

- How to identify problems
- What diagnostic information and tools you can use
- How to search problem reporting data bases

Because HCD is part of z/OS, problems with HCD must generally be handled as explained in z/OS Problem Management.

Product Identifiers

- Module Prefix: CBD
- Component ID: 5695SC1XL

For more information on identifiers, refer to the z/OS Program Directory.

Identifying problems

Before you can begin to diagnose a system problem, you have to know what kind of problem you have.

The following table contains examples of symptoms you can use to determine a problem. Each symptom refers to a corresponding section for further problem diagnostic.

Table 12. Symptoms of system problems

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Corresponding Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCD terminates abnormally</td>
<td>“HCD abnormal termination” on page 448</td>
</tr>
<tr>
<td>Wait State during IPL</td>
<td>“Error during IPL (Wait State Codes)” on page 450</td>
</tr>
<tr>
<td>A function key assignment does not match the functions that can be performed on the panel.</td>
<td>“Problems with panels and function key assignment” on page 451</td>
</tr>
<tr>
<td>Messages CBDA400I to CBDA420I are displayed</td>
<td>“Problems with help information provided by HCD” on page 452</td>
</tr>
<tr>
<td>Output of textual report is incorrect or incomplete</td>
<td>“Problems with output of HCD textual reports” on page 453</td>
</tr>
<tr>
<td>Output of graphical report is incorrect or incomplete</td>
<td>“Problems with output of HCD graphical reports” on page 454</td>
</tr>
<tr>
<td>Messages during initialization of HCD</td>
<td>“Problems during initialization of HCD” on page 456</td>
</tr>
<tr>
<td>A string like ?PARMnn? appears on the Define Device Parameters / Features panel</td>
<td>“Problems with UIMs” on page 457</td>
</tr>
<tr>
<td>A UIM is flagged in error on the Installed UIMs panel</td>
<td>“Problems with UIMs” on page 457</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1994, 2009
Table 12. Symptoms of system problems (continued)

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Corresponding Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Messages during migration</td>
<td>“Resolving migration errors” on page 302</td>
</tr>
<tr>
<td>HCD does not display an error message when you make a mistake</td>
<td>“HCD internal problems” on page 458</td>
</tr>
<tr>
<td>An HCD generated IOCP input data set fails when using the IOCP program</td>
<td>“HCD internal problems” on page 458</td>
</tr>
<tr>
<td>Transmit configuration package action does not produce the expected results</td>
<td>“Problems with ‘Transmit Configuration Package’ action” on page 458</td>
</tr>
<tr>
<td>HCD LDAP backend terminates abnormally</td>
<td>“HCD LDAP backend abnormal termination” on page 463</td>
</tr>
<tr>
<td>HCD LDAP backend returns one of the error codes LDAP_OPERATIONS_ERROR, LDAP_NO_MEMORY, or LDAP_LOCAL_ERROR</td>
<td>“Adverse LDAP return code from the HCD LDAP Backend” on page 464</td>
</tr>
<tr>
<td>Message CBDO009E is printed to the started task’s log</td>
<td>“Error in the HCD LDAP backend” on page 464</td>
</tr>
</tbody>
</table>

**HCD abnormal termination**

If HCD terminates abnormally, view the HCD message log that contains the termination message CBDA0001 specifying the system abend code (also called *abend code*) and the reason code in the HCD message log.

Refer to the message descriptions shown in *z/OS and z/VM HCD Messages* Take the action as described under “Programmer Response”. If the message points to a probable logic error in one of the modules of HCD, develop a search argument for the problem-reporting data bases. If the search finds that the problem has been reported before, request the problem fix; if not, report the problem to IBM. For a list of additional information that should be provided, see the appropriate message explanation.

**Diagnosing system abend code ‘00F’**

If HCD terminates with system abend code ‘00F’, this abend code is accompanied by a reason code, which refers to one of the HCD messages describing the reason of the failure. Note that you have to view the HCD message log for the system abend code and reason code. The reason code consists of eight digits and has the format ‘mnnnllll’ where:

- **m** is the prefix indicating the HCD message range:
  - 0: CBDAxxx messages
  - 1: CBDBxxxx messages
  - 2: CBDCxxxx messages
  - 6: CBDGxxxx messages

- **nnn** is the message number within the HCD message range.

- **llll** is the message reason code describing in more detail the reason of the message.

The information provided by the abend code can be used as a quick reference into the message. For example, the reason code 00990106 means that:

- The message CBDA099I was issued.
- The message reason code is 0106.

The reason code 00150095 means that:

- The message CBDA015 was issued.
The message reason code is 95.

Table 13 and Table 14 show what the search argument and the problem data could look like.

**Table 13. Search Argument**

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>AB/S0hhh</td>
<td>System abend code</td>
<td>AB/S000F</td>
</tr>
<tr>
<td>PRCS/mnnlllll</td>
<td>Reason code</td>
<td>PRCS/00990106</td>
</tr>
<tr>
<td>MS/ccccnnms</td>
<td>Message identifier</td>
<td>MS/CBDA099I</td>
</tr>
</tbody>
</table>

**Table 14. Problem Data**

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBDA000 abend code and reason code</td>
<td>00F and 00990406</td>
</tr>
<tr>
<td>CBDA099 reason code</td>
<td>406</td>
</tr>
<tr>
<td>CBDA099 additional error information (content of HCDMLOG)</td>
<td></td>
</tr>
<tr>
<td>It is important that all information shown in HCDMLOG is recorded.</td>
<td></td>
</tr>
<tr>
<td>The ID of the panel where the error occurred</td>
<td>CBDPPRF0</td>
</tr>
<tr>
<td>Description of what type of action the user wanted to perform when the problem occurred</td>
<td>Add a Processor</td>
</tr>
<tr>
<td>The TRACE output data set (See &quot;TRACE command&quot; on page 469 for instructions how to produce an HCD trace output.)</td>
<td></td>
</tr>
</tbody>
</table>

**ISPF list file and abend panel:** Additional information on errors may be recorded in the ISPF list file.

For abends, additional information may be displayed on the ISPF abend panel.

**Diagnosing system abend other than '00F'**

If HCD terminates with an abend code other than '00F' (indicated in the terminating message), proceed as follows:

1. Look at the explanation of the abend code and any reason code that accompanies the abend code. Take the recommended actions.
2. Look for any messages that accompany the abend. Take the recommended actions.
3. Obtain the SYS1.LOGREC record. (Format the SYS1.LOGREC record using EREP.)
4. In SYS1.LOGREC find the SDWAVRA information which is as follows:
   - The CSECT (module) names found in the diagnostic stack.
     The CSECT names are separated by a blank. The SDWAVRA contains all CSECT names from the diagnostic stack as long as they fit into it. If the SDWAVRA is too small to contain all names, the premature end of the CSECT name trace is indicated by an asterisk.
   - The data from each diagnostic stack entry that is marked as VRA data.
This is normally the input parameter list of the modules corresponding to the CSECT name trace.

Table 15 and Table 16 show what the search argument and the problem data associated with our example could look like.

Table 15. Search Argument

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIDS/CBDcccccc</td>
<td>CSECT name</td>
<td>RIDS/CBDMGHCP</td>
</tr>
<tr>
<td>AB/S0hhhh</td>
<td>System abend code</td>
<td>AB/S0106</td>
</tr>
<tr>
<td>PRCS/mnnnlllll</td>
<td>Reason code</td>
<td>PRCS/0000000B</td>
</tr>
<tr>
<td>MS/ccnnns</td>
<td>Message identifier</td>
<td>MS/CSV011I</td>
</tr>
<tr>
<td>FLDS/SDWAVRA</td>
<td>SDWAVRA contents</td>
<td></td>
</tr>
<tr>
<td>VALU/cccc</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 16. Problem Data

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYS1.LOGREC error record</td>
<td></td>
</tr>
<tr>
<td>SDWAVRA information</td>
<td></td>
</tr>
<tr>
<td>Accompanying messages</td>
<td></td>
</tr>
<tr>
<td>Component ID and FMID</td>
<td></td>
</tr>
<tr>
<td>Linkage editor output</td>
<td></td>
</tr>
<tr>
<td>Description of what type of action the user wanted to perform when the problem occurred</td>
<td></td>
</tr>
<tr>
<td>The TRACE output data set</td>
<td></td>
</tr>
<tr>
<td>(See “TRACE command” on page 469 for instructions how to produce an HCD trace output.)</td>
<td></td>
</tr>
</tbody>
</table>

Error during IPL (Wait State Codes)

IOS may issue wait state codes during IPL when using an IODF to perform an IPL. The wait state codes indicate that there is a problem, for example, with an IODF data set or with device specifications in the IODF or UIM. The reason codes with the wait state codes point to the cause of the problem. For information about the codes, refer to z/OS MVS System Codes.

Depending on the code that was issued, you have to use the arguments that apply to the specific situation listed in the box below. The same applies for submitting problem data.

Table 17 and Table 18 on page 451 show what the search argument and the problem data could look like.

Table 17. Search Argument

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>WS/D0hhhh</td>
<td>Wait state code</td>
<td>WS/D0083</td>
</tr>
<tr>
<td>PRCS/mnnnlllll</td>
<td>Reason code</td>
<td>PRCS/00000002</td>
</tr>
</tbody>
</table>


Table 17. Search Argument (continued)

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS/ccnnns</td>
<td>Message identifier</td>
<td></td>
</tr>
<tr>
<td>PIDS/name of UIM</td>
<td>Program name</td>
<td>PIDS/CBDUS005</td>
</tr>
<tr>
<td>VALU/Ccccccccc</td>
<td>Message variable text</td>
<td></td>
</tr>
</tbody>
</table>

Table 18. Problem Data

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wait State Code</td>
<td>D0083</td>
</tr>
<tr>
<td>Reason Code</td>
<td>01</td>
</tr>
<tr>
<td>Accompanying message</td>
<td></td>
</tr>
<tr>
<td>UIM name (if available)</td>
<td>CBDUS005</td>
</tr>
<tr>
<td>Stand-alone dump</td>
<td></td>
</tr>
<tr>
<td>IODF dump</td>
<td></td>
</tr>
</tbody>
</table>

Problems with panels and function key assignment

If problems with panels or the assignment of function keys occur, ensure the following:

- Data set SYS1.SCBDTENU must be allocated to ISPTLIB and data set SYS1.SCBDPENU to ISPPLIB.
- SYS1.SCBDCLST must be allocated to SYSPROC
- SYS1.SCBDPENU, SYS1.SCBDMENU, and SYS1.SCBDTENU must be dynamically allocated when HCD has been started. Compare with LIBDEF definitions in CBDCHCD.

If the library allocation is correct, develop a search argument, and if no problem solution is found, report the problem. To display the panel identifier, use the ISPF command PANELID. The name of the function panel will be shown in the upper left corner of the panel.

Table 19 and Table 20 show what the search argument and the problem data could look like.

Table 19. Search Argument

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIDS/CBDccccc</td>
<td>Panel identifier</td>
<td>RIDS/CBDPHW10</td>
</tr>
</tbody>
</table>

Table 20. Problem Data

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Panel identifier and name of the panel where the error was detected.</td>
<td>CBPDFW10</td>
</tr>
<tr>
<td>Type of error found.</td>
<td>Define, Modify, or View Configuration Data</td>
</tr>
</tbody>
</table>
Problems with help information provided by HCD

Messages that relate to problems with the HCD help facility have the identifiers CBDA400I to CBDA420I. Use the commands:

- HELPID to display the name of the help panel at the end of the command line. The name is displayed in the command line just before the scroll field. It can be used as search argument.
- HELPTEST to display the help panel while in help mode. It allows you to review or test any help panel while in help mode. That is, it eliminates the need to create the appropriate situation if a review or test of a help panel is required.

This command can also be used to get the help information for a message. The help member for a message consists of the message ID minus the trailing severity indicator (such as "I"). For example, the help member for message CBDA200I has the name CBDA200. Thus you may get an explanation for messages that are not yet listed in the messages documentation.

Using LookAt to look up message explanations

LookAt is an online facility that lets you look up explanations for most of the IBM® messages you encounter, as well as for some system abends and codes. Using LookAt to find information is faster than a conventional search because in most cases LookAt goes directly to the message explanation.

You can use LookAt from these locations to find IBM message explanations for z/OS elements and features, z/VM®, VSE/ESA™, and Clusters for AIX® and Linux®:

- The Internet. You can access IBM message explanations directly from the LookAt Web site at [www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/].
- Your z/OS TSO/E host system. You can install code on your z/OS systems to access IBM message explanations using LookAt from a TSO/E command line (for example: TSO/E prompt, ISPF, or z/OS UNIX® System Services).
- Your Microsoft® Windows® workstation. You can install LookAt directly from the z/OS Collection (SK3T-4269) or the z/OS and Software Products DVD Collection (SK3T-4271) and use it from the resulting Windows graphical user interface (GUI). The command prompt (also known as the DOS > command line) version can still be used from the directory in which you install the Windows version of LookAt.
- Your wireless handheld device. You can use the LookAt Mobile Edition from [www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/lookatm.html] with a handheld device that has wireless access and an Internet browser.

You can obtain code to install LookAt on your host system or Microsoft Windows workstation from:

- A CD in the z/OS Collection (SK3T-4269).
- The z/OS and Software Products DVD Collection (SK3T-4271).
- The LookAt Web site (click Download and then select the platform, release, collection, and location that suit your needs). More information is available in the LOOKAT.ME files available during the download process.

Problem with content, wording, mismatch

If problems with content, wording, or mismatches are encountered, obtain the help panel name by using the HELPID command. The help panel name is displayed at the end of the command line.
Table 21 shows what the search argument could look like.

Table 21. Search Argument

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIDS/CBDcccc</td>
<td>Help panel name</td>
<td>RIDS/CBDF403</td>
</tr>
<tr>
<td>RIDS/CBDcccccc</td>
<td>Panel identifier</td>
<td>RIDS/CBPDVF0</td>
</tr>
</tbody>
</table>

Problem shown by help messages CBDA400I or CBDA405I

If message CBDA400I or CBDA405I is issued, check:
1. The library concatenation for your HCD invocation.
2. That the help members are installed in the proper libraries (in SYS1.SCBDHENU).
3. That the help library is allocated either in LINKLST member or to ISPLLIB.

Problem shown by help messages other than CBDA400I and CBDA405I

If a message in the range CBDA400I to CBDA420I but other than CBDA400I and CBDA405I occurs, you have probably encountered a logic error in the dialog.

Table 22 and Table 23 show what the search argument and the problem data could look like.

Table 22. Search Argument

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS/ccccnnns</td>
<td>Message identifier</td>
<td>MS/CBDA404</td>
</tr>
<tr>
<td>RIDS/CBDcccc</td>
<td>Help panel name</td>
<td>RIDS/CBDF401</td>
</tr>
</tbody>
</table>

Table 23. Problem Data

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message identifier of the message that was issued. All additional information shown in the message, such as the name of the help panel or the reference phrase.</td>
<td>CBDA404</td>
</tr>
<tr>
<td>Panel identifier</td>
<td>CBDPDVF0</td>
</tr>
<tr>
<td>Help panel name</td>
<td>CBDF401</td>
</tr>
<tr>
<td>Description of the related field (in case of field help).</td>
<td>Action entry field</td>
</tr>
<tr>
<td>Name of the action choice (in case of action bar help).</td>
<td></td>
</tr>
<tr>
<td>Name of the command (in case of command help).</td>
<td></td>
</tr>
</tbody>
</table>

Problems with output of HCD textual reports

Use this procedure if you find problems that relate to the HCD report facility.

1. Check the output of the report job for messages that provide additional information.
2. For incorrect or incomplete output:
   a. Compare the contents of the IODF with the output. For example, if the device features are not shown correctly in the OS device detail report, use
the action View device definition on the I/O Device List panel to display
the definitions of the device for which the report seems to be incorrect.

b. If you find out that the definitions in the IODF are correct, but the report
output is incorrect, report this problem to IBM.

Table 24 shows what the problem data could look like.

Table 24. Problem Data

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of report that was to be created.</td>
<td>Control unit detail report</td>
</tr>
<tr>
<td>JCL that was used to create the report.</td>
<td></td>
</tr>
<tr>
<td>The content of the IODF for which the report was requested.</td>
<td>Refer to &quot;TRACE command&quot; on page 469 to create an IODF dump.</td>
</tr>
</tbody>
</table>

Problems with output of HCD graphical reports

When you have any problems printing or viewing a graphical configuration report,
read the following problem descriptions. If you have any other problem, report the
problem to IBM.

Screen shows four dots

When displaying the configuration, the screen shows only four dots.

Check that the background and foreground color you specified in the HCD profile
match. Choose a foreground color that is visible on the background color.

Incorrect DCF, GDF, or GML format in output data set

You can simply check, whether you have created the output format you have
specified in the profile using the keyword GCR_FORMAT. Browse or edit the
created data set.

BookMaster format will look like this:

```
:userdoc
.layout 1
.dr thick weight .4mm
.rh on
.sp 2
```

DCF format will look like this:

```
.df graph font X0GT20
.tr 31 AC BE BC 76 AB 30 BB 15 8F 77 CB 78 CC 80 EB 64 EC 6A FA 24 BF
.li 240mm
.dr thick weight .4mm
.rh on
.sp 2
```

GML format will look like this:

```
:gdoc
.df graph font X0GT20
.tr 31 AC BE BC 76 AB 30 BB 15 8F 77 CB 78 CC 80 EB 64 EC 6A FA 24 BF
.li 240mm
.dr thick weight .4mm
.rh on
.sp 2
```
If the output is not correct, make sure that:

- The HCD profile is allocated with ddname HCDPROF before invoking HCD.
- The profile contains the keywords GCR_FORMAT=DCF or GML and GCR_FONT with an appropriate font (for example, X0GT20 for 3820 printers).

For information on how to specify keywords in the HCD profile, refer to “Defining an HCD profile” on page 23.

**Illegible printout when using DCF or GML**
The output of a report data set contains correct DCF and GML format, but the printout is illegible.

Make sure that:

- A monospaced font (for example, X0GT20 for 3820 printers) is specified in the HCD profile using the keyword GCR_FONT.
- The specified font is installed on your printer.

**Output exceeds page boundary**
The printed output exceeds page boundary.

Make sure that:

- During printing you specified a parameter to print the report in landscape format, that is to rotate the printout by 90 degree.
- The parameter LAYOUT 1 was specified to use the full page for the report.

**Box characters are not correct**
When using DCF formatting, the box characters are not correct.

The graphical print facility uses special hex characters for the various box characters. These special characters are then translated to real box characters by means of the .tr command. If the selected font does not contain the box characters, you must either choose another font or modify the .tr command in the file generated by the graphical print facility. See the following table for information on which hex combinations HCD uses for the various box characters.

```
   .tr 31 AC BE BC 76 AB 30 BB 15 8F 77 CB 78 CC 80 EB 64 EC 6A FA 24 BF
```

where

- 31 Upper left corner
- 78 Downward T
- BE Upper right corner
- 80 Leftward T
- 76 Lower left corner
- 64 Rightward T
- 30 Upper right corner
- 6A Bar
- 15 Junction (+)
- 24 Hyphen (dash)
- 77 Upward T

[Table 25](#) shows what the problem data could look like.

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of report that was to be created.</td>
<td>LCU report</td>
</tr>
</tbody>
</table>
Problems during initialization of HCD

If a problem occurs during initialization, HCD does one of the following:

- Issues a message and continues the initialization
- Terminates the initialization

Whether the initialization of HCD continues or terminates depends on the error that is encountered as explained in the following.

Initialization continues

If a UIM service routine encounters an error during initialization, HCD works without this UIM. HCD pops up messages on the user's terminal to inform the user that messages were written to the message log.

**Note:** If an error is encountered in a UIM and if SYSUDUMP is allocated, HCD does not continue. An HCD abend '00F' is forced to provide a dump at the point where the error was detected.

Use the option **List Installed UIMs** to display the panel “Installed UIMs”. On this panel, the UIM is marked as in error. (Refer to “Query installed UIMs” on page 259.)

If you do not have access to any UIM, check if your UIMs are correctly installed. In the HCD profile you can specify the name and volume serial number of the library that contains the UIMs (see “Defining an HCD profile” on page 23). If you do not specify a name in the profile, SYS1.NUCLEUS is assumed as default names for the UIMs.

Initialization is terminated

The initialization is terminated either with an abend or with a message.

In case of an abend, the dialog:

- Pops up message CBDA040I on the user’s terminal. The message informs the user that HCD has abnormally terminated.
- Puts message CBDA050I with abend code ‘00F’ in the message log. The message also provides a reason code.
- Puts the message that is issued by a UIM service routine in the message log.

If an error is encountered in a UIM and if SYSUDUMP is allocated, an HCD abend ‘00F’ is forced to provide a dump at the point where the error was detected.

**Message CBDA041I:** Means that HCD is not able to find the UIMs. If this message is issued during initialization using the "CIT" variable, make sure that the UIMs are installed in SYS1.NUCLEUS.
Table 26 and Table 27 show what the search argument and the problem data could look like.

**Table 26. Search Argument**

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS/cccnnns</td>
<td>Message identifier</td>
<td>MS/CBDA041I</td>
</tr>
<tr>
<td>PIDS/UIM name</td>
<td>UIM name</td>
<td></td>
</tr>
</tbody>
</table>

**Table 27. Problem Data**

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>UIM name</td>
<td></td>
</tr>
<tr>
<td>Message ID(s) and full message text</td>
<td>CBDA041I</td>
</tr>
<tr>
<td>TRACE output data set</td>
<td></td>
</tr>
<tr>
<td>Refer to &quot;TRACE command&quot; on page 469</td>
<td></td>
</tr>
</tbody>
</table>

**Problems with UIMs**

For information on converting and testing UIMs, refer to [z/OS MVS Device Validation Support](#).

The following explanations apply to UIMs provided by the installation and to UIMs provided by IBM.

**Messages during initialization of HCD**

Internal logic errors in UIMs are primarily found during the initialization of HCD. Refer to "Problems during initialization of HCD" on page 456 for information on how to proceed in case of initialization problems.

**UIM problems after initialization of HCD**

Internal logic errors in UIMs may also be discovered:

- During the definition of a device (as a string like ?PARMnn? on the Device Parameter Feature panel). Message CBDA381I indicates that you may have installed a back-level UIM.
- On the Installed UIMs panel when a UIM is flagged in error. In this case messages CBDA070I or CBDA096I may be issued. Message CBDA070I means that the UIM does not match the corresponding UDT. Message CBDA096I means an unresolvable conflict between a VM and MVS UIM.

You can use the message log, together with SYSUDUMP and HCDTRACE to find the error in the UIM that failed.

If the error relates to an installation-provided UIM, make appropriate corrections. For information on converting UIMs, refer to [z/OS MVS Device Validation Support](#).

If the error relates to a UIM provided by IBM, report the problem.

Table 28 and Table 29 on page 458 show what the search argument and the problem data could look like.

**Table 28. Search Argument**

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS/cccnnns</td>
<td>Message identifier</td>
<td>CBDA070I</td>
</tr>
</tbody>
</table>
### HCD internal problems

When you have one of the following error situations, you probably have detected an internal HCD error:

- HCD displays wrong messages or does not display a message at all when you made a mistake.
- An HCD generated IOCP input data set causes error messages when using the IOCP program.

Report problems like this to IBM.

Tables 30 and 31 show what the search argument and the problem data could look like.

#### Table 30. Search Argument

<table>
<thead>
<tr>
<th>Search Argument</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS/ccnnns</td>
<td>Message identifier</td>
<td></td>
</tr>
<tr>
<td>RIDS/CBDcccc</td>
<td>Panel identifier</td>
<td>RIDS/CBDPCH30</td>
</tr>
</tbody>
</table>

#### Table 31. Problem Data

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message ID(s)</td>
<td></td>
</tr>
<tr>
<td>Panel identifier</td>
<td>CBDPCH30</td>
</tr>
<tr>
<td>Type of action the user wanted to perform</td>
<td>Change Channel path definition</td>
</tr>
<tr>
<td>Description of configuration</td>
<td></td>
</tr>
</tbody>
</table>

### Problems with 'Transmit Configuration Package' action

If a problem occurs during the Transmit configuration package action, HCD may:

- Not start the action due to authorization problems
- Not submit the job
- Submit the job but not complete it

#### Job steps of the Transmit Procedure

*Transmit configuration package* builds a batch job with multiple steps. The step names are:

**GO**

Creates an IDCAMS CLUSTER for a temporary work IODF.
ALLOCT2  Creates an IDCAMS CLUSTER for a temporary production IODF.
INIT1    Initializes the temporary work IODF.
INIT2    Initializes the temporary production IODF.
BLDPR1   The processor configurations that are contained in the
         configuration package are built into a data set as control
         statements. The processor list is specified via DD name HCDCNTL.
         If a processor contains a CFS channel path that has a connection to
         a CF partition external to the configuration package, the processor
         containing the CF partition is also included in the output data set.
MIGRPR1  The generated processor configuration control statements are
         migrated into the temporary work IODF while preserving the
         processor tokens from the master IODF.
PRINTPR1 The MESSAGES and LISTING data set are deleted if no error
         occurred.
BLDOS1   All OS configurations included in the configuration package are
         built into a data set as control statements. The OS configuration list
         is specified via DD name HCDCNTL.
MIGROS1  The generated OS configuration control statements are migrated
         into the temporary work IODF.
PRINTOS1 The MESSAGES and LISTING data set are deleted if no error
         occurred.
BLDSW1   The switch configurations of all switches containing ports that are
         connected to either a channel path or control unit of the processors
         of the configuration package are built into a data set as control
         statements. The switch list is specified via DD name HCDCNTL.
MIGRSW1  The generated switch configuration control statements are migrated
         into the temporary work IODF.
PRINTSW1 The MESSAGES and LISTING data sets are deleted if no error
         occurred.
BPROD    A temporary production IODF is built from the temporary work
         IODF.
EXPOATT  The temporary production IODF is exported attended to the
         specified user/node ID.
EXPOUATT The production IODF is exported unattended to the specified
         system.
DEL1     The temporary work IODF is deleted.
DEL2     The temporary production IODF is deleted.

Note: Stepname GO is used for HCDDECK, HCDMLOG, HCDLIB, HCDTRACE,
       and HCDPROF. Thus, the data sets are made available to the steps which
       require them.

Temporary data sets created by the Transmit Procedure
The transmit procedure creates the following data sets:
• hlq.IODFx.xzzz (production IODF)
• hlq.IODFx.xMIT.package.WORK (work IODF)
• hlq.IODFx.xMIT.package.DECK (configuration decks)
• hlq.IODFx.xMIT.package.MSGLOG (HCDMLOG)
• hlq.IODFx.xMIT.package.sss.MESSAGES (HCDPRINT migration messages)
• hlq.IODFx.xMIT.package.sss.LISTING (HCDASMP migration listing)

where:
hlq          is the high level qualifier specified on the transmit panel or the HLQ
             parameter of the batch utility.
xx            is the suffix of the target IODF name specified with the package
package      is the name of the configuration package to be transmitted
zzzz are the qualifiers 3-n of the target IODF name

ss

qualify the migration type (PR1 for processor, OS1 for operating systems, SW1 for switch configurations)

After a successfully completed transmit action all these data sets, except the message log file, are deleted. The message log file is preserved until it is overwritten, when another transmit action using the same package name and IODF suffix is performed.

Apart from the production IODF, all redundant data sets remaining from a cancelled transmit action are identified by their common data set name qualifiers. hlq.IODFxx.XMIT.package.

Authorization problems

Because the last sent date of the IODF from which the transmit action is performed is updated with the current date, you require write access to the accessed production IODF. Otherwise message CBDG247I is displayed.

You also need permission to write to the data sets with the qualifiers of the IODF to be created and transmitted.

Job is not submitted

If the work IODF or production IODF to be created temporarily exists already, it is not possible to start the transmit action. This may happen when a previous transmit job was cancelled, ended with an error or another transmit job is running which uses the same high level qualifier and target IODF name.

Depending on the source of the problem this may be resolved by deleting the existing temporary IODFs or by specifying a different high level qualifier for the target IODF.

Job is not completed

The transmit action generates a batch job. Check the HCD message log file to find out if the job was executed. It is shown as a sequence of HCD batch job steps ending with a successful export message. There are several possible causes if this is not the case:

- JCL errors: check the job output. To see all statements including the inline statements which are generated by the HCD dialog and submitted, issue
  `TRACE ON ID=JCL`
  and perform the transmit action. For an example of a trace refer to “Customization unsuccessful” on page 461.

- If the HCD message log file shows that a particular step failed, check the job output for potential allocation problems.

- If one of the migration steps failed, check the LISTING and MESSAGES data sets. (Refer to “Temporary data sets created by the Transmit Procedure” on page 459 for more information on the data sets created during the action.) Ensure that the same versions of UIMs are available for the Transmit configuration package action as for creation of the IODF.

  The MESSAGES and LISTING data sets, as well as the generated decks, are deleted if no errors occurred. If you want to keep them, you can modify the conditional statements in procedure CBDJXMIT for the migrate steps. Do this by copying CBDJXMIT to a new procedure.

  Proceed as follows, if you need to trace particular steps:
1. Define a profile including a TRACE statement.
2. Specify the stepname.HCDPROF DD with the profile name.
3. Allocate a trace data set name.
4. Specify GO.HCDTRACE DD with the name of the trace data set in order to use it for all steps to be traced or stepname.HCDTRACE to use the trace data set only for the single step.
5. To specify HCDDECK, HCDMLOG, HCDLIB, HCDTRACE, or HCDPROF use GO as the step name. The other steps refer to the definitions in the GO step. If you want to preserve a specific output data set, pre-allocate it to HCDDECK (see "Build I/O configuration data" on page 320).

**Customization unsuccessful**

This section describes points to be considered when customizing the transmit procedure.

The transmit procedure exploits the migration batch utility, which uses parsing macro CBPDZPARS (residing in SYS1.MACLIB). If you want to use a different macro library, specify this as GO.HCDLIB.

The dialog always generates and submits the following statements:
- All parameters for procedure CBDJXMIT
- The JOB card, JOBLIB and overwrite statements given by the user
- IDCAMS DEFINE CLUSTER and DELETE CLUSTER statements in steps GO, ALLOCT2, DEL1 and DEL2
- An HCDDECK DD statement for at least one of BLDPR1, BLDS1 or BLDSW1 job steps, dependent on the package content
- The SYSTSIN for EXPOATT or EXPOUATT, depending on whether attended or unattended export is selected

The following is a sample trace showing the batch job built by a transmit action.

```
09:53:01  97-11-04  Trace started.
//XMIT  JOB (3243),'OS390H1',MSGCLASS=X,CLASS=A,REGION=4M
//JOBLIB DD DSN=SYS1.SCBHENU,DISP=SHR
//XMT0 EXEC PROC=CBDJXMIT,PR=1,OS=1,SW=1,
// ATTEND=1,RECORDS='1684',
// DESC1='IODFST',DESC2='IODF88',
// QUALX='IODF88.XMIT.CBBB',
// IODFTP='05390H1.IODF88',
// IODFTW='05390H1.IODF88.XMIT.CBBB.WORK',
// IODFSP='100FST.100F11.MASTER'
//GO.SYSIN DD *
/*
   DEFINE CLUSTER(   
      NAME ( 05390H1.IODF88.XMIT.CBBB.WORK.CLUSTER) - LINEAR - 
      RECORDS (1684) - 
      VOLUMES (DS7001) - 
      DATA(NAME(05390H1.IODF88.XMIT.CBBB.WORK)) 
   ) */
//ALLOCT2.SYSIN DD *
/*
   DEFINE CLUSTER(   
      NAME ( 05390H1.IODF88.CLUSTER) - LINEAR - 
      RECORDS (1684) - 
      VOLUMES (DS7001) - 
      DATA(NAME(05390H1.IODF88)) 
   ) */
//BLDSPR1.HCDDECK DD *
```
CB88
CF14, CF
/*
//BLDOS1.HCDCNTL DD *
B710
/*
//BLDSW1.HCDCNTL DD *
71
72
74
77
/*
//EXP0ATT.SYTSIN DD *
CALL 'SYS1.LINKLIB(CBDMGHCP)', +
'EXPORT,OS390H1,PKSTCB88'
/*
//DEL1.SYTSIN DD *
DELETE OS390H1.IODF88.XMIT.CB88.WORK.CLUSTER
/*
//DEL2.SYTSIN DD *
DELETE OS390H1.IODF88.CLUSTER
/*
09:53:29 97-11-04 Trace stopped.

Table 32 shows what the problem data could look like.

<table>
<thead>
<tr>
<th>Problem Data</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job output</td>
<td>See example trace shown in section &quot;Customization unsuccessful&quot; on page 461.</td>
</tr>
<tr>
<td>Message log file</td>
<td></td>
</tr>
<tr>
<td>Trace of failing step</td>
<td></td>
</tr>
<tr>
<td>Deck for failing step</td>
<td></td>
</tr>
<tr>
<td>LISTING data set</td>
<td>See &quot;Job steps of the Transmit Procedure&quot; on page 458</td>
</tr>
<tr>
<td>MESSAGES data set</td>
<td>See &quot;Temporary data sets created by the Transmit Procedure&quot; on page 459</td>
</tr>
<tr>
<td>Submitted job (via TRACE ID=JCL)</td>
<td></td>
</tr>
<tr>
<td>Procedure used (if modified)</td>
<td></td>
</tr>
</tbody>
</table>

**LDAP problem determination**

In addition to the regular text messages from the HCD LDAP backend, debug output can be requested. This debug output is switched on or off by specifying a debug level for the IBM Tivoli Directory Server for z/OS (see [IBM Tivoli Directory Server Administration and Use for z/OS](https://www.ibm.com)) and is printed to the same location as the HCD LDAP backend text messages. HCD LDAP backend’s debug output is only available in English language and character representation IBM-1047.

The HCD LDAP backend mainly supports two debug levels: LDAP_DEBUG_ERROR and LDAP_DEBUG_TRACE.
LDAP_DEBUG_ERROR causes all the information concerning errors, detected while performing operations, to be printed. From the HCD LDAP backend’s point of view, LDAP_DEBUG_ERROR can be switched on regularly without significant performance loss.

LDAP_DEBUG_TRACE causes all important program operations to be printed. From the HCD LDAP backend’s point of view, LDAP_DEBUG_TRACE should only be used when reproducing error situations for problem determination.

If you have general problems getting the IBM Tivoli Directory Server for z/OS, RACF Backend, or any other plug-ins running, leave out all HCD LDAP backend definitions from the started task and configuration files (ds.conf). If the server and plug-ins still will not run, contact the IBM service.

If everything functions correctly without the HCD LDAP backend and abends with the HCD LDAP backend, then request a dump and check there for further information.

If the dump indicates that functions in DLL GLDSLP31 or GLDCLDAP are missing, then maybe the IBM Tivoli Directory Server for z/OS functionality is not available.

**HCD LDAP backend abnormal termination**

There are two different locations at which the HCD LDAP backend might terminate abnormally:

- The IBM Tivoli Directory Server for z/OS address space
- The HCD instance address space

The way to find out more information about the abnormal termination depends on in which of these locations it terminated see ["IBM Tivoli Directory Server for z/OS address space"](http://www.ibm.com/support/docview.wss?uid=swg21238680) or ["HCD instance address space"](http://www.ibm.com/support/docview.wss?uid=swg21238680) on page 464.

In general, whenever the HCD LDAP backend or parts of it terminate abnormally, some global resources may remain in memory. These resources are so called POSIX message queues and are used for the Inter Process Communication (IPC) between the HCD LDAP backend and the HCD instances. They have to be removed from the system manually.

To remove POSIX message queues from the system proceed as follows:

1. Look into the IBM Tivoli Directory Server for z/OS started task log to find the identifiers (ID’s) of all message queues allocated: The HCD LDAP backend prints out all ID’s using the message C8000041. Note that you have to find all of these C8000041 messages.
2. Use the UNIX System Services command line tool ipcs to list all the POSIX message queues in memory.
3. Find those identifiers in the list which correspond to POSIX message queues belonging to the abnormally terminated HCD LDAP backend.
4. Remove the appropriate message queues using UNIX System Service command line tool ipcrm.

**IBM Tivoli Directory Server for z/OS address space**: In general, diagnostic output can be found in the IBM Tivoli Directory Server for z/OS started task log. Investigate this log to find information, about the system or user abend code, the reason code, and additional messages and follow the instructions in ["HCD abnormal termination"](http://www.ibm.com/support/docview.wss?uid=swg21238680) on page 448.
For more information on problem determination, refer to z/OS Problem Management.

Read the information on how to obtain a dump of the abnormally terminated program and determine whether the problem was caused by the HCD LDAP backend or by any other program in the IBM Tivoli Directory Server for z/OS address space.

**HCD instance address space:** Since an HCD instance resides in a different address space (created with a call to spawn()) to that of the corresponding IBM Tivoli Directory Server for z/OS, it is not permitted to print its output to the IBM Tivoli Directory Server for z/OS started task log. Instead, an HCD instance creates an HFS file containing diagnostic output. The name and location of this file follows the conventions described in the publication z/OS Language Environment Debugging Guide in the unit "Using Language Environment Debugging Facilities". Note that the environment variable _CEE_DMPTARG is set to /tmp by default.

Example:
Assume that a language environment dump of the HCD instance with the process ID 197 is taken at 05:55:01 pm on September 18, 1998. The file containing the dump will be called CEEDUMP.19980918.175501.197 and it will be located in the HFS directory /tmp.

**Adverse LDAP return code from the HCD LDAP Backend**

Each LDAP client's request is answered by the IBM Tivoli Directory Server for z/OS with an LDAP response containing a return code for the request. Such a return code can be either good (LDAP_SUCCESS) or bad (e.g. LDAP_UNWILLING_TO_PERFORM). These numeric return codes can be transformed into text messages (e.g. "DSA is unwilling to perform.") using the LDAP client API.

Return codes such as LDAP_OPERATIONS_ERROR, LDAP_NO_MEMORY, or LDAP_LOCAL_ERROR typically indicate a problem within the IBM Tivoli Directory Server for z/OS, the HCD LDAP backend, or the LDAP client. If you assume the problem to be located in the IBM Tivoli Directory Server for z/OS or the HCD LDAP backend, look into the IBM Tivoli Directory Server for z/OS started task log and search for error messages.

Return codes such as LDAP_INAPPROPRIATE_MATCHING, LDAP_INVALID_DN_SYNTAX, or LDAP_UNWILLING_TO_PERFORM typically indicate an incorrect LDAP request. Check if your request conforms with the appropriate constraints. You may also check the IBM Tivoli Directory Server for z/OS started task log for error messages.

**Error in the HCD LDAP backend**

If the HCD LDAP backend's message CBD0009E is printed to the IBM Tivoli Directory Server for z/OS started task log, an error situation was encountered inside the HCD LDAP backend while performing an operation. Switch on IBM Tivoli Directory Server for z/OS debug level LDAP_DEBUG_ERROR and reproduce the problem. The log will contain more information concerning the reason of the error. You may additionally specify debug level LDAP_DEBUG_TRACE to get detailed informations about the operations performed by the HCD LDAP backend.

The following illustrates two typical error situations with advice on how to check for these problems.

**HCD Instance Startup Fails:** If an HCD instance cannot be started correctly, this will not lead to error messages in the IBM Tivoli Directory Server for z/OS started task log or to bad return codes for the LDAP client in all cases.
Consider the following possible scenario: The IBM Tivoli Directory Server for z/OS is started together with the HCD LDAP backend which, in turn, launches at least one HCD instance. All launched HCD instances fail to start. As the HCD LDAP backend has no indication that the HCD instances failed to start, it will wait forever for an acknowledgement from the HCD instances. Incoming requests will be scheduled for an HCD instance and will be blocked up until an acknowledgement arrives. In this way, the requests will never be performed and no LDAP response will be sent to the requesting LDAP client.

From the LDAP client’s point of view: If the very first request for the HCD LDAP backend is pending for a long time, the HCD instances might have failed to start up. In general, if one HCD instance fails to start up, all HCD instances will fail.

To check whether an HCD instance failed to start, look in the IBM Tivoli Directory Server for z/OS started task log: HCD LDAP backend’s message CBD0007I will indicate how many HCD instances will be started. If all of these HCD instances are able to start up correctly, you will find an appropriate number of CBD1002I messages. If you cannot find them it’s most probable that the HCD instances failed to start up.

**HCD instance user ID switch fails:** If you have set up the IBM Tivoli Directory Server for z/OS or the HCD LDAP backend incorrectly, the HCD instances might have problems switching their User ID appropriately. It might be possible that the User ID switch fails completely, or only for one or more specific User IDs.

If you assume that LDAP requests can’t be performed because of a failed User ID switch, you can check your assumption as follows: Search for the CBD0009E message in the IBM Tivoli Directory Server for z/OS started task log. If you can find it switch on IBM Tivoli Directory Server for z/OS debug level LDAP_DEBUG_ERROR and search for message EDC5139I. This message indicates that the User ID switch has failed.

Finally, check the setup of your HCD LDAP backend.

---

**Diagnostic information and tools**

The information and tools described in this section help you to diagnose system problems.

**HCD messages**

In case of an error, HCD issues messages. Depending on what you are currently doing, the messages are written:

- To the terminal as a single message
- To the terminal in a message list
- In a message log
- In a migration log
- In the output of a batch job

**Terminal messages**

User-errors, such as erroneous syntax entry and contextually wrong definitions, are handled by the dialog at the time of data entry. That is, the dialog displays messages at the terminal and the user can take corrective action immediately.

Some operations produce multiple messages. In this case, HCD displays a message list. You can save the displayed messages from the message list into the message log. See "Message lists" on page 66 on how to work with message lists.
Message log
Errors that are of low interest for the end user, such as incomplete UIMs during initialization, are only written to the message log. The user will be informed about this when leaving the dialog or switching to another IODF. Only in critical situations (for example, when the message log is not available), will the messages be written into the ISPF list data set. If this, however, also fails, the message will be written into the operating system log.

To see a message in the message log, issue the SHOWMSG command or use the View message log pull-down choice from the Query action bar on any action list panel.

Migration log
HCD maintains a migration log that contains messages issued by the migration process. You can view this migration log through ISPF.

Figure 191 shows where you can find messages while working with HCD.

<table>
<thead>
<tr>
<th>Mode of Operation</th>
<th>You Find the Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dialog Mode</td>
<td>On the terminal In the message log</td>
</tr>
<tr>
<td>Batch mode</td>
<td>In the message log, that is the data set allocated with ddbname HCDMLOG In the output (SYSPRINT) of the batch job</td>
</tr>
<tr>
<td>Migration of input data sets</td>
<td>In the migration log, that is the data set allocated with ddbname HCDPRINT</td>
</tr>
<tr>
<td>IPL</td>
<td>Trapped by IPL. A wait state code is issued.</td>
</tr>
<tr>
<td>HCD LDAP backend</td>
<td>In the LDAP response, which the IBM Tivoli Directory Server for z/OS sends to the LDAP client. In the IBM Tivoli Directory Server for z/OS started task’s log.</td>
</tr>
</tbody>
</table>

Note: ¹ The output shows the message number and text; for explanations refer to z/OS and z/VM HCD Messages

Figure 191. Where to find HCD messages

LDAP response messages
If HCD is used as part of the HCD LDAP backend, HCD will issue messages if problems occur. These messages can be made visible in the following way:
In general, LDAP clients send their requests to the IBM Tivoli Directory Server for z/OS and get an LDAP response for each request. If an LDAP client requests an operation from HCD via the HCD LDAP backend and HCD encounters any problems while working on the request, HCD issues one or more text messages. These messages are sent back to the LDAP client as a part of the LDAP response and can be extracted using the LDAP client API.

The command line utilities (e.g. ldapsearch or ldapmodify) which are part of the IBM Tivoli Directory Server for z/OS show the messages contained in the LDAP response as “additional info”.

IBM Tivoli Directory Server for z/OS started task log
The HCD LDAP backend runs as a part of the IBM Tivoli Directory Server for z/OS and all HCD LDAP backend messages are printed to the same location as the IBM Tivoli Directory Server for z/OS messages. The IBM Tivoli Directory...
Server for z/OS runs as started task for the HCD LDAP backend and the messages from the server and all of its backends are printed to the IBM Tivoli Directory Server for z/OS started task log.

The HCD LDAP backend and its HCD instances print some informational and error messages to the IBM Tivoli Directory Server for z/OS started task log. If the LDAP server debug is switched on (i.e. debug level is not LDAP_DEBUG_OFF), all diagnostic output is also printed to the started task log. The regular HCD LDAP backend messages and the additional debug output can be used to verify that the HCD LDAP backend is working correctly.

**HCD trace facility**

The output of the HCD trace facility provides information to locate internal HCD problems. It helps the IBM program system representative to identify the cause of a failure.

**Data Set**

The trace records generated by HCD are recorded in the trace data set.

The trace data set must be preallocated with a ddname of HCDTRACE. If the trace data set is not allocated when HCD is invoked, no tracing takes place. The default CLIST that is provided with HCD allocates a trace data set with the name HCD.TRACE, prefixed by your user ID.

**Trace records**

The trace records show the control flow within the various HCD modules. Trace information is written into the trace data set:

- Whenever a module (CSECT) gets control.
  
  In this case, the passed parameter list is recorded together with the name and description of the invoked module.

- Whenever a module (CSECT) returns to its calling routine.
  
  The passed parameter list containing the return and reason codes is recorded.

When HCD is invoked, HCD overwrites the existing trace with the new trace data. If you want to retain the existing data, you have to make sure that the data is saved.

Figure 192 on page 468 is an example of a trace output. The following explains the records you see:

```
17:00:13 96-08-01 Trace started
```

The time stamp shows when the trace facility was started. This record is useful to identify the trace data sets when multiple traces are produced on the same day.

```
PUSH CBDMSMSG - Message Routine 96214 HCS6031 17001352 08472028
```

This line indicates that control has been passed to another module at a certain time. In this example, the module named CBDMSMSG with service level ‘96214 HCS6031’ received control at 17:00:13.52 at storage address X‘08472028’. The records also give a short description of what the module does, and the parameter list that is passed to the called module.

```
CBDMSMSG Message destination: Screen
```

This trace entry is written by the called module.

```
POP CBDMSMSG - Message Routine 17001353
```

Appendix C. Problem determination for HCD 467
Indicates that control from the module named CBDMSMSG is returned to the calling routine at 17:00:13.53. Also, the passed parameter list is shown again, but now the parameter list contains the return/reason code indicating how successful the requested function was.

The last entry in the trace output is a time stamp that indicates end of tracing.

Figure 193 on page 469 is an extract of a trace output when an abend occurred. The following explains the record in the example.

**ESTAE:** The entries show information that was recorded by the HCD ESTAE routine. Entries under *Diagnostic stack* list the modules that were executing when the abnormal termination occurred, together with the service levels. The first entry names the module that ended abnormally; in this example, the module is CBDMDPK5. The entries also show the control flow between the various HCD modules. In this example:

- Module CBDMDPK0 called module CBDMDPK5.
- Module CBDMGDIA called module CBDMDPK0.
- Module CBDMGHCP called module CBDMGDIA.
- and so on.
Activating the trace

The trace can be activated either:

- Dynamically by using the TRACE command in the HCD dialog. The command is optional, and can be entered whenever a displayed panel has a command line. For information on the TRACE command, refer to "TRACE command."

- By invoking HCD (for details see Chapter 12, "How to invoke HCD batch utility functions," on page 307) with the TRACE option specified in the passed parameter string (this is done automatically by the entry in the CLIST).

   //BWINJOB JOB (3259,RZ-28),BWIN,NOTIFY=BWIN,CLASS=A,
   // MSGCLASS=Q,MSGLEVEL=(1,1),REGION=4M
   //REPORT1 EXEC PGM=CBDMGHCP,
   // PARM='TRACE,REPORT,CSMEN,PROC1,PART1,MVS1,00'
   //HCDTRACE DD DSN=BWIN.HCD.TRACE,DISP=OLD

- By specifying the TRACE command in the HCD profile, for example,

   TRACE ON RESET,HOM,V,R LEVEL=255

The tracing stays active until either turned off by the TRACE command, or until HCD terminates.

TRACE command

The TRACE command activates and deactivates the HCD trace facility. The command allows you also to limit the detail of data written into the trace data set by requesting that only certain functions and details should be traced.

The TRACE command can be entered on any HCD panel showing a command line. The command can also be specified in the HCD profile data set. It is, however, not shown in the HCD Profile Options dialog.

For HCD to write the output to the trace data set, ON must be specified with at least one trace category (or you must have invoked HCD with the TRACE
To view the trace output, you have to close the trace data set first. You can do this by either leaving HCD or by entering the command TRACE OFF,CLOSE.

The format of the command is as follows:

```
TRACE Command

TRACE ON trace-category

trace-category:
  ALL
  Batch
  Command
  Dialog
  DYName
  HOM
  Migration
  Repository
  RepService
  RESET
  Service
  UIM
  UIMService
  Utility
  Validation
  Other

ON: Starts the trace facility.
OFF: Stops the trace facility.
CLOSE: Closes the trace data set.

trace category: Specifies the functional scope to be traced:
  ALL: Trace everything.
  Batch: Trace all batch routine.
  Command: Trace all command routines.
  Dialog: Trace all dialog routines.
  DYName: Trace all dynamic routines.
  HOM: Trace all object management routines.
  Migration: Trace all migration routines.
  Repository: Trace all repository main routines.
  RepService: Trace all repository service routines.
  RESET: Reset all currently active categories, LEVEL and ID.
```

Note: You may abbreviate some of the keywords. The characters you have to use are indicated by uppercase (you must then omit lowercase). For example RepService may be abbreviated as RS.
**Service**
Trace all service routines.

**UIM**
Trace all UIM routines.

**UIMSernoise**
Trace all UIM service routines.

**Utility**
Trace all utility routines.

**Validation**
Trace all validation routines.

**Other**
Trace all other not yet mentioned routines.

**LEVEL=n**
Assigns a level of detail to the functions to be traced, where
\( n \) is a decimal number ranging from 0 to 255. If the option is
omitted, the default level of 5 is assumed. The TRACE
option described in “Input parameter string” on page 308 is
equivalent to the command `TRACE ON,ALL,LEVEL=255`.

**ID=IODF**
Writes an IODF dump into the trace data set. This parameter
cannot be specified in the HCD profile. If you have a
consistent IODF, an output in the trace data set is only
shown when you set `LEVEL=128` or higher. Otherwise, an
output is only shown if the IODF contains defects.

**REPAIR**
Removes detected errors in the work IODF and
reports corrections in the trace data set. Before you
use the REPAIR option, you must set the work IODF
in update mode.

**ID=JCL**
Writes into the trace data set all statements generated when
action `Transmit configuration package` is invoked from the
HCD dialog.

**ID=IOOPSOUT**
Writes all responses of I/O Operations IHVAPI2 calls into
the trace data set. These are the results of I/O Operations
query requests.

**ID=CLOG**
Writes the contents of the change log file into the HCD trace
data set. You should use this option together with `LEVEL=8`.
This parameter cannot be specified in the HCD profile.

---

**Trace command via HCD profile**

- If HCD is started with the TRACE keyword, (for example, either started via
HCM with the HCD Trace box selected in the HCM login dialog, or via the IBM
Tivoli Directory Server for z/OS configuration file), initially all the trace
categories will be traced. After the HCD profile has been read, however, the
TRACE parameters there may modify the TRACE behavior.

- The TRACE parameters set in the HCD profile will also influence the TRACE
behavior if you activate the tracing in HCM at a later time.

- The TRACE parameters set in the HCD profile will determine the contents of the
trace for the rest of the session.
  - If you use the RESET option followed by trace categories `cat1,cat2,...,catn`, then
  only the categories `cat1,cat2,...,catn` will be considered. The RESET option must
  be the first option because the trace categories are additive (LEVEL is set to
  0).
  - Specifying the keyword off will terminate the startup trace.
  - Specifying the keyword on will start the HCD trace (if not already started)
  and will invoke the trace parameters of the TRACE statement.
  - Specifying `LEVEL=n` will set the level of trace detail. If the LEVEL parameter is
  not set, then the trace will use the default level of 5.

- If no categories are set explicitly, then all trace categories will be active.
**IODF dump**

Use the ID=IODF parameter of the TRACE command to produce an IODF dump. This command goes through your IODF, checks it for corrupted data, and writes all records and defects into the trace data set. If you have a consistent IODF, you must set the LEVEL parameter to LEVEL=128 or higher to get an output. Otherwise, an output is only shown if the IODF contains defects.

If your IODF has defects, error message CBDA999I ‘Defect(s) detected in IODF xxx’ is displayed, and message CBDA099I is written into the message log data set. In addition, the trace data set records defects with the string ‘Error:’ followed by the reason. You can locate the reported defects by searching to that string in the trace data set.

If no defects are detected in the IODF, message CBDA126I ‘TRACE command was accepted’ is given.

If you cannot invoke HCD, and therefore, cannot use the TRACE command any longer, use a JCL stream for producing a dump. Figure 194 is an example of the JCL stream for producing an IODF dump.

Make changes to the entries according to your installation requirements.

```
//WAS$IODF JOB '3259,BOX01,S=C', 'SMITH', MSGLEVEL=(1,1),
  NOTIFY=WAS,CLASS=A,MSGCLASS=Q,REGION=4096K
//PERTVSAM EXEC PGM=IDCAMS
//INPUT DD DSNAME=WAS.IODF02.WORK,DISP=SHR
//OUTPUT DD DSNAME=WAS.IODF02.DUMP,UNIT=SYSALLDA,
  // SPACE=(CYL,(20,10)),
  // DCB=(LRECL=125,BLKSIZE=629,RECFM=VBA),
  // DISP=(NEW,CATLG)
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  PRINT -
    INFILE(INPUT) -
    DUMP -
    OUTFILE(OUTPUT)
/*
*/
```

Figure 194. Example: JCL stream for producing an IODF dump

**Repair an IODF**

If your IODF contains defects, some of them can be repaired with the REPAIR option of the TRACE ID=IODF command.

First, your work IODF must be set in update mode to correct defects. You can do this, for example, by changing a description field in your work IODF. A production IODF cannot be repaired.

To repair defects in your work IODF, add the REPAIR option to the TRACE ID=IODF command: TRACE ON,REPAIR,ID=IODF.

Each corrected defect is recorded in the trace data set with the string ‘Defect has been corrected’.

If a defect has been repaired, message CBDA998I, ‘Defect(s) detected in IODF xxx. Repair action performed.’ is issued. Repeat the TRACE ID=IODF command to check whether all defects could be corrected.
An IODF that has been enabled for multi-user access cannot be repaired. You first must disable it for multi-user access before it can be set into update mode and be repaired.

**MVS dumps and traces**

To aid in diagnosing problems, z/OS automatically provides messages and error records, and on request dumps and traces. HCD uses those services to record errors. For information about:

- Dumps and traces, refer to [z/OS MVS Diagnosis: Tools and Service Aids](#).
- Using the diagnostic information, refer to [z/OS Problem Management](#).

**IPCS reports**

z/OS allows you to format dumps into diagnostic reports. To produce the reports, use the Interactive Problem Control System (IPCS).

For information, refer to:

- [z/OS MVS IPCS User’s Guide](#)
- [z/OS MVS IPCS Commands](#)
- [z/OS MVS IPCS Customization](#)

**Searching problem reporting data bases and reporting problems**

Search arguments are used to search problem reporting data bases. If the problem being diagnosed was already reported and the symptoms entered into the data base, the search will produce a match.

To perform a search, do the following:

1. Analyze the problem reporting data base and develop a search argument using the information provided in the boxes labeled Search Argument.
2. Complete the digits (such as ccc, nnn, hhh) according to the applicable conditions. For example, if the message CBDA099I was received, the developed search argument for message identifier would be: MS/CBDA099I. An example is shown in Table 13 on page 449.
3. Use the search arguments to search problem reporting data bases. If the search finds that the problem has been reported before, request a fix from IBM.
   
   If the search is unsuccessful, report the problem to the IBM Support Center. Submit the information that is listed in the Problem Data tables. An example is shown in Table 14 on page 449.

For more detailed information on these steps, refer to [z/OS Problem Management](#).

**Sending an IODF to a different location**

There may be situations, in which an IODF is to be transferred to a different location or system. Usually, the HCD Export/Import Utility can be used to transmit the IODF to the desired target destination. Sometimes, however, there are situations, where this may not be possible (e.g. a direct connection does not exist). The following steps describe a simple method in which you can transfer your IODF data from one z/OS host to any other z/OS host even if a direct transmission path not available.

1. Examine the size of the IODF, which is to be transferred. Check for the number of allocated 4K blocks in the HCD dialog: Maintain I/O definition files ---> View I/O definition file information.
2. Use the HCD Export function, found under the HCD dialog Define, modify, or view configuration data, Export I/O definition file in the HCD dialog. Send the IODF to your own user ID; that is, to the user ID of the host on which you are currently working. HCD will export the IODF data as a sequential data set to your own user ID. If you specify an asterisk (*) for the target user ID and target node ID, the sequential data set is not transmitted but rather written directly to the data set user.EXPORTED.IODFnn.xxxx. In this case, you can skip step 4.

3. Exit the HCD dialog.

4. Use the TSO RECEIVE command to retrieve the IODF data from your internal reader. Per default, you will get a sequential data set user.EXPORTED.IODFnn.xxxx. This data set has the record organization FB and LRECL=BLKSIZE=4096 and the number of blocks as mentioned under step 1 on page 473.

5. Download this data set to your workstation. It is important that you ensure that the download is in binary mode.

6. To save storage resources, you may consider compressing (zipping) the downloaded file on the workstation.

7. Now you can transfer the IODF to a different workstation/location. Once the IODF data has arrived at the target workstation, you have to decompress (unzip) the file if it has been compressed for transportation.

8. On the target z/OS host pre-allocate a data set into which the IODF data is to be uploaded. It must be a sequential data set with FB, BLKSIZE=LRECL=4096 and the number of blocks must be the number of allocated blocks of the original IODF (see step 1 on page 473).

9. Upload the IODF data from the workstation to the z/OS host in binary mode into the pre-allocated sequential data set.

10. Use the HCD Import function, found under the HCD dialog Define, modify, or view configuration data, Import I/O definition file and specify the sequential data set to be imported into an IODF data set using the name of your choice.
Appendix D. HCD object management services

Programming Interface information

The HCD object management services (HOM) provide an application programming interface for retrieving data from the IODF, such as switch data, device type, or control unit type. The programs requesting the services cannot run in APF-authorized state.

The mapping macros CBDZHRB, CBDZHIEX, CBDZHOEX, and CBDZHCEX (see “Data input and output areas” on page 476, and “Request block (HRB)” on page 476) are not available as source code. The macros are listed in z/OS MVS Data Areas, Vol 1 (ABEP-DALT), and must be coded by the application writer.

How to invoke the HOM services

Programs can invoke the services from the HCD routine CBDMGHOM. An application issuing a request must have its own copy of the CBDMGHOM routine dynamically loaded or linked. For every request, the application has to pass the parameters shown in Table 33 using standard linkage conventions.

Table 33. Used registers and passed parameters

<table>
<thead>
<tr>
<th>Register</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Undefined</td>
</tr>
<tr>
<td>1</td>
<td>Address of five-word parameter list:</td>
</tr>
<tr>
<td>2</td>
<td>Address of request control block (HRB)  4-byte field containing the address of the request block. The request block contains the function, the object to which the function is applied, and qualifiers, attributes, and parameters. See “Request block (HRB)” on page 476 for more details.</td>
</tr>
<tr>
<td>3</td>
<td>Address of (pointer to input data or zero)  4-byte field containing the address of the address of the data input block if the request requires input. It is required on a HRB_SETUP request. See “Data input and output areas” on page 476 for more details.</td>
</tr>
<tr>
<td>4</td>
<td>Address of (length of input data or zero)  4-byte field containing the address of the fullword fixed binary integer containing the length of the input data. It must correspond to the exact length of the data contained in the data-input block, that is, no trailing or intermediate blanks are allowed.</td>
</tr>
<tr>
<td>5</td>
<td>Address of (pointer to output data or zero)  4-byte field containing the address of the address of the data output block if the request returned output. It is required on HRB_DGET and HRB_MGET requests to obtain the data and messages from the API. See “Data input and output areas” on page 476 for more details.</td>
</tr>
<tr>
<td>6</td>
<td>Address of (length of output data or zero)  4-byte field containing the address of the fullword fixed binary integer containing the length of the output data.</td>
</tr>
</tbody>
</table>

The parameters must be coded in the order shown. Only the first parameter (address of request block) is mandatory. The others are optional and depend on the type of request, as shown in Figure 195 on page 479. If you omit an optional parameter, you must specify a zero instead.
Table 33. Used registers and passed parameters (continued)

<table>
<thead>
<tr>
<th>Register</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-12</td>
<td>Undefined</td>
</tr>
<tr>
<td>13</td>
<td>Address of 18 word save area</td>
</tr>
<tr>
<td>14</td>
<td>Return address</td>
</tr>
<tr>
<td>15</td>
<td>Entry point address</td>
</tr>
</tbody>
</table>

Note: The service supports calls for both 24-bit and 31-bit addressing mode.

Data input and output areas

Data input and output areas must be contiguous areas of main storage allocated by the application in private storage and freed later on.

The areas have no header section, that is, the data starts at the first byte of the area and continues without gaps. The data contained in these areas are the interface records, which are described in the mapping macros CBDZHOEX and CBDZHIEX in \textit{z/OS MVS Data Areas, Vol 1 (ABEP-DALT)}.

For the GET request, the output area might contain the definition of multiple objects on return, whereas the input area normally contains only one object.

Issue a HRB_DGET function to get the provided output data. The size of the output data is returned by the previous GET request. The application is responsible to allocate the correct output size. If the data does not fit into the size allocated by the application for the output area, the data will be truncated.

Request block (HRB)

The HOM request block (HRB) you have to set up is described fully as mapping macro CBDZHRB in \textit{z/OS MVS Data Areas, Vol 1 (ABEP-DALT)} \[Table 34\] summarizes the request block names and constants you can specify for the functions shown in Figure 195 on page 479.

On input, this block contains the detailed request to the HOM services. On output, it contains the data requested, messages, return codes, and reason codes.

The field HRB_OBJECT with all its subfields describes the object that should be processed on the request to the API.

The object code HRB_OBJ_CODE must be coded for every request, because it identifies the class of objects that are subject of the actual request.

The constants and flags required to describe the objects are contained in CBDZHCEX, which is documented in \textit{z/OS MVS Data Areas, Vol 1 (ABEP-DALT)}.

Table 34. Summary of Request Block Names and Related Constants

<table>
<thead>
<tr>
<th>Name</th>
<th>Constants</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HRB_SDESC</td>
<td>HRB_SDESC_C</td>
<td>Request block storage descriptor. Required for all requests.</td>
</tr>
<tr>
<td>HRB_LENGTH</td>
<td></td>
<td>Length of the request block HRB.</td>
</tr>
<tr>
<td>HRB_USE_IODF</td>
<td></td>
<td>Name of the IODF to be used for the request.</td>
</tr>
<tr>
<td>Name</td>
<td>Constants</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>HRB_FUNCTION</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HRB_SETUP</td>
<td>Function code</td>
</tr>
<tr>
<td></td>
<td>HRB_OPEN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HRB_GET</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HRB_ACT_STATUS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HRB_DGET</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HRB_MGET</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HRB_CLOSE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HRB_TERMINATE</td>
<td></td>
</tr>
<tr>
<td>HRB_OBJ_CODE</td>
<td>HRB_HCD</td>
<td>Required for SETUP and TERMINATE.</td>
</tr>
<tr>
<td></td>
<td>HRB_IODF</td>
<td>Required for OPEN and CLOSE.</td>
</tr>
<tr>
<td></td>
<td>HRB_PROCESSOR</td>
<td>Required for processor.</td>
</tr>
<tr>
<td></td>
<td>HRB_CSS</td>
<td>Required for channel subsystem.</td>
</tr>
<tr>
<td></td>
<td>HRB_PCU</td>
<td>Required for physical control unit.</td>
</tr>
<tr>
<td></td>
<td>HRBDEVICE</td>
<td>Required for device.</td>
</tr>
<tr>
<td></td>
<td>HRB_SWITCH</td>
<td>Required for switch.</td>
</tr>
<tr>
<td></td>
<td>HRBCHANNEL</td>
<td>Required for channel path.</td>
</tr>
<tr>
<td></td>
<td>HRB_DATA</td>
<td>Required for DGET.</td>
</tr>
<tr>
<td></td>
<td>HRB_MESSAGE</td>
<td>Required for MGET.</td>
</tr>
<tr>
<td></td>
<td>HRB_IODF</td>
<td>Required for ACT_STATUS.</td>
</tr>
<tr>
<td>HRB_OBJ_NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HRB_OBJ_NR</td>
<td></td>
<td>May be used to specify the name and number of an object. For devices, the number includes the suffix.</td>
</tr>
<tr>
<td>HRB_Q_CODE</td>
<td>HRB_PCU</td>
<td>May be used together with HRB_PROCESSOR to specify that the processor data is qualified by the control unit, device, or switch.</td>
</tr>
<tr>
<td></td>
<td>HRBDEVICE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HRB_SWITCH</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HRBPROCESSOR</td>
<td>May be used together with HRB_CHANNEL to determine the kind of channel path data.</td>
</tr>
<tr>
<td>HRB_Q_NAME</td>
<td></td>
<td>May be used to specify the qualifier name and number for composite names.</td>
</tr>
<tr>
<td>HRB_Q_NR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HRB_REQ_MODE</td>
<td>HRB_MODE_ID</td>
<td>Gets objects starting with the ID specified. The ID of the object must be set in HRB_OBJ_NR or HRB_OBJ_NAME. The HRB_RANGE_VALUE must not be zero.</td>
</tr>
<tr>
<td></td>
<td>HRB_MODE_FIRST</td>
<td>Gets first object in the defined scope.</td>
</tr>
<tr>
<td></td>
<td>HRB_MODE_LAST</td>
<td>Gets last object in the defined scope.</td>
</tr>
<tr>
<td></td>
<td>HRB_MODE_ALL</td>
<td>Gets all objects in the defined scope.</td>
</tr>
<tr>
<td></td>
<td>HRB_MODE_CHAIN</td>
<td>Gets all objects within the chain defined by the given object, for example, all devices of a multi-exposure device.</td>
</tr>
<tr>
<td>HRB_RANGE_VALUE</td>
<td></td>
<td>May be used to specify the number and direction of objects to be processed (positive number = subsequent objects; negative number = preceding objects).</td>
</tr>
<tr>
<td>HRB_TRACE</td>
<td>HRB_YES</td>
<td>The request is traced. Make sure that the trace data set is allocated with a DD name of HCDTRACE.</td>
</tr>
<tr>
<td>HRB_RESULT</td>
<td></td>
<td>The subfields of HRB_RESULT contain the output of the request, such as the data requested, the size of the output data, or return codes.</td>
</tr>
</tbody>
</table>
The HCD application programming interface provides the functions described in Figure 195 on page 479. The functions are listed as you need them while requesting data from the HCD HOM services. The constants you have to specify for HRB_FUNCTION and HRB_OBJ_CODE are included in the figure.
<table>
<thead>
<tr>
<th>Task</th>
<th>Fields in Request Block (HRB)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Set up the connection to the HCD API</strong></td>
<td><strong>HRB_FUNCTION</strong>=</td>
<td>Establish the HCD environment by passing the setup function in the request block. <strong>Input:</strong> HCD session interface (HSI) record.</td>
</tr>
<tr>
<td></td>
<td><strong>HRB_OBJ_CODE</strong>=</td>
<td></td>
</tr>
<tr>
<td>Setup connection</td>
<td>HRB_SETUP</td>
<td>HRB_HCD</td>
</tr>
<tr>
<td><strong>2. Open the IODF</strong></td>
<td></td>
<td>Open an IODF, for which you have read authority, by passing the HRB_OPEN function in the request block.</td>
</tr>
<tr>
<td>Open IODF</td>
<td>HRB_OPEN</td>
<td>HRB_IODF</td>
</tr>
<tr>
<td><strong>3. Request data for HCD objects</strong></td>
<td></td>
<td>Issue a request with the GET function to retrieve data from the IODF. The request returns the address and the length of the data output block. Issue a request with the HRB_DGET function to obtain the retrieved information. <strong>Note:</strong> Issue HRB_GET and HRB_MGET before requesting additional data, because the new request deletes all data and messages from the previous request.</td>
</tr>
<tr>
<td>Get Processor</td>
<td>HRB_GET</td>
<td>HRB_PROCESSOR</td>
</tr>
<tr>
<td>Get Channel Subsystem</td>
<td>HRB_GET</td>
<td>HRB_CSS</td>
</tr>
<tr>
<td>Get Channel Path</td>
<td>HRB_GET</td>
<td>HRB_CHANNEL</td>
</tr>
<tr>
<td>Get Switch</td>
<td>HRB_GET</td>
<td>HRB_SWITCH</td>
</tr>
<tr>
<td>Get Physical Control Unit</td>
<td>HRB_GET</td>
<td>HRB_PCU</td>
</tr>
<tr>
<td>Get Device</td>
<td>HRB_GET</td>
<td>HRB_DEVICE</td>
</tr>
<tr>
<td>Get Activation Status</td>
<td>HRB_ACT_STATUS</td>
<td>HRB_IODF</td>
</tr>
<tr>
<td><strong>4. Get the data from the previous GET request</strong></td>
<td></td>
<td>Issue the request with the address and length of the output area to obtain the data retrieved with the previous GET function. <strong>Output:</strong> Interface record for the object.</td>
</tr>
<tr>
<td>Data Get</td>
<td>HRB_DGET</td>
<td>HRB_DATA</td>
</tr>
<tr>
<td>Message Get</td>
<td>HRB_MGET</td>
<td>HRB_MESSAGE</td>
</tr>
<tr>
<td><strong>5. Close the IODF</strong></td>
<td></td>
<td>When you do not need the IODF anymore, close the IODF by issuing a request with the close function.</td>
</tr>
<tr>
<td>Close IODF</td>
<td>HRB_CLOSE</td>
<td>HRB_IODF</td>
</tr>
<tr>
<td><strong>6. Terminate the connection to the HCD API</strong></td>
<td></td>
<td>When you do not need the HOM services anymore, terminate the connection by issuing a request with the terminate function.</td>
</tr>
<tr>
<td>Terminate connection</td>
<td>HRB_TERMINATE</td>
<td>HRB_HCD</td>
</tr>
</tbody>
</table>

*Figure 195. Functions provided by the HOM services*
Example

The example shows how to get a range of 20 devices, starting with ID X'414' and connected to control unit X'21'. The example is shown in pseudo-code because the actual syntax and declarations depend on the programming language used.

... HRB_SDESC = HRB_SDESC_C
HRB_LENGTH = length-of-HRB
HRB_FUNCTION = HRB_SETUP
HRB_OBJ_CODE = HRB_HCD
Load or link CBDMGHOM with: (HRB,HSI-address,HSI-length,0,0)
...
HRB_SDESC = HRB_SDESC_C
HRB_LENGTH = length-of-HRB
HRB_USE_IODF = IODF-name
HRB_FUNCTION = HRB_OPEN
HRB_OBJ_CODE = HRB_IODF
Load or link CBDMGHOM with: (HRB,0,0,0,0)
...
HRB_SDESC = HRB_SDESC_C
HRB_LENGTH = length-of-HRB
HRB_USE_IODF = IODF-name
HRB_FUNCTION = HRB_GET
HRB_OBJ_CODE = HRB_DEVICE
HRB_OBJ_NR = X'0414000'
HRB_Q_CODE(1) = HRB_PCU
HRB_Q_NR(1) = X'00000021'
HRB_REQ_MODE = HRB_MODE_ID
HRB_RANGE_VALUE = X'00000014'
Load or link CBDMGHOM with: (HRB,0,0,0,0)
...
HRB_SDESC = HRB_SDESC_C
HRB_LENGTH = length-of-HRB
HRB_USE_IODF = IODF-name
HRB_FUNCTION = HRB_DGET
HRB_OBJ_CODE = HRB_DATA
Load or link CBDMGHOM with: (HRB,0,0,DVI-address,DVI-length)
....
HRB_SDESC = HRB_SDESC_C
HRB_LENGTH = length-of-HRB
HRB_USE_IODF = IODF-name
HRB_FUNCTION = HRB_MGET
HRB_OBJ_CODE = HRB_MESSAGE
Load or link CBDMGHOM with: (HRB,0,0,MSI-address,MSI-length)
....
HRB_SDESC = HRB_SDESC_C
HRB_LENGTH = length-of-HRB
HRB_USE_IODF = IODF-name
HRB_FUNCTION = HRB_CLOSE
HRB_OBJ_CODE = HRB_IODF
Load or link CBDMGHOM with: (HRB,0,0,0,0)
....
HRB_SDESC = HRB_SDESC_C
HRB_LENGTH = length-of-HRB
HRB_FUNCTION = HRB_TERMINATE
HRB_OBJ_CODE = HRB_HCD
Load or link CBDMGHOM with: (HRB,0,0,0,0)
...
Return codes

On return, HRB_RETURN_CODE in the request block HRB contains the severity of an error:

- HRB_SEVERE indicates that processing has been terminated and a new setup is required. Issue HCD_MGET to retrieve the messages describing the error.
- HRB_SYNTAX indicates that the request was given to the API in an incorrect syntax and therefore, the request has not been processed.
- HRB_WARNING and HRB_ERROR are given for the remaining errors. Issue HCD_MGET to retrieve the messages describing the error.
- HRB_OK tells you that no problems occurred.

Reason codes

HRB_REASON_CODE in the request block HRB specifies the error in more detail.
Appendix E. Scenarios

The diagram in Figure 196 on page 484 shows an IODF from the viewpoint of the control unit. It shows all attachments of the CU via switches up the processor, as well as the devices attached to the CU. The information displayed for each object is listed below:

• For processors:
  – Processor identification
  – Processor type/model
  – Description
  – Configuration mode
  – Partitions associated with CHPIDs
  – CHPIDs (ID, type, operation mode)

• For switches:
  – Switch identifier and ports
  – Switch type/model

• For CUs:
  – CU type/model
  – CU number
  – Serial number or description

• For devices:
  – Device type/model
  – Device number (starting number and range)

This diagram represents the validated IODF after the completion of the scenario steps described hereafter.

The following scenarios cover the main definition tasks required to produce the IODF illustrated by the diagram. The scenarios have an exemplary character and are not meant to be complete or repetitive (that means, not to show a definition step for every object of the same type).

For detailed information on the Define tasks refer to Chapter 6, “How to define, modify, or view a configuration,” on page 77.
Define operating system configuration data

You have to define the operating systems running on the processor(s), or in one of its partitions first. This includes their EDTs and esoteric names.

1. On the Primary Task Selection panel, select Define, modify, or view configuration data. Press the Enter key. HCD displays the Define, Modify, or View Configuration Data panel.

2. On the Define, Modify, or View Configuration Data panel, select Operating system configurations. HCD displays the Operating System Configuration List panel. Initially, the panel is empty.

3. Use F11=Add to define an operating system. HCD displays the following panel:

   ┌──────────────── Add Operating System Configuration ─────────────────┐
   │ │
   │ Specify or revise the following values. │
   │ OS configuration ID ....... DFSYP501 │
   │ Operating system type .... MVS + │
   │ Description ............. z/OS operating system │
   └────────────────────────────────────────────────────────────────────┐

Figure 196. Example of a Hardware Configuration
4. Specify the name (ID) of the operating system, its type and description, and press the Enter key. HCD displays the updated Operating System Configuration List panel showing the just defined operating system.

<table>
<thead>
<tr>
<th>Config. ID</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPSYS01</td>
<td>MVS</td>
<td>z/OS operating system</td>
</tr>
</tbody>
</table>

After you have defined the operating system, define the EDTs and the esoterics belonging to the EDTs.

To define an EDT proceed as follows:
1. On the Operating System Configuration List panel, select the operating system and the Work with EDTs action from the context menu (or action code 5). HCD displays the EDT List panel. Initially, the panel is empty.
2. Use F11=Add to define an EDT. HCD displays the following panel:

   Specify the following values.
   Configuration ID . : OPSYS01 z/OS operating system
   EDT identifier . . . : A1
   Description . . . : special

3. Specify the EDT identifier and description, and press the Enter key. HCD displays the EDT List panel showing the just defined EDT.

   Select one or more EDTs, then press Enter. To add, use F11.
   Configuration ID . : OPSYS01 z/OS operating system
   / EDT Last Update By Description
   _ A1 2002-10-02 KLAU special

4. Define the esoterics for the just defined EDT. Perform the following steps:
   a. On the EDT List panel, select the EDT and the Work with esoterics action from the context menu (or action code 5). HCD displays the Esoteric List panel. Initially, the panel is empty.
b. Use F11=Add to define an esoteric. HCD displays the following panel:

```
 ┌───────────────────────────── Esoteric List ────────────────────────────┐
 │ Goto Filter Backup Query Help │                                  │
 │ - ┌───────────────────── Add Esoteric ─────────────────────┐ --------- │
 │ │ │ │                                           │                   │
 │ S │ Specify the following values. │ │                    │
 │ C │ Esoteric name . . . __ │ │                    │
 │ E │ VIO eligible . . . No (Yes or No) │   □              │
 │ / │ Token . . . . ___           │   □              │
 └────────────────────────────────────────────────────────┘ │                   │
```

You may define additional operating systems using the steps described above.

Define switches

Because the configuration contains two ESCON directors (switches), you have to define them at this point. This also includes the definition of the switch configuration for each switch. You will have to connect the switches to the various channel paths and control units at a later time.

1. On the Define, Modify, or View Configuration Data panel, select Switches.
   Pressing Enter displays the Switch List panel showing all switches defined in the IODF. Initially, the panel is empty.
2. Use F11=Add to define a switch. HCD displays the following panel:
Enter the required data; you must at least specify:
• Switch identifier
• Switch type

Specify also the switch control unit numbers and switch device numbers. HCD then defines the switch control units, the switch devices, and connects the switch devices to the control units. The switch control units are automatically connected to port FE.

3. Press the Enter key. HCD displays the Switch List panel showing the switch you just created.

4. To define the switch configuration, proceed as follows:
   a. On the Switch List panel, select the switch and the Work with switch configurations (or action code 4). HCD displays all switch configurations defined for the switch. Initially, the panel is empty.
   b. Use F11=Add to define a new switch configuration. HCD displays the following panel:
Define processor-related data

The definition of a processor consists of defining the processor itself, the partitions and the channel paths.

Define processor

Define the processor as follows:

1. On the Define, Modify, or View Configuration Data panel select **Processors**. Press the Enter key. HCD displays the Processor List panel. Initially, the panel is empty.

2. Use F11=Add to define a processor. HCD displays the following panel:

   - **Switch ID**.......: 98  First Switch
   - **Switch Default**...
   - **Connection + Description**
   - **Allow night-shift configuration**

Return to the Switch List panel and repeat the procedure described above to define switch 99 as switch type 9032.
Enter the required data; you must at least specify:
- Processor identifier
- Processor type/model
- Configuration mode

If, for processors other than XMP processors, you specify a value for the field **Number of channel subsystems**, this is ignored.

3. Because more than one support level exists for the 2064-1C1 processor type, HCD displays the following panel:

   ![Available Support Levels Panel]

   **Note:** As indicated by the message on the Available Support Levels panel, you can retrieve an explanation of the processor support level for zSeries processors: Position the cursor on the processor support level description and press PF1 to get an enumeration of functions provided by this support level.

   Select the appropriate support level (for example, Basic 2064 support, IQD, FCP, CF Duplex) and press the Enter key.
4. This completes the processor definition and HCD displays the Processor List panel showing the just defined processor.

Define partitions

To define partition(s) for the channel subsystem, proceed as follows:

1. On the Channel Subsystem List, select the created channel subsystem (CSS ID 0) and the Work with partitions action from the context menu (or action code ). HCD displays all partitions defined for the processor. Initially, the panel is empty.

2. Use F11=Add to define a new partition. HCD displays the following panel:

3. Enter the required data; you must at least specify the name of the partition. Press the Enter key. HCD displays the updated Partition List panel showing the just defined partition.

Define the partitions PROD2, TEST1, and TEST2 following the procedure described above. Finally, the Partition List panel looks like the one shown below:
Define channel path data

After you finished to define the partitions, you can define the channels. Proceed as follows:

1. On the Processor List, select the processor and the Work with attached channel paths from the context menu (or action code S). HCD displays all CHPIDs defined for the processor. Initially, the panel is empty.

2. Use F11=Add to define a new channel path. HCD displays the following panel:

Enter the required data; you must at least specify:
- Channel path identifier
- Channel path type
- Operation mode

Note: Four channel paths are defined in one step.

3. Press the Enter key. HCD displays the Define Access List panel. The panel shows all partitions defined for the processor.
4. Select the partition that must have access to the channel by using a /.
Because the channel path’s operation mode is dedicated, you can only select one partition.

5. Press the Enter key. HCD displays the updated Channel Path List.

Define channel paths 10 and 11 in the same way, however with operation mode REC. Note that they are connected to partition PROD2 (see Figure 196 on page 484).

The definition of a shared channel path (20 and 21) connected to a switch is slightly different. Proceed as follows:

1. Use F11=Add to define a new channel path. HCD displays the following panel:

2. Enter the required data. Note that the values for the switch and port values are only valid for channel path 20. After pressing the Enter key, HCD displays the Update CHPID Settings panel. Update the values for channel path 21 as shown in the following figure.
3. Press the Enter key. HCD displays the Define Access List panel. This panel shows all partitions defined for the processor. Select the partitions that are in the access list of the channel paths.

4. Press the Enter key. HCD displays the Define Candidate List panel. Select the partitions that are in the candidate list of the channel paths.

5. Press the Enter key. HCD displays the updated Channel Path List.

Define channel paths 25 and 26 in a similar way so that you will to achieve the following resulting Channel Path List:
To view the matrix of the Channel Path list, scroll to the right with the F20=Right key.

Use the F3=Exit key to return to the Processor List panel. Use the F3=Exit key again, to return to the Define, Modify, or View Configuration Data panel.

Define processor PROC2 in the same way.
Complete switch definitions

After you have defined the processors and the channel paths, you can complete the switch definitions. Basically you have to connect the switch control units to processor PROC1. Because the switches are chained together, this connection will be established as well.

**Connect switch control unit to processor**

1. On the Define, Modify, or View Configuration Data panel, select Control units. Press the Enter key. HCD displays the Control Unit List panel. This panel shows the two switch control units.

   ```
   Goto Filter Backup Query Help
   --------------------------------------------------------------------------
   Control Unit List Row 1 of 2
   Select one or more control units, then press Enter. To add, use F11.
   ---#---
   / CU Type + CUADD CSS MC Serial-# + Description
   c 0098 9033 55-9999 First Switch
   _ 0099 9032 55-8888 Second Switch
   ********************************************************** Bottom of data **********************************************************
   ```

2. Select control unit 0098 and the Change action from the context menu (or action code /SF580000c/SF590000). HCD displays the Change Control Unit Definition panel with the control unit that already has been correctly defined. Press the Enter key. HCD displays the Select Processor / CU panel.

   ```
   Select Processor / CU Row 1 of 2 More: >
   Command ==> __________________________ Scroll ==> PAGE
   Select processors to change CU/processor parameters, then press Enter.
   Control unit number .. : 0098  Control unit type .. : 9033
   -------------------------------------------------- Channel Path ID . Link Address +
   / Proc.CSSID 1------ 2------ 3------ 4------ 5------ 6------ 7------ 8------
   s PROC1 _______ _______ _______ _______ _______ _______ _______ _______
   _ PROC2 _______ _______ _______ _______ _______ _______ _______ _______
   ```

3. Select the processor PROC1 and the Select (connect/change) action from the context menu (or action code /S). HCD displays the following panel.
4. Enter the required data; you must at least specify:
   - Channel path
   - Link address
   - Unit address

5. Press the Enter key. HCD displays the Modify Device Parameters panel. This panel indicates that the unit address 98, that HCD defined when the switch device was specified, was changed to 00.

6. Press the Enter key to accept the change. HCD displays the Select Processor / CU panel showing the just specified switch control unit settings.

7. Press once more the Enter key to return to the Control Unit List panel.

   Establish the connection of switch 99 to processor PROC1 in the same way as shown in Figure 196 on page 484.

**Complete port data**

1. On the Define, Modify, or View Configuration Data panel select **Switches**. The Switch List panel showing all defined switches is displayed.
2. On the Switch List panel, select switch 98 and the **Work with ports** action from the context menu (or action code [p]). HCD displays the Port List, showing all ports with their attachments to channels, control units or other switches that are supported by the selected switch.

```
Goto Options Filter Backup Query Help
------------------------------------------------------------------------
Port List   Row 1 of 17                                               Scroll ==> PAGE
------------------------------------------------------------------------
Command ===> _____________________________________________ Scroll ==> PAGE

Select one or more ports, then press Enter.

Switch ID . . . . . : 98   Address :   First Switch
Switch configuration ID : night shift configuration

/ Port H Name +       Unit ID   Unit Type O B CON +
  _C0 Y               N N
  _C1 Y PROCICHPID20  PR PROC1   CHP 20 2064-1C1 N N
  _C2 Y               N N
  _C3 Y               N N
  _C4 Y PROCICHPID21  PR PROC1   CHP 21 2064-1C1 N N
  _C5 Y               N N
  _C6 Y               N N
  _C7 Y PROCICHPID25  PR PROC1   CHP 25 2064-1C1 N N
  _C8 N               -- --
  _C9 N               -- --
  _CA N               -- --
  _CB N               -- --
  _CC N               -- --
  _CD N               -- --
  _CE N               -- --
  _CF N               -- --
  _FE Y               CU 0098 9033 N N
------------------------------------------------------------------------
```
2. Press the Enter key. HCD displays the updated Port List panel. Specify the port name for port C5.

Specify a dedicated connection between port C4 and port C5 and port C1 and C2.

```
Goto Options Filter Backup Query Help
------------------------------------------------------------------------
Command ==> Port List Row 1 of 17 Scroll ==> PAGE
------------------------------------------------------------------------
Select one or more ports, then press Enter.
Switch ID . . . . . . . . . . . . : 98 Address : First Switch
Switch configuration ID : BASIC night shift configuration
/ Port H Name + Unit ID Unit Type O B Con+ Ded
- C0 Y N N
- C1 Y PROCICHPID20 PR PROC1 CHP 20 2064-1C1 N N C2
- C2 Y N N C1
- C3 Y N N
- C4 Y PROCICHPID21 PR PROC1 CHP 21 2064-1C1 N N C5
- C5 Y CHAIN_FROM_99 SW 99 PO FD 9032 N N C6
- C6 Y N N
- C7 Y PROCICHPID25 PR PROC1 CHP 25 2064-1C1 N N
- C8 N
- C9 N
- CA N
- CB N
- CC N
- CD N
- CE N
- CF N
- FE Y CU 0098 9033 N N
------------------------------------------------------------------------
******************************* Bottom of data ********************************
```

3. Use F3=Exit to return to the Switch List panel. Use F3=Exit again to return to the Define, Modify, or View Configuration Data panel.

**Define control unit data**

After defining the processor, its partitions and channel paths, and switches, you can define control units:

1. On the Define, Modify, or View Configuration Data panel select **Control units**.

   Press the Enter key. HCD displays the Control Unit List panel. So far, the panel shows the switch control units.

2. Use F11=Add to define a control unit. HCD displays the following panel:
Control Unit List

Select one or more control units, then press Enter. To add, use F11.

---#---

/ CU Type + CUADD CSS MC Serial-# + Description
_ 0098 9033 55-9999 First Switch
_ 009

Specify or revise the following values.
Control unit number .... 00E2 +
Control unit type .... 3174 +
Serial number ........ Terminal control unit
Description ........... Terminal control unit
Connected to switches ... 99 +
Ports ............. E4 +

If connected to a switch:
Define more than eight ports . . 1. Yes
2. No
Propose CHPID/link addresses and
unit addresses ........ 1. Yes
2. No

Enter the required data; you must at least specify:
• Control unit number
• Control unit type

If the control unit is connected to a switch, also specify the switch/ports.

3. Press the Enter key. HCD displays the Select Processor / CU panel. This panel
shows a list of all processors you already have defined.

Select Processor / CU Row 1 of 2 More: >
Command ==> __________________________ Scroll ==> PAGE

Select processors to change CU/processor parameters, then press Enter.
Control unit number . : 00E2  Control unit type . : 3174

-----------Channel Path ID . Link Address + -----------
/ Proc.CSSID 1----- 2----- 3----- 4----- 5----- 6----- 7----- 8-----
s PROC1
PROC2

******************************* Bottom of data ********************************

4. On the Select Processor / CU panel, select the processor and the Select
(connect/change) action from the context menu (or action code 5). HCD
displays the Add Control Unit panel.
5. Specify channel path IDs and link address (21 and E4). For this example, change the number of units to 32.

Press the Enter key. HCD displays the Select Processor / CU panel, now indicating that the processor PROC1 is connected to the control unit. This is shown by a Yes in the Att column, which you can reach by scrolling to the right.

6. This completes the definition of control unit 00E2, because it is only attached to processor PROC1. Press the Enter key to return to the Control Unit List panel now showing the just defined control unit.

7. The other control units are defined similarly, resulting in the following Control Unit List panel:
8. Press the F3=Exit key to return to the Define, Modify, or View Configuration Data panel.

**Define I/O device data**

When you define a device or a group of devices, you have to connect them to the processors (CSS), and then to the operating systems supposed to access the device(s). The operating system definition for the device(s) consists of specifying the parameters and features, and of grouping these device(s) to esoterics.

Proceed as follows:

1. On the Define, Modify, or View Configuration Data panel select **I/O devices**. Press the Enter key. HCD displays the I/O Device List. This list shows the switch devices.

2. Use F11=Add to define a device or a range of devices. HCD displays the following panel:

```
Add Device

Specify or revise the following values.
Device number ........ 0001  (0000 - FFFF)
Number of devices .... 1
Device type .......... 3278-3
Serial number ......... __________
Description ........... ________________________________
Volume serial number ..... ______ (for DASD)
Connected to CUs ... 00E1

F1=Help  F2=Split  F3=Exit  F4=Prompt  F5=Reset  F9=Swap
F12=Cancel
```

Specify the physical characteristics of the device. You must at least specify:
- Device number
- Device type/model
Specify also the control unit(s) the device is attached to.

Press the Enter key. HCD displays the Device / Processor Definition panel showing all processors that have a path to the control units the device is attached to.

| Row1of2 |
| Select processors to change device/processor definitions, then press Enter. |
| Device number ..: 0001 Number of devices ..: 1 |
| Device type ..: 3278-3 |
| / Proc.CSSID SS+ UA+ Time-Out STADET CHPID + Explicit Null |
| Preferred Device Candidate List |
| _ PROC1 __ Yes Yes __ No ___ |
| _ PROC2 __ Yes No __ ___ |
| ************************** BOTTOM OF DATA ******************************* |

3. Press the Enter key to accept a default unit address (unit address 01). HCD displays the Define Device to Operating System Configuration panel. This panel shows all operating systems defined in the IODF.

| Row1of1 | Define Device to Operating System Configuration |
| Select OSs to connect or disconnect devices, then press Enter. |
| Device number ..: 0001 Number of devices ..: 1 |
| Device type ..: 3278-3 |
| / Config. ID Type SS Description Defined |
| OPSYS01 MVS z/OS operating system |
| ************************** BOTTOM OF DATA ******************************* |

To define the device to an operating system proceed as follows:

a. On the Define Device to Operating System Configuration panel, select the operating system and the Select (connect/change) action from the context menu (or action code 5). HCD displays the Define Device Parameters / Features panel.
b. Specify the parameters and features and press the Enter key. HCD displays the Assign/Unassign Device to Esoteric panel.

c. Overwrite the values in the Assigned column to assign (Yes) and unassign (No) devices to the esoterics.

d. Press the Enter key to display the Define Device to Operating System Configuration panel.

e. Press the Enter key to return to the I/O Device List panel.

f. Define additional devices in the same way.

g. Press the F3=Exit key to display the Define, Modify, and View Configuration panel.

Define NIPCON data

1. On the Define, Modify, or View Configuration Data panel select *Operating system configurations*. Press the Enter key. HCD displays the Operating System Configuration List panel.

2. On the Operating System Configuration List panel, select the operating system for which a NIP console has to be defined and the *Work with consoles* action from the context menu (or action code [M]). Press the Enter key. HCD displays the NIP Console List panel. Initially, the panel is empty.

3. Use the F11=Add key to define a NIP console. HCD displays the following panel:
4. Enter the required data. Press the Enter key to return to the updated NIP Console List panel.
5. Press the PF3=Exit key to return to the Operating System Configuration List panel.
6. Press the PF3=Exit key to return to the Define, Modify, or View Configuration Data panel.
7. Press the F3=Exit key to return to the Primary Task Selection panel.
Appendix F. IODF data model

The following tables describe the IODF data model in terms of object class and attribute definitions. These definitions conform to the data that is kept in the IODF.

The object classes and attribute types introduced by HCD start with the prefix hcd which helps with identifying them.

The object hierarchy below the HCD Backend suffix is structured as follows:

hcdIodf (RDN: hcdIodfId)
  ├── hcdControlUnit (RDN: hcdControlUnitNumber)
  │    └── hcdDevice (RDN: hcdDeviceNumber[hcdDeviceSuffix])
  │       └── hcdDistPackage (RDN: hcdDistPackageId)
  │           └── hcdOsConfig (RDN: hcdOsConfigId)
  │               ├── hcdEligibleDeviceTable (RDN: hcdEligibleDeviceTableId)
  │               │     └── hcdEsotericDeviceGroup (RDN: hcdEsotericDeviceGroupId)
  │               │         └── hcdGenericDeviceType (RDN: hcdGenericDeviceTypeId)
  │               │             └── hcdOsDevice (RDN: hcdDeviceNumber[hcdDeviceSuffix])
  │               └── hcdProcessorConfig (RDN: hcdProcessorConfigId[hcdCssId])
  │                   └── hcdChannelPath (RDN: hcdChannelPathId)
  │                       └── hcdCssControlUnit (RDN: hcdControlUnitNumber)
  │                           └── hcdCssDevice (RDN: hcdDeviceNumber[hcdDeviceSuffix])
  │                               └── hcdIocds (RDN: hcdIocdsId)
  │                                   └── hcdPartition (RDN: hcdPartitionId[hcdPartitionNumber])
  │                                       └── hcdSwitch (RDN: hcdSwitchId)
  │                                           └── hcdPort (RDN: hcdPortId)
  │                                               └── hcdSwitchConfig (RDN: hcdSwitchConfigId)
  │                                                   └── hcdPortConfig (RDN: hcdPortId)
IODF object class definitions within LDAP

The following tables describe the object classes which are used for the HCD LDAP DIT. In addition to other information, they contain the names of the attributes of each object class. The required attributes (shown in the row 'Must Contain') must be specified when adding an object of that class. Their values cannot be deleted. Optional attributes (shown in the row 'May Contain') may not always be required. They may, however, be necessary for certain objects according to HCD configuration rules. Attributes which are used to build the RDN of an object must not be modified for the corresponding object class.

The row ‘Supported Requests’ contains information about which operations are possible on entries of the corresponding object class. There are some object classes which do not allow all contained (non-RDN) attributes to be added (because they are read-only attributes) or modified. In this case, the list of attributes which may be modified is specified in the ‘Supported Requests’ row in parentheses after the MODIFY identifier.

Note: For certain entries, there may be restrictions to this general description as a result of HCD configuration rules.

The following rules apply for update requests on control unit and device object classes:

1. Control Unit

The general attributes of a specific control unit exist in an appropriate entry in class hcdControlUnit. For each connected processor, the processor related attributes of that control unit are contained in an entry of class hcdCssControlUnit. Hence, as a result of this relation, an entry can only be added to class hcdCssControlUnit if the corresponding entry in class hcdControlUnit is already present. Likewise, if an entry of class hcdControlUnit is deleted, all the entries for the corresponding control unit in class hcdCssControlUnit will implicitly also be deleted.

2. Devices

In a manner similar to the control units, the general attributes of a specific device exist in an entry of class hcdDevice and may additionally exist in entries of classes hcdCssDevice and hcdOsDevice. The device attributes relating a device to a particular processor are contained in a corresponding entry of class hcdCssDevice. Such an entry can only exist if an entry for the device in class hcdDevice also exists and the device is connected to a control unit defined via, both an entry in the class hcdControlUnit, and an entry in class hcdCssControlUnit. This dependency means that it is not possible to explicitly add or delete an entry of class hcdCssDevice directly. Instead, an entry of class hcdCssDevice will be implicitly added if:

- A control unit defined in an hcdControlUnit entry that has the particular device attached to it is defined also as an entry in class hcdCssControlUnit.
- An entry in class hcdDevice is created containing the attribute hcdConnControlUnits with a control unit number of an hcdControlUnit entry that also has a corresponding hcdCssControlUnit entry defined.

Implicitly added entries of class hcdCssDevice initially have default attribute values. These attributes can be modified to get their proper values.

An entry of class hcdCssDevice will be implicitly deleted if:
- The entry in corresponding class hcdDevice is deleted, or
- The connected control units in class hcdCssControlUnit are deleted, or
- The connected control units in class hcdControlUnit are deleted.

Changes to the attributes of control unit and device entries may affect all corresponding entries in the same logical control unit of a processor configuration, i.e. all entries of classes hcdCssControlUnit and hcdCssDevice with the same value of the attribute hcdLogicalControlUnit. In such a case, it may be necessary to delete all entries belonging to the same logical control unit and add them again with the changed attributes. This should be done using the update sequence in transaction mode.

For each operating system configuration, the operating system related device attributes are contained in a corresponding entry in class hcdOsDevice. An entry in class hcdOsDevice can only exist if the corresponding entry in class hcdDevice also exists. An entry in class hcdOsDevice can be explicitly added or deleted. This means that the device is connected to or disconnected from the operating system configuration. An entry in class hcdOsDevice is implicitly deleted if the corresponding entry in class hcdDevice is deleted.

If a device entry in class hcdDevice is added which has the same RDN attribute (hcdDeviceNumber) value of an existing entry in that class, HCD assigns a non-zero suffix to the device number (RDN attribute hcdDeviceSuffix) to make the RDN unique. The attribute hcdDeviceSuffix cannot be specified with an add request. If the value of attribute hcdDeviceSuffix is 0000, it need not be specified in the RDN, as this is the default value. If the value of the attribute hcdDeviceSuffix is not equal to 0000, it has to be specified in the RDN of the MODIFY and DELETE requests for the corresponding device entry. Therefore, the complete device RDN has to be retrieved via a preceding search request in order to be able to specify the correct RDN of a device entry.

**Note:** To facilitate the search of the correct hcdDeviceSuffix of devices in a device group, you can specify an LDAP search request for a specific device number, but without device suffix, or with device suffix 0000. Such a request will return all device entries with the same device number with its different device suffixes. From the returned entries, you can retrieve the suffix for the device that you now can modify or delete with a subsequent request.

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdChannelPath</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes a channel path of the processor configuration</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdProcessorConfig</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdChannelPathId, hcdChannelPathType, hcdOperationMode, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDescription, hcdDynSwitch, hcdConnChannelPath, hcdAccessingPartitions, hcdCandidatePartitions, hcdConnPort, hcdIsOccupied, hcdIsManaged, hcdSysplex, hcdMaximumFrameSize, hcdSpanningChannelSubsystems, hcdPhysicalChannelId, hcdHasPrioQueuesDisabled, hcdHcaAdapterId, hcdHcaPort</td>
</tr>
<tr>
<td>Class</td>
<td>hcdChannelPath</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH, ADD, MODIFY, DELETE</td>
</tr>
<tr>
<td>Special Notes</td>
<td>Attribute hcdIsOccupied is only shown if set to Yes. Attribute hcdSpanningChannelSubsystems is output only. A spanned channel path is added/modified in one channel subsystem while specifying partitions from more than one channel subsystem. It can be defined with operation mode SHR or SPAN. Attribute hcdHasPrioQueuesDisabled is only shown if applicable.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdControlUnit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes a control unit</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td>Derived from top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdIodf</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdControlUnitNumber, hcdUnit, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDescription, hcdModel, hcdSerialNumber, hcdConnPorts</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH, ADD, MODIFY, DELETE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdCssControlUnit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes a control unit from the Channel Subsystem viewpoint</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td>Derived from top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdProcessorConfig</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdUnitAddressRanges, hcdControlUnitAddress, hcdIOConcurrencyLevel, hcdControlUnitProtocol, hcdLogicalControlUnit, hcdManagedChannelPathsCount</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdUnitAddressRanges, hcdControlUnitAddress, hcdIOConcurrencyLevel, hcdControlUnitProtocol, hcdManagedChannelPathsCount</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH, ADD (requires the existence of an object in class hcdControlUnit with same RDN), MODIFY (hcdConnChannelPaths, hcdUnitAddressRanges, hcdControlUnitAddress, hcdIOConcurrencyLevel, hcdControlUnitProtocol, hcdManagedChannelPathsCount), DELETE</td>
</tr>
<tr>
<td>Class</td>
<td>hcdCssDevice</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>Description</td>
<td>Describes a device from the Channel Subsystem viewpoint</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdProcessorConfig</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdDeviceNumber, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDeviceSuffix, hcdDeviceRange, hcdUnitAddress, hcdStatusDetection, hcdTimeOut, hcdPreferredChannelPath, hcdCandidatePartitions, hcdLogicalControlUnit, hcdSubchannelSetId</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH</td>
</tr>
<tr>
<td></td>
<td>MODIFY (hcdDeviceRange, hcdUnitAddress, hcdStatusDetection, hcdTimeOut, hcdPreferredChannelPath, hcdCandidatePartitions)</td>
</tr>
<tr>
<td>Special Notes</td>
<td>For a MODIFY request, hcdDeviceRange specifies the number of devices to which the request will be applied.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdDevice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes an I/O device</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdIodf</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdDeviceNumber, hcdUnit, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDescription, hcdModel, hcdDeviceSuffix, hcdDeviceRange, hcdSerialNumber, hcdVolser, hcdConnControlUnits</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH</td>
</tr>
<tr>
<td></td>
<td>ADD</td>
</tr>
<tr>
<td></td>
<td>MODIFY (hcdDeviceRange, hcdUnit, hcdDescription, hcdModel, hcdSerialNumber, hcdVolser, hcdConnControlUnits)</td>
</tr>
<tr>
<td></td>
<td>DELETE</td>
</tr>
<tr>
<td>Special Notes</td>
<td>For ADD and MODIFY requests, hcdDeviceRange specifies the number of devices to which the request will be applied.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdDistPackage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes a distribution package of I/O configurations</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Class</td>
<td>hcdDistPackage</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdIodf</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdDistPackageId, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDescription, hcdTargetNode, hcdTargetUser, hcdIsAttended, hcdTargetIodf, hcdTargetVolser, hcdSentDate, hcdSentTime, hcdOsConfigs, hcdProcessorConfigs</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH ADD MODIFY (hcdDescription, hcdTargetNode, hcdTargetUser, hcdIsAttended, hcdTargetIodf, hcdTargetVolser, hcdOsConfigs, hcdProcessorConfigs) DELETE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdEligibleDeviceTable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes an Eligible Device Table of an OS configuration</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdOsConfig</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdEligibleDeviceTableId, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDescription, hcdLastUpdateDate, hcdUpdatedBy</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH ADD MODIFY (hcdDescription) DELETE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdEsotericDeviceGroup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes an esoteric device group of an EDT of an OS configuration</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdEligibleDeviceTable</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdEsotericDeviceGroupId, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdEsotericDeviceGroupToken, hcdVirtualIO, hcdDeviceRanges</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH ADD MODIFY DELETE</td>
</tr>
<tr>
<td>Special Notes</td>
<td>Attribute hcdVirtualIO is only shown if set to Yes.</td>
</tr>
</tbody>
</table>
### Class hcdGenericDeviceType

<table>
<thead>
<tr>
<th>Description</th>
<th>Describes an update to a generic device type for an EDT of an OS configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdEligibleDeviceTable</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdGenericDeviceTypeId, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDevicePreferenceValue, hcdVirtualIO</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH, MODIFY</td>
</tr>
<tr>
<td>Special Notes</td>
<td>Entries of class hcdGenericDeviceType are implicitly added when the first device entry with the corresponding generic device type is added in class hcdOsDevice. An entry of this class is deleted, if the last device with the corresponding generic device type is deleted from class hcdOsDevice. Attribute hcdVirtualIO is only shown if set to Yes.</td>
</tr>
</tbody>
</table>

### Class hcdIocds

<table>
<thead>
<tr>
<th>Description</th>
<th>Describes an IOCDS of a processor configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdProcessorConfig</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdIocdsId, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdIocdsName, hcdLastUpdateDate, hcdLastUpdateTime, hcdProcessorConfigMode</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH</td>
</tr>
<tr>
<td>Special Notes</td>
<td>Entries in this class are implicitly added when an entry in class hcdProcessorConfig is added. Entries in this class are implicitly deleted if the appropriate entry in class hcdProcessorConfig is deleted.</td>
</tr>
</tbody>
</table>

### Class hcdIodf

<table>
<thead>
<tr>
<th>Description</th>
<th>Describes the I/O configurations defined in an IODF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td></td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdIodfId, objectClass</td>
</tr>
<tr>
<td>Class</td>
<td>hcdIodf</td>
</tr>
<tr>
<td>---------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdIodfType, hcdIodfDescription, hcdBackupIodf, hcdBlocksAllocated, hcdBlocksUsed, hcdCreationDate, hcdLastUpdateDate, hcdLastUpdateTime</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH</td>
</tr>
<tr>
<td></td>
<td>MODIFY (hcdIodfDescription)</td>
</tr>
<tr>
<td>Special Notes</td>
<td>Entries of this class cannot be added or deleted with the HCD LDAP Backend.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdOsConfig</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes an OS configuration</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td>Derived from top</td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdIodf</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdOsConfigId, hcdOsConfigType, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDescription</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH</td>
</tr>
<tr>
<td></td>
<td>ADD</td>
</tr>
<tr>
<td></td>
<td>MODIFY (hcdDescription)</td>
</tr>
<tr>
<td></td>
<td>DELETE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdOsDevice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes an I/O device from the OS viewpoint</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td>Derived from top</td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdOsConfig</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdDeviceNumber, hcdUnit, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDeviceSuffix, hcdDeviceRange, hcdModel, hcdGenericDeviceTypeId, hcdDeviceParameters, hcdDeviceFeatures, hcdConsoleNumber, hcdSubchannelSetId</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH</td>
</tr>
<tr>
<td></td>
<td>ADD (requires the existence of an object in class hcdDevice with the same RDN)</td>
</tr>
<tr>
<td></td>
<td>MODIFY (hcdDeviceRange, hcdDeviceParameters, hcdDeviceFeatures, hcdConsoleNumber)</td>
</tr>
<tr>
<td></td>
<td>DELETE</td>
</tr>
<tr>
<td>Special Notes</td>
<td>For ADD and MODIFY requests, hcdDeviceRange specifies the number of devices to which the request will be applied.</td>
</tr>
<tr>
<td>Class</td>
<td>hcdPartition</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Description</td>
<td>Describes a logical partition (LP) of a processor configuration</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxilliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdProcessorConfig</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdPartitionId, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDescription, hcdPartitionNumber, hcdPartitionUsage</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH, ADD, MODIFY, DELETE</td>
</tr>
<tr>
<td>Special Notes</td>
<td>You can use the hcdPartitionNumber attribute to identify a reserved partition.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdPort</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes a port of an ESCON director</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxilliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdSwitch</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdPortId, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdPortName, hcdIsOccupied, hcdIsInstalled, hcdConnPort</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH, MODIFY</td>
</tr>
<tr>
<td>Special Notes</td>
<td>Attribute hcdIsOccupied is only shown if set to Yes. Entries of this class are implicitly added or deleted if the corresponding switch entry in class hcdSwitch is added or deleted, respectively.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>hcdPortConfig</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes a port configuration of a switch configuration</td>
</tr>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td></td>
</tr>
<tr>
<td>Derived from</td>
<td>top</td>
</tr>
<tr>
<td>Auxilliary Classes</td>
<td></td>
</tr>
<tr>
<td>Possible Superiors</td>
<td>hcdSwitchConfig</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdPortId, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdAllowedDynConnPorts, hcdProhibitedDynConnPorts, hcdDedicatedPort, hcdIsBlocked</td>
</tr>
</tbody>
</table>
### Class hcdPortConfig

**Supported Requests**
- SEARCH
- MODIFY

**Special Notes**
Entries in this class are implicitly added or deleted if the corresponding entry in class hcdSwitchConfig is added or deleted, respectively.

### Class hcdProcessorConfig

**Description**
Describes a processor configuration or, if it is identified by an RDN of the form hcdProcessorConfigId+CsslID, it describes a channel subsystem of an XMP processor.

**Type**
structural

**OID**
Derived from top

**Auxiliary Classes**

**Possible Superiors**
hcdIodf

**Must Contain**
hcdProcessorConfigId, hcdUnit, hcdProcessorConfigMode, objectClass

**May Contain**
hcdModel, hcdDescription, hcdSerialNumber, hcdNetworkName, hcdSystem, hcdSupportLevel, hcdCssId, hcdSetZeroMaxDevices, hcdSetOneMaxDevices, hcdLocalSystemName

**Supported Requests**
- SEARCH
- ADD
- MODIFY
- DELETE

**Special Notes**
Adding a channel subsystem requires that the processor configuration has already been added.

To add, change or delete a channel subsystem, an RDN consisting of hcdProcessorConfigId+CsslID has to be used. The only applicable further attributes for a channel subsystem are hcdDescription, hcdSetZeroMaxDevices and hcdSetOneMaxDevices. The values of the required parameters hcdUnit and hcdProcessorConfigMode are ignored.

### Class hcdSwitch

**Description**
Describes a switch (ESCON or FICON Director or Fibre Channel switch)

**Type**
structural

**OID**
Derived from top

**Auxiliary Classes**

**Possible Superiors**
hcdIodf

**Must Contain**
hcdSwitchId, hcdUnit, objectClass

**May Contain**
hcdDescription, hcdModel, hcdSerialNumber, hcdSwitchAddress
### Class hcdSwitch

<table>
<thead>
<tr>
<th>Supported Requests</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEARCH</td>
</tr>
<tr>
<td>ADD</td>
</tr>
<tr>
<td>MODIFY</td>
</tr>
<tr>
<td>DELETE</td>
</tr>
</tbody>
</table>

### Class hcdSwitchConfig

<table>
<thead>
<tr>
<th>Description</th>
<th>Describes a configuration of an ESCON Director</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>structural</td>
</tr>
<tr>
<td>OID</td>
<td>Derived from top</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td>Possible Superiors: hcdSwitch</td>
</tr>
<tr>
<td>Must Contain</td>
<td>hcdSwitchConfigId, objectClass</td>
</tr>
<tr>
<td>May Contain</td>
<td>hcdDescription, hcdDefaultConn</td>
</tr>
<tr>
<td>Supported Requests</td>
<td>SEARCH</td>
</tr>
<tr>
<td></td>
<td>ADD</td>
</tr>
<tr>
<td></td>
<td>MODIFY</td>
</tr>
<tr>
<td></td>
<td>DELETE</td>
</tr>
</tbody>
</table>

### IODF attribute table

The following table describes the names and the properties of the attributes (alias names are given in parentheses).

**Notes:**

1. All attribute type names ending with the plural "s" represent multi-valued attributes. HCD interprets the values as an ordered list. This is important if there are two lists which have corresponding list items.
2. In a request, a mix of base names and alias names for attributes is not supported.

<table>
<thead>
<tr>
<th>Attribute (alias name in parentheses)</th>
<th>Description</th>
<th>Syntax</th>
<th>Valued</th>
<th>Access Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>hcdAccessingPartitions (hcdAccList)</td>
<td>Channel path access list (up to 8 char partition names)</td>
<td>cis / 8</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>hcdAllowedDynConnPorts (hcdAllowedList)</td>
<td>Ports to which an allowed dynamic connection exists (2 hexadecimal digit port address)</td>
<td>cis / 2</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>hcdBackupIodf (hcdBkupIodf)</td>
<td>Data set name of backup IODF (up to 35 character full-qualified IODF data set name)</td>
<td>cis / 35</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdBlocksAllocated (hcdAllocBlks)</td>
<td>Number of allocated blocks for IODF (up to 6 digit decimal number, read-only)</td>
<td>cis / 6</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>Attribute (alias name in parentheses)</td>
<td>Description</td>
<td>Syntax</td>
<td>Valued</td>
<td>Access Class</td>
</tr>
<tr>
<td>-------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>hcdBlocksUsed (hcdUsedBlks)</td>
<td>Number of used blocks in IODF (up to 6 digit decimal number, read-only)</td>
<td>cis / 6</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdCandidatePartitions (hcdCandList)</td>
<td>Partition candidate list (up to 8 character partition names or a 0 (zero) for a null device candidate list)</td>
<td>cis / 8</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>hcdChannelPathId (hcdChpId)</td>
<td>Channel path identifier (2 digit hexadecimal number)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdChannelPathType (hcdChpType)</td>
<td>Channel path type (4 character channel path type acronym)</td>
<td>cis / 4</td>
<td>single</td>
<td>normal</td>
</tr>
</tbody>
</table>
| hcdConnChannelPath (hcdConnChp)                        | Connected channel path of a coupling facility connection (qualified coupling facility connection: prid.chpid.cunum.devn (if the target is an SMP processor) or prid.cssid.chpid.cunum.devn (if the target is an XMP processor), where
   prid is a up to 8 character processor name,
   cssid is a one hexadecimal digit channel subsystem ID,
   chpid is a 2 hexadecimal digit channel path,
   cunum is a 4 hexadecimal digit control unit number,
   devn is a 4 hexadecimal digit device number)
   If the values for cunum and devn are missing, they are defaulted as needed. | cis / 24 | single | normal       |
| hcdConnChannelPaths (hcdChpList)                       | Connected channel paths / link addresses (pp or pp.ll or pp.llll, where
   pp is the 2-digit hexadecimal number of the channel path, optionally qualified by
   ll as a 2 hexadecimal number for the link address or by llll as a 4 hexadecimal number for the link address) | cis / 7 | multi  | normal       |
<p>| hcdConnControlUnits (hcdCuList)                        | Connected control units (4 hex digit numbers)                              | cis / 4 | multi  | normal       |</p>
<table>
<thead>
<tr>
<th>Attribute (alias name in parentheses)</th>
<th>Description</th>
<th>Syntax</th>
<th>Valued</th>
<th>Access Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>hcdConnPort</td>
<td>Connected switch port (qualified port address ss.pp, where ss is the 2 digit hexadecimal switch number, pp is the 2 digit hexadecimal port address)</td>
<td>cis / 5</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdConnPorts</td>
<td>List of connected switch ports (qualified port addresses ss.pp, where ss is the 2 digit hexadecimal switch number, pp is the 2 digit hexadecimal port address)</td>
<td>cis / 5</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>hcdConsoleNumber</td>
<td>Console order number (up to 2 digit decimal number)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdControlUnitAddress</td>
<td>Logical control unit address (up to 2 hex digit number)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdControlUnitNumber</td>
<td>Control unit number (4 hex digit number)</td>
<td>cis / 4</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdControlUnitProtocol</td>
<td>Control unit protocol (D, S, S4)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdCreationDate</td>
<td>Creation Date (yyyy-mm-dd, read-only)</td>
<td>cis / 10</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdCssId</td>
<td>Channel subsystem identifier (1 digit hexadecimal number)</td>
<td>cis / 1</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDedicatedPort</td>
<td>Port to which a dedicated connection exists (2 digit hexadecimal number)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDefaultConn</td>
<td>Dynamic connection default of a switch configuration (ALLOW, PROHIBIT)</td>
<td>cis / 10</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDescription</td>
<td>Description field (up to 32 character string)</td>
<td>cis / 32</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDeviceFeatures</td>
<td>Device feature list (up to 10 character identifier)</td>
<td>cis / 10</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDeviceNumber</td>
<td>Device number (4 digit hexadecimal number)</td>
<td>cis / 4</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDeviceParameters</td>
<td>Device parameter list (keyword=value1,...)</td>
<td>cis / 50</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDevicePreferenceValue</td>
<td>Device preference value (up to 8 digit decimal number)</td>
<td>cis / 8</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>Attribute (alias name in parentheses)</td>
<td>Description</td>
<td>Syntax</td>
<td>Valued</td>
<td>Access Class</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-------------</td>
<td>--------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>hcdDeviceRange (hcdRange)</td>
<td>Range of devices with same attributes (up to 4 digit decimal number)</td>
<td>cis / 4</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDeviceRanges (hcdDevRngeList)</td>
<td>Device range list (qualified value: xxxx.ddd, where xxxx is a 4 hex digit device number, ddd is a up to 3 decimal digit range number,</td>
<td>cis / 8</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDeviceSuffix (hcdSuf)</td>
<td>Device ID suffix (4 digit hexadecimal number)</td>
<td>cis / 4</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDistPackageld (hcdPkgName)</td>
<td>Name of distribution package (up to 8 alphanumeric character identifier)</td>
<td>cis / 8</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdDynSwitch</td>
<td>Dynamic switch of channel path (2 digit hexadecimal number)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdEligibleDeviceTableId (hcdEdtId)</td>
<td>EDT identifier (2 character identifier)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdEsotericDeviceGroupId (hcdEsoId)</td>
<td>Esoteric name (up to 8 character identifier)</td>
<td>cis / 8</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdEsotericDeviceGroupToken (hcdEsoToken)</td>
<td>Esoteric token (up to 4 decimal digit number)</td>
<td>cis / 4</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdGenericDeviceTypeId (hcdGenId)</td>
<td>Generic device type name (up to 8 character identifier)</td>
<td>cis / 8</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdHasPrioQueuesDisabled</td>
<td>Indicates if an OSD channel path has queue prioritizing disabled (Yes, No).</td>
<td>cis / 3</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdHcaAdapterId</td>
<td>HCA adapter ID (hexadecimal number).</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdHcaPort</td>
<td>HCA port (decimal number).</td>
<td>cis / 1</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdIocdsId</td>
<td>IOCDS identifier (2 digit hexadecimal number)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdIocdsName</td>
<td>IOCDS name (up to 8 character identifier)</td>
<td>cis / 8</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdIOLConcurrencyLevel (hcdIocl)</td>
<td>I/O concurrency level of control unit (1 digit decimal number)</td>
<td>cis / 1</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdIodfDescription (hcdIodfDesc)</td>
<td>IODF description (up to 128 character string)</td>
<td>cis / 128</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdIodfId</td>
<td>IODF name (up to 35 character full-qualified data set name)</td>
<td>cis / 35</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdIodfType</td>
<td>IODF type (I (initial), W (work), P (production), read-only)</td>
<td>cis / 1</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>Attribute (alias name in parentheses)</td>
<td>Description</td>
<td>Syntax</td>
<td>Valued</td>
<td>Access Class</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>hcdIsAttended</td>
<td>Indicates if target node is attended (Yes, No)</td>
<td>cis / 3 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdIsBlocked</td>
<td>Indicates if the port is blocked (Yes, No)</td>
<td>cis / 3 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdIsInstalled</td>
<td>Indicates if object (port) is installed (Yes, No)</td>
<td>cis / 3 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdIsManaged</td>
<td>Indicates if a channel path is managed (Yes, No)</td>
<td>cis / 3 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdIsOccupied</td>
<td>Indicates if object (port, channel path) is occupied (Yes, No)</td>
<td>cis / 3 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdLastUpdateDate</td>
<td>Date of last update (yyyy-mm-dd, read-only)</td>
<td>cis / 10 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdLastUpdateTime</td>
<td>Time of last update (hh:mm:ss, read-only)</td>
<td>cis / 8 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdLocalSystemName</td>
<td>Local system name (up to 8 alphanumeric character identifier)</td>
<td>cis / 8 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdLogicalControlUnit</td>
<td>Logical control unit number (4 digit hexadecimal number, read-only)</td>
<td>cis / 4 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdManagedChannelPathsCount</td>
<td>Maximum number of managed channel path connections to a control unit (1 digit decimal number)</td>
<td>cis / 1 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdMaximumFrameSize</td>
<td>Maximum frame size for the transmission unit on an IQD channel path in KB (decimal number)</td>
<td>cis / 6 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdModel</td>
<td>Model (4 character identifier)</td>
<td>cis / 4 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdNetworkName</td>
<td>CPC network name (up to 8 alphanumeric identifier)</td>
<td>cis / 8 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdOperationMode</td>
<td>Channel path operation mode (DED, REC, SHR)</td>
<td>cis / 3 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdOsConfigId</td>
<td>OS configuration identifier (up to 8 alphanumeric character identifier)</td>
<td>cis / 8 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdOsConfigs</td>
<td>List of OS configurations (up to 8 alphanumeric character identifiers)</td>
<td>cis / 8 multi normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdOsConfigType</td>
<td>OS configuration type (MVS, VM)</td>
<td>cis / 3 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdPartitionId</td>
<td>Logical partition name (up to 8 alphanumeric character identifier or an * for reserved partitions)</td>
<td>cis / 8 single normal</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>Attribute (alias name in parentheses)</td>
<td>Description</td>
<td>Syntax</td>
<td>Valued</td>
<td>Access Class</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-------------</td>
<td>--------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>hcdPartitionNumber (hcdPartNo)</td>
<td>Partition image number (1 digit hexadecimal number)</td>
<td>cis / 1</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdPartitionUsage (hcdPartUsage)</td>
<td>Partition usage (OS, CF, CF/OS)</td>
<td>cis / 5</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdPhysicalChannelId</td>
<td>Physical channel identifier for a channel path of an XMP processor (3 digit hexadecimal number or ‘*’ for an over-defined channel path)</td>
<td>cis / 3</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdPortId</td>
<td>Switch port identifier (2 digit hexadecimal number)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdPortName</td>
<td>Switch port name (up to 24 character identifier)</td>
<td>cis / 24</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdPreferredChannelPath (hcdPrefChp)</td>
<td>Preferred channel path (2 digit hexadecimal number)</td>
<td>cis / 2</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdProcessorConfigId (hcdProcId)</td>
<td>Processor configuration ID (up to 8 alphanumeric character identifier)</td>
<td>cis / 8</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdProcessorConfigMode (hcdConfMode)</td>
<td>Processor configuration mode (BASIC, LPAR)</td>
<td>cis / 5</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdProcessorConfigs (hcdProcList)</td>
<td>List of processor configurations (up to 8 alphanumeric character identifiers)</td>
<td>cis / 8</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>hcdProhibitedDynConnPorts (hcdProhibitList)</td>
<td>Ports to which a Prohibited dynamic connection exists (2 digit hexadecimal numbers)</td>
<td>cis / 2</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>hcdSentDate</td>
<td>Date of last distribution (yyyy-mm-dd, read-only)</td>
<td>cis / 10</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdSentTime</td>
<td>Time of last distribution (hh:mm:ss, read-only)</td>
<td>cis / 8</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdSerialNumber (hcdSerialNo)</td>
<td>Serial number (up to 10 character serial number)</td>
<td>cis / 10</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdSetOneMaxDevices</td>
<td>Maximum number of devices for subchannel set 1 (SS 1) in a channel subsystem of an XMP processor (6 digit decimal number).</td>
<td>cis / 6</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdSetZeroMaxDevices</td>
<td>Maximum number of devices for subchannel set 0 (SS 0) in a channel subsystem of an XMP processor (6 digit decimal number).</td>
<td>cis / 6</td>
<td>single</td>
<td>normal</td>
</tr>
<tr>
<td>hcdSpanningChannelSubsystems</td>
<td>List of spanning channel subsystems for a spanned channel path (1 digit hexadecimal number, read only).</td>
<td>cis / 1</td>
<td>multi</td>
<td>normal</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Syntax</td>
<td>Valued</td>
<td>Access Class</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>hcdStatusDetection (hcdStadet)</td>
<td>Status detection facility of device (Yes, No)</td>
<td>cis / 3 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdSubchannelSetId</td>
<td>ID of the subchannel set where a device is located in a channel subsystem of an XMP processor.</td>
<td>cis / 1 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdSupportLevel</td>
<td>Processor support level (8 character identifier)</td>
<td>cis / 8 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdSwitchAddress</td>
<td>Address of a Fibre Channel switch in a fabric (2 digit hexadecimal number)</td>
<td>cis / 2 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdSwitchConfigId (hcdSwConfId)</td>
<td>Switch configuration identifier (up to 8 alphanumeric character identifier)</td>
<td>cis / 8 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdSwitchId (hcdSwID)</td>
<td>Switch identifier (2 digit hexadecimal number)</td>
<td>cis / 2 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdSysplex</td>
<td>Sysplex name (up to 8 character alphanumeric identifier)</td>
<td>cis / 8 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdSystem</td>
<td>CPC system name (up to 8 character alphanumeric identifier)</td>
<td>cis / 8 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdTargetIodf</td>
<td>Data set name of target IODF (up to 35 character full-qualified data set name)</td>
<td>cis / 35 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdTargetNode</td>
<td>Target node (up to 8 character alphanumeric identifier)</td>
<td>cis / 8 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdTargetUser</td>
<td>Target user (up to 8 character alphanumeric identifier)</td>
<td>cis / 8 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdTargetVolser (hcdTargetVol)</td>
<td>Volume serial number of target IODF data set (up to 6 character alphanumeric identifier)</td>
<td>cis / 6 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdTimeOut</td>
<td>Time out facility (Yes, No)</td>
<td>cis / 3 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdUnit</td>
<td>Unit (up to 8 alphanumeric character identifier)</td>
<td>cis / 8 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdUnitAddress (hcdUa)</td>
<td>Unit address (2 digit hexadecimal number)</td>
<td>cis / 2 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdUnitAddressRanges (hcdUaList)</td>
<td>Unit address range of control unit (qualified number xx.ddd, where xx is a 2 digit hexadecimal unit address ddd is a up to 3 digit decimal range)</td>
<td>cis / 6 multi</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdUpdatedBy</td>
<td>Identifier of last update user (up to 8 alphanumeric character identifier, read-only)</td>
<td>cis / 8 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>hcdVirtualIO (hcdVio)</td>
<td>Virtual I/O (Yes, No)</td>
<td>cis / 3 single</td>
<td>normal</td>
<td></td>
</tr>
<tr>
<td>Attribute (alias name in parentheses)</td>
<td>Description</td>
<td>Syntax</td>
<td>Valued</td>
<td>Access Class</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-------------</td>
<td>--------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>hcdVolser (hcdVolume)</td>
<td>Volume serial number (up to 6 character alphanumeric identifier)</td>
<td>cis / 6</td>
<td>single</td>
<td>normal</td>
</tr>
</tbody>
</table>
Accessibility

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/Library Server versions of z/OS books in the Internet library at:

http://www.ibm.com/systems/z/os/zos/bkserv/
Notices

This information was developed for products and services offered in the U.S.A.

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.

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:

IBM Corporation
Mail Station P300
522 South Road
Poughkeepsie New York 12601-5400
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document 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.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, 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

© Copyright IBM Corp. 1994, 2009
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 pointers in this publication to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement. IBM accepts no responsibility for the content or use of non-IBM Web sites specifically mentioned in this publication or accessed through an IBM Web site that is mentioned in this publication.

If you are viewing this information in softcopy form, the photographs and color illustrations may not appear.

Policy for unsupported hardware

Various z/OS elements, such as DFSMS™, HCD, JES2, JES3, and MVS™, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

IBM
The IBM logo
ibm.com
ACF/VTAM
AFP
BookMaster
DB2
DFSMS/MVS
DFSORT
ESCON
ES/9000
FICON
GDDM
Language Environment
MVS
MVS/DFP
MVS/ESA
MVS/SP
MVS/XA
OS/390
Parallel Sysplex
Print Services Facility
Processor Resource/Systems Manager
PR/SM
RACF
Resource Link
RETAIN
SP
System/390
System z
System z9
System z10
S/370
S/390
S/390 Parallel Enterprise Server
VTAM
z/Architecture
z/OS
zSeries
z/VM
z9
z10

IBM, the IBM logo, ibm.com and DB2 are registered trademarks of International Business Machines Corporation in the United States, other countries, or both.
Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

Other company, product, and service names may be trademarks or service marks of others.

Programming interface information

This guide primarily documents information that is NOT intended to be used as a Programming Interface of Hardware Configuration Definition (HCD).

This information unit also documents intended Programming Interfaces that allow the customer to write programs to obtain the services of HCD. This information is identified where it occurs, either by an introductory statement to a topic or by the following marking:

Programming Interface information

End of Programming Interface information

Accessing licensed books on the web

z/OS licensed documentation in PDF format is available on the Internet at the IBM Resource Link Web site at:

http://www.ibm.com/servers/resourcelink

Licensed books are available only to customers with a z/OS license. Access to these books requires an IBM Resource Link Web user ID and password, and a key code. With your z/OS order you received a memo that includes this key code.

To obtain your IBM Resource Link Web user ID and password log on to:

http://www.ibm.com/servers/resourcelink

To register for access to the z/OS licensed books:
1. Log on to Resource Link using your Resource Link user ID and password.
2. Click on User Profiles located on the left-hand navigation bar.
3. Click on Access Profile.
4. Click on Request Access to Licensed books.
5. Supply your key code where requested and click on the Submit button.

If you supplied the correct key code you will receive confirmation that your request is being processed. After your request is processed you will receive an e-mail confirmation.

Note: You cannot access the z/OS licensed books unless you have registered for access to them and received an e-mail confirmation informing you that your request has been processed.
To access the licensed books:
1. Log on to Resource Link using your Resource Link user ID and password.
2. Click on Library.
3. Click on zSeries.
4. Click on Software.
5. Click on z/OS.
6. Access the licensed book by selecting the appropriate element.

Online message lookup

The LookAt system for online lookup of all z/OS messages has been introduced. For more information, see "Using LookAt to look up message explanations" on page 452.
Glossary

This glossary defines technical terms and abbreviations used in the Hardware Configuration Definition (HCD) documentation. If you do not find the term you are looking for, refer to the index of the appropriate HCD document or view the IBM Glossary of Computing Terms, available from: http://www.ibm.com/ibm/terminology.

A

Activity log. The activity log is a sequential data set with the name of the associated IODF and the suffix ACTLOG. Use the activity log to document all definitions you made to the current IODF using HCD.

B

Base. Base is the base device number of a multiple exposure device, which is accessible by more than one device number. You assign the first device number and the system generates the additional device numbers.

C

Central processor complex (CPC). A physical collection of hardware that consists of central storage, one or more central processors, timers, and channels.

CFReport. When a machine is ordered, the output of the order process is a binary file that represents the physical description of the final machine. One of the components of that file is the type and physical location, including the Physical Channel Identifier (PCHID) value assigned to that location, of all the I/O features in the final machine. This file is called a CFReport.

Change log. The change log is a VSAM data set with the name of the associated IODF and the suffix CHLOG. It will be automatically created if change logging and automatic activity logging is active. A subset of its generated entries will then be used to create the activity log entries.

Channel subsystem (CSS). A collection of subchannels that directs the flow of information between I/O devices and main storage. It uses one or more channel paths as the communication link in managing the flow of information to or from I/O devices. Within the CSS is one subchannel set and logical partitions. One subchannel from the set is provided for and dedicated to each I/O device accessible to the CSS. Logical partitions use subchannels to communicate with I/O devices. The maximum number of CSSs supported by a processor also depends on the processor type. If more than one CSS is supported by a processor, each CSS has a processor unique single hexadecimal digit CSS identifier (CSS ID).

CHPID. A logical processor contains a number of CHPIDs, or Channel Path IDs, which are the logical equivalent of channels in the physical processor. See also:
• dedicated CHPID
• reconfigurable CHPID
• shared CHPID
• spanned CHPID

CHPID Mapping Tool. The CHPID Mapping Tool aids the customer in developing a CHPID-to-PCHID relationship for XMP processors. It accepts an IOCP input file without PCHID values, allows the user to assign the logical CHPID values in the input to the PCHIDs available with his ordered machine, and returns an updated IOCP input file that contains the PCHID values.

CMT. See CHPID Mapping Tool.

Coupling Facility (CF). The hardware element that provides high-speed caching, list processing, and locking functions in a sysplex. To enable data sharing between a CF partition and the central processor complexes, special types of high-speed, CF channels are required to provide the connectivity. A receiving CF channel path, attached to a CF partition, is to be connected to a sending CF channel path, attached to a partition in which an operating system (OS) is running.

Coupling Facility Channel. A high bandwidth fiber optic channel that provides the high-speed connectivity required for data sharing between a coupling facility and the central processor complexes directly attached to it.

CSS. See channel subsystem.

D

Dedicated CHPID. A CHPID can be dedicated to one partition; only that partition can access I/O devices on this CHPID. All CHPID types can operate in DED (dedicated) mode.

Dynamic reconfiguration. The ability to make changes to the channel subsystem and to the operating system while the system is running.
EDT. An EDT (eligible device table) is an installation-defined and named representation of the devices that are eligible for allocation. The EDT defines the esoteric and generic relationship of these devices. During IPL, the installation identifies the EDT that the operating system uses. After IPL, jobs can request device allocation from any of the esoteric device groups assigned to the selected EDT. An EDT is identified by a unique ID (two digits), and contains one or more esoterics and generics. Define at least one EDT for each operating system configuration.

Enterprise Systems Connection (ESCON). A set of products and services that provides a dynamically connected environment using optical cables as a transmission medium.

ESCON Manager (ESCM). A licensed program that provided host control to help manage connections that use ESCON Directors. The functionality has been incorporated into the I/O Operations component of System Automation for z/OS.

ESCON Multiple Image Facility (EMIF). EMIF is now referred to as MIF. See below.

Esoteric. Esoteric (or esoteric device group) is an installation-defined and named grouping of I/O devices of usually the same device group. EDTs define the esoteric and generic relationship of these devices. The name you assign to an esoteric is used in the JCL DD statement. The job then allocates a device from that group instead of a specific device number or generic device group.

FICON. Fiber Connection Environment (FICON) is an improved optical fiber communication method offering channels with high data rate, high bandwidth, increased distance and a greater number of devices per control unit for S/390 systems. It can work together with, or replace ESCON links.

Generic. Generic (or generic device type) is an MVS-defined grouping of devices with similar characteristics. For example: the device types 3270-X, 3277-2, 3278-2, -2A, -3, -4, and 3279-2a, -2b, -2c, -3a, -3b belong to the same generic. Every generic has a generic name that is used for device allocation in the JCL DD statement. MVS interprets this name as "take any device in that group". In an operating system configuration, each EDT has the same list of generics. This list can only vary by the preference values and VIO indicators that are assigned to the generics.

HCPURO data set. The data set containing a real I/O configuration of a VM system.

Hardware Management Console. A console used to monitor and control hardware such as the System/390 microprocessors.

IOCDS. An input/output configuration data set (IOCDS) contains different configuration definitions for the selected processor. Only one IOCDS is used at a time. The IOCDS contains I/O configuration data on the files associated with the processor controller on the host processor, as it is used by the channel subsystem. The CSS uses the configuration data to control I/O requests. The IOCDS is built from the production IODF.

I/O Cluster. An I/O cluster is a sysplex that owns a managed channel path for an LPAR processor configuration.

IOCP. An IOCP (I/O configuration program) is the hardware utility that defines the hardware I/O configuration to the channel subsystem. For this definition IOCP retrieves information from the IOCP input data set about the following: the channel paths in the processor complex, control units attached to the channel paths, and I/O devices assigned to the control unit. HCD users can build the IOCP input data set from a production IODF.

IODF. An IODF (input/output definition file) is a VSAM linear data set that contains I/O definition information. This information includes processor I/O definitions (formerly specified by IOCP input streams) and operating system I/O definitions (formerly specified by MVSCP input streams). A single IODF can contain several processor and several operating system I/O definitions. See also [531].

I/O Operations. A component of System Automation for z/OS providing functionality formerly available with ESCON Manager.

Initial program load (IPL). The process that loads the system programs from the auxiliary storage, checks the system hardware, and prepares the system for user operations.

LCSS. Logical channel subsystems. See also channel subsystem.

LDAP. LDAP (lightweight directory access protocol) is an Internet protocol standard, based on the TCP/IP protocol and serves to access and manipulate data.
organized in a Directory Information Tree (DIT). LDAP V3 is specified in RFC 2251 and is specifically targeted at management and browser applications that provide read/write interactive access to directories. HCD makes IODF data accessible via LDAP using the IBM Tivoli Directory Server for z/OS.

Local system name. When defining an XMP processor, you can specify an optional local CPC designator. If you do not specify a local system name, and a CPC name is given, the local system name defaults to the CPC name.

Logical control unit. A logical control unit (LCU) can be a single CU with or without attached devices or a group of one or more CUs that share devices. In a channel subsystem, a logical CU represents a set of CUs that physically or logically attach I/O devices in common. A logical CU is built from the information specified in the CU definitions. The physical CUs the device is attached to form part of a logical CU.

Logically partitioned (LPAR) mode. A central processor complex (CPC) power-on reset mode that enables use of the PR/SM feature and allows an operator to allocate CPC hardware resources (including central processors, central storage, expanded storage, and channel paths) among logical partitions. Contrast with basic mode.

M

Master configuration file (MCF). The HCM master configuration file (MCF) is an HCM configuration stored on the host. It provides a central shared repository, allowing several HCD/HCM users to work on a single configuration cooperatively and safely.

Master IODF. A master IODF is a centrally kept IODF containing I/O definitions for several systems or even for a complete enterprise structure. Master IODFs help to maintain consistent I/O data within a system and can provide comprehensive reports. From the master IODF subset IODF may be generated to serve as production IODFs for particular systems within the structure.

MCF. See master configuration file.

Migration. Refers to activities that relate to the installation of a new version or release of a program to replace an earlier level. Completion of these activities ensures that the applications and resources on your system will function correctly at the new level.

Multiple Image Facility (MIF). A facility that allows channels to be shared among PR/SM logical partitions in an ESCON or FICON environment.

Multiple exposure device. A multiple exposure device is allocated by a single device number, but accessed by several device numbers, whereby each device number represents one exposure. The device number by which the device is allocated is the base exposure; all other device numbers are called non-base exposures.

Multi-user access. Users can define the multi-user access attribute for IODFs so that multiple users can simultaneously update this IODF. An IODF is kept in exclusive update mode only for the duration of a single transaction. If the updates of the transaction are committed, another user may update the IODF without requiring the first user to release it. Though a user’s changes are not immediately refreshed in the views of the other users, each user has a consistent view of the data either from the initial access to the IODF or after each last update that he had applied to the IODF.

MVS system. An MVS image together with its associated hardware, which collectively are often referred to simply as a system, or MVS system.

MVSCP. MVSCP (MVS configuration program) is the program that defines the I/O configuration to MVS. For this definition, information about devices, EDTs, and NIP consoles is required.

N

NIP console. A NIP (nucleus initialization program) console is a device that NIP uses as a console to display system messages. To define a device as a NIP console, it must first be defined to the channel subsystem and the current operating system (OS) configuration.

P

PCHID. See physical channel identifier.

Peer coupling channel. A peer coupling channel is a coupling channel operating in peer mode, which means it can be used as a sender and receiver at the same time. It may be shared by several logical OS partitions (such as CF sender channels) and by a CF logical partition. In addition, peer channels provide more buffer sets and channel bandwidth than their counterparts. Peer channels are supported only on zSeries 900 servers and their successors.

Physical channel identifier (PCHID). The physical address of a channel path in the hardware. Logical CHPIDs have corresponding physical channels. Real I/O hardware is attached to a processor via physical channels. Channels have a physical channel identifier (PCHID) which determines the physical location of a channel in the processor. For XMP processors, the PCHIDs must be defined in the configuration. The PCHID is a three hexadecimal digit number and is assigned by the processor. One logical channel path (CHPID) provided by a channel subsystem may be associated with a physical channel (PCHID). There is no standard mapping between CHPIDs and PCHIDs.
The CHPID Mapping Tool aids the customer in developing a CHPID-to-PCHID relationship. See also CHPID Mapping Tool.

Preference value. Preference value is the value that is assigned to each generic. This value determines the sequence of allocation. The generics and the associated values are system-defined. The predefined order can be changed by means of the preference value.

Processor Resource/Systems Manager (PR/SM). The feature that allows the processor to use several OS images simultaneously and provides logical partitioning capability. See also LPAR.

Production IODF. The production IODF is used by MVS/IPL to build UCBs and EDTs. It is also used to build IOCDSs and IOP input data sets. Several users can view a production IODF concurrently and make reports of it, but it cannot be modified. The production IODF that is used for IPL must be specified by a LOADxx member. The LOADxx member can reside either in SYS1.PARMLIB or SYSn.IPLPARM. If the LOADxx member resides in SYSn.IPLPARM, then SYSn.IPLPARM must reside on the IODF volume. If the LOADxx member resides in SYS1.PARMLIB, then SYS1.PARMLIB can reside on either the system residence (sysres) volume or the IODF volume.

Reconfigurable CHPID. A reconfigurable CHPID is an unshared CHPID that you can reconfigure offline from one partition, then online to another. That is, the CHPID can be reconfigured between logical partitions after a power-on reset. Only one partition can access I/O devices on this CHPID at a time. All CHPID types can operate in REC (reconfigurable) mode.

Server Time Protocol link. A coupling facility connection which will be used as a timing-only link, providing the Server Time Protocol (STP) function. The STP is a time synchronization feature, which is designed to provide the capability for multiple System z9 and zSeries servers to maintain time synchronization with each other. STP is designed to allow events occurring in different System z9 and zSeries servers to be properly sequenced in time.

Shared CHPID. A shared CHPID can be configured online to one or more partitions at the same time. One or more partitions can access I/O devices at the same time using this CHPID.

SMP processor. In this book, this term designates processors supporting a single channel subsystem. For SMP processors, the single channel subsystem is implicitly defined with the processor. This term is used in contrast to the term XMP processor, which designates processors supporting multiple logical channel subsystems.

SNA address. The system network architecture (SNA) address is a means to identify the support element of a CPC configured in a S/390 microprocessor cluster. It consists of the network name (the network identifier of the LAN the support element of a CPC is connected to), and the system name (the identifier of the CPC within the network). In HCD, used as part of a processor definition for a CPC, the SNA address provides the association of a processor defined in the IODF with a CPC configured in an S/390 microprocessor cluster.

Spanned CHPID. With XMP processors, supporting multiple logical channel subsystems, some types of channel paths can be shared across partitions from multiple logical channel subsystems. It is dependent on the processor support, which channel types can be defined as spanned. Such a channel path is called a spanned channel path.

A spanned channel path will be created with the same CHPID number in all channel subsystems that are using it. For example, if you have a processor MCSSPRO1 with channel subsystems 0 through 3, and you create CHPID 1A (type IQD, SPAN) and let it access partitions from CSS 0, 2, and 3, you will get this CHPID 1A in CSSs 0, 2, and 3, but not in CSS 1.

STP link. See Server Time Protocol link.

Subchannel set. With a subchannel set you can define the placement of devices either relative to a channel subsystem or to an operating system. Starting with IBM System z9 Enterprise Class (z9 EC) processors and z/OS V1R7, users can define an additional subchannel set with ID 1 (SS 1) on top of the existing subchannel set (SS 0) in a channel subsystem. This function relieves the constraint for the number of devices that can be accessed by an LPAR. The machine implementation for IBM System z9 Enterprise Class processors or later supports 63.75K devices in subchannel set 0, and up to 64K-1 devices in the additional subchannel set 1. The z/OS V1R7 implementation limits the exploitation of subchannel set 1 to parallel access volume (PAV) alias devices only (device types 3380A, 3390A of the 2105, 2107 and 1750 DASD control units).

Sysplex. A set of operating systems communicating and cooperating with each other through certain multisystem hardware components and software services to process customer workloads. See also MVS system.

S/390 Microprocessor Cluster. S/390 microprocessor cluster is a configuration that consists of CPCs (central processor complexes), one or more Hardware Management Consoles, and may have one or more coupling facilities. The support elements that are attached to the CPCs are connected to a network. A
Hardware Management Console connected to the same network allows the system operator to configure the CPCs, observe and control hardware operations, and perform software functions.

**U**

UCB. Unit control block

UIM. UIMs (unit information modules) perform the device-dependent part of the operating system configuration definition. There is a UIM for each supported device or device group. Each UIM recognizes and processes the values coded for its device or device group. HCD routines load all UIMs, either IBM or customer supplied, into virtual storage and make calls to the UIMs:
- During initialization
- During processing of an Add device or Change device request
- During generation of a print report
- During IPL

**V**

Validated work IODF. A validated work IODF satisfies all validation rules for building production IODFs. It may lack physical channel identifiers (PCHIDs) for XMP processors. In cooperation with HCD and the CHPID Mapping Tool a validated work IODF is required to accept new or updated PCHIDs. From such a validated work IODF, an IOCP input deck suitable for the use with the CHPID Mapping Tool is generated. As soon as all PCHIDs are inserted or updated in the validated work IODF, the production IODF can be built.

VIO. VIO (virtual input/output) is the allocation of data sets that exist in paging storage only. Only DASDs are eligible for VIO. Data sets are allocated to a paging device instead of to a real device.

**W**

Work IODF. The work IODF is used to update an I/O definition and reflects the most recent status of the hardware configuration. After you have completed the updates, you can use the work IODF to create a production IODF. While you can update a work IODF and generate reports from it, it cannot be used to build UCBS and EDTs, nor can it be used to generate an IOCDs, or an IOCP input data set.

**X**

XMP processor. In the S/390 context, this term designates processors that support multiple logical channel subsystems (LCSS). It is used in contrast to the term SMP processor, which designates processors of previous generations that support only one channel subsystem. In general, the different CSSs including their channel paths and logical partitions provided by an XMP processor operate independently from each other. Channel paths can be spanned over multiple logical channel subsystems on the same processor depending on the channel path type. See also SMP processor and channel subsystem.
Index

Special characters
*SHCD* 210
*SHCDC* 210

Numerics
4-character model 209
8-character control unit 209

A
abend code
   00F 448
   other than 00F 449
abnormal termination 448
access authority 339
access list 114
accessibility 523
action bar 59, 371
action code 65, 371
action list 64
action-to-action processing 64
actions on objects, migration enhancements 292
activate messages 226
activating
   configuration dynamically 219
   switch configuration 191
   sysplex-wide 224
trace 469
active hardware configuration token stored 225
active IODF, view 219
active switch 189
active sysplex member list 224
activity log
   glossary definition 529
   naming convention 36
Activity Log panel 55
activity logging 54
   automatic xxiv, 26, 55
   browsing 251
   HCD profile 26
   printing 251
   rules 56
   volume serial number 26
adding
   configuration package object 52
   configuration packages 50
   device to esoteric device list, migration 295
objects through migration 292
advantages of extended migration 273
all-character readable token 209
ALLOC_SPACE 27
allocating output data sets
   volume serial number 26
allow hardware deletes 221
alter access 340
alternate DD names 338
analyze errors
   batch migration 272
   dialog migration 271
   HCD problems 447
API provided by HCD 475
application programming interface 475
assembler check, migration 271, 303
assembler statements 264
assembly listing 303
assigning devices to esoterics 89
associated data sets 36
associated with processor and partition, migration 270
attended export 45
attribute group change 150
auto-assign 131
automatic activity logging
   rules 56
build
   CONFIGxx member, dialog 228
   CONFIGxx, batch 320
   FCP device data, batch 320
   HCPRIOR input data set, batch 319
   IOCDs, batch 316
   IOCDs, dialog 201
   IOCP input data set 210
   IOCP input data set, batch 316
   IOCP input data set, dialog 208
   JES3, batch 320
   OS configuration data set 215
   production IOCD, batch 272, 314
   production IODF, dialog 196
   S/390 microprocessor IOCDs, dialog 203
   work IODF, batch 316
build configuration statements 273

B
back-level IODF 15, 48
backup an IODF 42
base, glossary definition 529
batch jobs
   generated by HCD 75
batch utility
   build configuration data 320
   build CONFIGxx member 320
   build IOCDs or IOCP input data set 316
   build JES3 initialization stream checker data 320
   build production IODF 314
   build work IODF 316
   compare IODFs or CSS/OS reports 328
   copy IODF 323
   create graphical configuration reports 327
   description 307
   example for copy IODF 335
   example for migration 272
   EXEC procedures 34
   export IODF 333
   FCP device data 320
   functions 307
   import IODF 332
   initialize IODF 309
   input parameter string 308
   job steps during transmit procedure 458
   migrate an input data set 272, 311
   print configuration reports 324
   starting the dialog 308
   upgrade IODF 310
BCTC 163
   blocking ports 185
   BookMaster 239
   BOTTOM command 60
   browse activity log 251
   build
   CONFIGxx, batch 320
   FCP device data, batch 320
   HCPRIOR input data set, batch 319
   IOCDs, batch 316
   IOCDs, dialog 201
   IOCP input data set 210
   IOCP input data set, batch 316
   IOCP input data set, dialog 208
   JES3, batch 320
   OS configuration data set 215
   production IOCD, batch 272, 314
   production IODF, dialog 196
   S/390 microprocessor IOCDs, dialog 203
   work IODF, batch 316
   build configuration statements 273
   C
candidate list 115
cascade switching 169
cascaded FICON fabric 133
catalog
   considerations 343
   scan utility 344
cataloged data set 343
   CBDCHCD 21, 22
   CBDMGHOM routine 475
centralized navigation 78
   CF channel path, migration enhancements 285
   CF duplexing 10
   CFReport 213, 529
   chained switches 170, 181
   change log
      glossary definition 529
      naming convention 36
   change log data set
      allocation space extension 27
   change log file 471
   name 55
   size 55
   CHANGE_LOG 26
changing
   channel path attributes, migration 294
   channel paths 123
   configuration packages 51
   consoles 158
   control unit and device attributes, migration 294
   control unit attachment parameters for multiple processors 135
   control units 134
coupling facility connection 126
CSS-related definitions for a group of devices 148
device number 153
devices 146
changing (continued)
DYNAMIC parameter for a group of devices 149
DET ID 86
DETIs 85
esoterics 89
esoterics for a group of devices 148
IODF attributes 44
LOCANY parameter for a group of devices 149
operating mode of channel paths 124
operating system configuration 82
OS configuration ID 82
OS-related definitions for a group of devices 148
partition name, migration 295
partitions 107
ports from installed to uninstalld 183
ports from occupied to not occupied 183
processor 93
processor ID 96
processor type 93
production IODF 42
switch configuration ID 186
switch configurations 175
switch data 175
switches 175
to another IODF 42
type of CF channel paths 124
type of channel paths 124
type/model of a group of devices 150
changing channel path ID 123
channel path
access list 114
aggregating 126
candidate lists 115
changing 123
CHPID statement 284
compare report, example 438
connectivity report, example 383
defining 113
deleting 128
detail report, example 381
migration enhancement 248
operation mode 112
spanned 119
summary report, example 380
working with 112
channel path ID
changing 123
channel paths
over-defining on XMP processors 119
channel subsystem 100
changing 104
copy 101
copying 103
defining 100
deleting 105
glossary definition 529
channel subsystem report, example 378
CHLOG_EXTENSION 27
CHPID
glossary definition 529
spanned 119
CHPID Mapping Tool 213
glossary definition 529
CHPID report 259
CHPID statement 284
CIB channel 118
CIB channel path 22
CLIST used by HCD 22
CNC channel 163
CNC side information 165
CNTLUNIT statement 287
costiction considerations 57
color settings for graphical reports 29
combined input data sets 269
command
syntax diagrams 29
command line 68
commands
BOTTOM 60
FIND 186
GOTO 70
HELPID 452
HELPTEST 452
how to use 68
INSTRUCT 60
LEFT 60
LOCATE 60
LOCATE, graphical report 243
PANELID 451
PFSHOW 68
REFRESH, graphical report 244
RIGHT 60
SAVE 243, 245
SHOWIODF 45
SHOWMSG 466
SHOWMSG ACTIVATE 222
TOP 60
TRACE 469
TRACE, HCM 471
compare
CONFIGxx members 230
CSS/OS views, dialog 250
IODFs or CSS/OS Reports, batch 328
IODFs, dialog 246
compare reports
channel path 438
control unit 440
control unit attachment 439
CSS / OS device 446
device 441
device attachment 440
esoteric 444
OS console 446
OS device 445
partition 437
processor 436
switch 442
switch configuration detail 443
switch detail 442
compatibility 57
IODF 18
component ID 447
concentration 19
configuration data
activating dynamically 219
consolidating 78
define 77
definition sequence 80
configuration data (continued)
delete 77
modify 77
process 195
validation 3
configuration package
add 52
define 50
delete 51, 52
edit 51
list 50, 51
merge 52
problem determination for transmit
procedure 458
replace 52
transmit 53
work with 50, 51
configuration report
build with batch 320
configuration reports
CF channel path connectivity report,
example 383
channel path detail report,
example 381
channel path summary report,
example 380
channel subsystem reports,
example 377
control unit detail report,
example 384
control unit summary report,
example 384
CSS report, description 235
CTC connection, description 236
CTC report, example 396
device detail report, example 387
device summary report, example 387
device table report,
example 393
examples 377
graphical configuration report 238
I/O definition reference 237
I/O definition reference,
example 431
I/O path report 397
I/O path, description 236
IOCDS report, example 379
LCU report, example 435
MVS device detail report,
example 393
MVS device report, example 391
NIP console report, example 394
operating system report,
description 236
operating system report,
example 391
operating system summary report,
example 391
partition report, example 378
print with batch 324
printing graphical reports 238
printing textual reports 239
problem determination 453, 454
processor summary report,
example 377
supported hardware report 401
supported hardware, description 237
configuration reports (continued)
Switch Configuration Compare Report 443
switch configuration detail report 391
switch configuration summary report, example 390
switch detail report, example 390
switch report, description 236
switch report, example 389
switch summary report 390
viewing graphical reports 238
VM console report, example 396
VM device detail report, example 395
VM device report, example 395
configuration status 226
CONFIGxx member
build using dialog 228
build with batch 320
compare using dialog 230
connecting
another switch to a switch 181
channel path to switch 179
control unit to switch 180
coupling facility channel paths 119
connections of switches, possibilities 169
connectivity report, example 383
console
changing 158
configuration report, description 236
configuration report, example 394
deleting 158
report, example 396
working with 157
consolidating configuration data 78
context menu 64
control unit
attachment compare report, example 439
attachment data to processor 131
changing 134
changing attachment parameters for multiple processors 135
CNTLUNIT statement 287
compare report, example 440
configuration report, example 384
defining 129
deleting 137
disconnecting 136
migration enhancement 287
priming, serial number 136
supported by system 257
working with 128
control unit list 79
control unit model 263
copying
IODEF, batch example 335
IODEF, batch utility function 323
IODEF, with dialog 43
coupling facility
changing connection 126
connecting channel paths 119
disconnecting channel paths 122
glossary definition 529
overview 9
view information 162
coupling facility connection report 239
coupling over InfiniBand xxii
CPC (central processor complex)
support in HCD 11
CPC (central processor complex)
glossary definition 529
specify name 92
Create Work I/O Definition File panel 37
create work IODF from production IODF 42
creating
graphical configuration report, batch 327
graphical configuration report, dialog 238
IODF using batch 272
IODF using dialog 36
JES3 initialization stream checker data 215
new objects 77
cross operating system support 5
CSS
changing 104
copy 101
copying 103
defining 100
deleting 105
CSS (channel subsystem)
configuration report, description 235
configuration report, example 377
group change action 148
overview 1
CSS-related definitions for devices 142
CSS/operating system device compare report, example 446
CSS/OS report, compare with batch 328
CTC channel 163
CTC Connection List 163
CTC connection report
description 236
diagnostic messages 166
filter 166
incomplete definitions 165
print 166
report, example 396
with MIF 164
with shared channels 164
CTC connections
copy/repeat 103
update 103
CTC side information 165
CU report 239
CUADD, CTC connection 164
customizing HCD 19
data entry panel 60
data sets
allocation for migration 305
cataloged with esoteric name 343
combined 269
element 211
generated during transmit
configuration package 459
migrate with batch 311
data sets (continued)
migration 261
migration using the dialog 269
relationship in HCD 7
data sets due to HCD requests
allocation space 27
data space 30
DCF 239
DCM 117
dddnames
alternate 338
default for migration 305
standard 337
dedicated connection 169, 185
dedicated operation mode 112
default connection 185
default switch matrix 187
defining
a sample configuration 483
configuration packages 50
connections to switches 179
switch configurations 184
switches 172
developing a migration strategy 13
device
activate messages 226
attachment compare report, example 440
changing 146
creating 146
defining CSS-related definitions for a group 148
defining DYNAMIC parameter 149
defining esoterics for a group 148
defining LOCANY parameter 149
defining number 153
defining OS-related definitions for a group 148
defining type/model of a group of devices 150
device migration 294
device, migration 294
developing a migration strategy 13
device list, migration 295
deploying a device 294
devices 150
channel paths 128
channel paths (continued)
channel path definitions 131
deleting 137
migration enhancement 287
priming, serial number 136
supported by system 257
work with 128
details of equipment in HCD 7
dir 7
D
data entry panel 60
data sets
allocation for migration 305
cataloged with esoteric name 343
combined 269
element 211
generated during transmit
configuration package 459
migrate with batch 311
data sets (continued)
migration 261
migration using the dialog 269
relationship in HCD 7
data sets due to HCD requests
allocation space 27
data space 30
DCF 239
DCM 117
dddnames
alternate 338
default for migration 305
standard 337
dedicated connection 169, 185
dedicated operation mode 112
default connection 185
default switch matrix 187
defining
a sample configuration 483
configuration packages 50
connections to switches 179
switch configurations 184
switches 172
developing a migration strategy 13
device
activate messages 226
attachment compare report, example 440
changing 146
creating 146
defining CSS-related definitions for a group 148
defining DYNAMIC parameter 149
defining esoterics for a group 148
defining LOCANY parameter 149
defining number 153
defining OS-related definitions for a group 148
defining type/model of a group of devices 150
device migration 294
device, migration 294
developing a migration strategy 13
device list, migration 295
deploying a device 294
devices 150
channel paths 128
channel paths (continued)
channel path definitions 131
deleting 137
migration enhancement 287
priming, serial number 136
supported by system 257
work with 128
details of equipment in HCD 7
dir 7
D
data entry panel 60
data sets
allocation for migration 305
cataloged with esoteric name 343
combined 269
element 211
generated during transmit
configuration package 459
migrate with batch 311
Index 537
device (continued)
IODF 43
migration enhancement 268
operating system configuration 84
partition, migration 294
partitions 111
priming, serial number 155
priming, volume serial number 155
processors 100
protection 342
restricting partition access 143
summary report, example 387
switch configuration 178
switch-to-switch connection, migration 295
switches 178
switches, confirm 178
working with 138
device candidate list
empty 144
null 144
device group xxv
device list 79
device serial number 141
device type group change 150
diagnostic information and tools 465
diagnostic message, CTC connection 166
diagnostic messages 396
dialog, usage 59
different IODF levels 57
disability 523
disable write protection 206
disabled marker 183
disconnecting
control units from a processor 136
coupling facility channel path connections 122
devices from operating system 153
multiple control units from a processor 136
display active sysplex member list 224
display diagnostic message, CTC connection 166
display graphical reports 242
duplicate control unit 265
duplicate device 267
dynamic channel path management 117
dynamic connection 163, 169
dynamic connection ports 186
DYNAMIC parameter 150
dynamic partition 107
dynamic reconfiguration management activation 219
failure, sysplex-wide 226
glossary definition 529
hardware and software changes allowed 220
identify device 141
overview 4
profile access 339
rejection 222
resource access 339
software only changes allowed 223
sysplex-wide 224
dynamic switch 169
E
editable fields 66
EDT
changing 85
configuration report, description 236
configuration report, example 393
defining 84
deleting 87
EDT statement 277
glossary definition 530
migration enhancement 277
repeating 86
working with 84
EDT statement 277
EMIF
glossary definition, see MIF 530
enable write protection 206
ending HCD 22
enhanced migration 273
entry port, CTC connection 163
entry switch 169
environment for HCD 8
error message, CTC connection 165, 166
errors
diagnosis 447
during IPL 450
during migration 271, 304
during transmit procedure 458
ESCON director, CTC connection 163
ESCON Manager 9
ESCON Manager, glossary definition 530
ESCON, glossary definition 530
esoteric compare report, example 444
esoterics
changing 89
defining 88
deleting 91
device group name 343
glossary definition 530
keyword to allow mixed 27
migration enhancement 277
repeating 90
token 277, 343, 344
token, migration 264
UNITNAME statement 277
working with 88
establishing coupling facility channel path connections 119
establishing dedicated connections 185
EXEC procedures 34
exit, function key 69
explicit device candidate list 110, 143
export
IODF, batch 333
IODF, dialog 45
EXPORTED_HLQ 27
extended migration 273
I/OCP input data set 210
F
fabric 169
failure, diagnosis 447
fast path 70
FCP device data 216
FCP SAN configuration template file 216
FCTC 163
fibre channel xxx
FICON
cascade switching 169
switch fabric 169
FICON Channel-to-Channel support 163
FICON glossary definition 530
FICON switch
downgrading link address 133
upgrading link address 133
FICON switches 169
fields
editable 66
input and output 61
promptable 67
filter CTC connection 166
filter mode 75
filtering
occupied ports xxiii
filtering a list 73
FIND command 186
font for graphical reports 29
formatting type for graphical reports 28
four-digit device numbers 138
free space in HSA 225
function keys
actions 59
add 77
backward 60
command 68
Enter 69
exit 69
forward 60
help 68
how to use 68
instruct 60
jump 242
left 60
previous 69
problem determination 451
prompt 61
reset 66
right 60
setting 24 keys on 22
split 345
zoom 243
function parameter strings 308
G
GDDM
installation 19
usage for reports 239
generate switch matrix 187
generic device type, glossary definition 530
generics 87
migration enhancement 278
UNITNAME statement 278
H

hardware and software changes allowed 220
hardware configuration token 225
Hardware Management Console 11
glossary definition 530
hardware report 401
hardware support 253
HCA 118
HCD (Hardware Configuration Definition)
    abnormal termination 448
    and coupling facility 9
    and I/O Operations (ESCON Manager) 9
    customizing 19
    data sets relationship 7
differences between HCD and MVSCP/IOCP 1
ending 22
environment 8
installation considerations 19
    object management services 475
overview 1
overview of functions 4
product identifier 447
setting up 19
starting 22
HCD LDAP backend 347
HCD LDAP backend configuration 356
HCD LDAP schema 355
HCD profile 23, 211
HCD Profile Options dialog 24
HCM change log data set allocation space extension 27
HCM master configuration file glossary definition 531
HCM MCF volume serial number 26
HCPRIO input data set
    build with batch 319
    build with dialog 215
glossary definition 530
migration 267
help
    how to get 68
    problem determination 452
testing panels 452
HELP command 452
HELPTEST command 452
hiding parameter/feature definitions 154
hierarchical navigation 78
highlighting 61
HLQ
    for exporting IODFs 27
HOM services 475
    example 480
    functions 478
    input and output areas 476
    invocation 475
    request block (HRB) 476
    return codes 481
host communication adapter 118
HSA 225
HSA token 205
I

I/O cluster
glossary definition 530
I/O concurrency level 110, 132, 134, 386, 387, 499
I/O configuration definition reference configuration report, example 431
description 237
print 324
I/O Operations
glossary definition 530
I/O Operations, overview 9
I/O path list 217
I/O path report
description 236
example 397
IBM ES/390
    build IOPC input data set 208
IBM Tivoli Directory Server for z/OS 347
    address space 463
ID statement 281
identifying problems 447
IEC161I (copy IODF batch utility) 336
IEFESOJL (catalog scan utility) 344
image number, CTC connection 164
import
    IODF, batch 332
    IODF, dialog 48
incomplete CTC definitions 165
increased system availability 4
incremental update 292
initialization problems 456
initialization stream checker data for JES3 215
initialize IODF from VSAM DIV file 309
input fields 61
input parameter string, batch utilities 308
input to stand-alone IOPC 208
installation-static 263
installed port range 174
installed ports 183
installed UIMs 259
installing
    for migration 16
    GDDM 19
    HCD 19
INSTRUCT command 60
instruction area 60
insufficient data set sizes 305
interface
    batch utility functions 307
dialog 59
    HCD object management services 475
introduction 1
invoking batch utility functions 307
IOCDS
    build S/390 microprocessor IOCDSs 203
    build with batch 316
    build with dialog 201
glossary definition 530
report, example 379
token 205
use with HCD 5
write in preparation for processor upgrade 202
IOCONFIG statement 276
IOCPS
    build input data set with batch 316
    build input data set with dialog 208
    example 211
    extended migration 210
glossary definition 530
migrating additional data sets 265
migrating input data sets 261
stand-alone 208
use with HCD 1
IODEVICE statement 288
IODF
activity log 56
available IODFs 63
change log 56
glossary definition 530
multi-user access status 63
V5 IODF 44, 49
validated work IODF 214
IODF (I/O definition file)
    activity logging 37, 54
    attributes 36
backup 42
build production IODF, batch 314
build production IODF, dialog 196
build work IODF, batch 316
change attributes 44
change during session 42
compare 246
compare reports, example 436
compare with batch 328
copy with batch 323
copy with dialog 43
data model 505

Index 539
showing parameter/feature definitions 154
SHOWIODF command 45
SHOWMSG ACTIVATE command 222
SHOWMSG command 466
single IODF 39
single point of control 4, 224
single selection lists 62
single switch activation 191
SMP processor
glossary definition 532
term definition xxvii, 91
SMS considerations 344
SNA address 93
software only changes allowed 223
solving migration errors 302
space allocation 37
spanned channel path xxviii
spanned channel paths 112
spanned operation mode 112
specifying input data sets for migration 269
IODF, dialog 36
split screen 345
SS (subchannel set) 142
SS-related definitions for devices 142
stand-alone CTC adapter 163
stand-alone IOCP 208
standard DD names 337
starting
HCD 22
migration 269
the dialog, batch 308
status of configuration 226
STP
glossary definition 532
STP link 122
STP link support xxviii
strategy, migration 13
subchannel set 101, 142
glossary definition 532
support level 92
online description 93
provided functions 93
supported hardware 253
supported hardware report 401
supported migration paths 14
surrogate user ID for exporting IODFs 47
SWCONF statement 280
switch
blocking ports 185
changing data 175
changing data 175
compare report, example 442
confirming delete 178
collections to switches 178
dedicated connection 169
default connection 185
defining 172
defining connections 179
deleting 178
disabled marker 183
dynamic connection 169
dynamic switch 169
entry switch 169
graphical report 239

Q
query supported hardware 253

R
RACF 339, 340, 342
RACF backend SDBM 347
re-migrate IOCP input data sets 210
read access 340
reason code 448
receiving an IODF 48
reconfigurable operation mode 112
REFRESH command, graphical report 244
refreshing a view on IODFs 158
refreshing the active sysplex member list 225
rejection of dynamic activation 222
relationship of data sets 7
release level compatibility 57
repair an IODF 472
repeating action 78
channel subsystem 101
description 78
EDTs 86
esoterics 90
operating system configuration 83
partitions 108
processors 96
replacing during build production IODF 196
replacing objects through migration 292
reporting problems 473
reports, text, options 28
requirements for setting up HCD 19
for using HCD 8
migrating IOCP/MVSCP data 262
reserved partition 107
reset function key 66
reset source configuration sysplex-wide 227
RESOURCE statement 282
restricting partition access for devices 143
resume activation sysplex-wide 227
RIGHT command 60
roadmap, migration 14
S
S/390 microprocessor
build IOCDs 203
manage IPL attributes 207
support in HCD 11
sample configuration 483
save messages 67
switch configuration 192
SAVE command 243, 245
SAVE command, CTC connection 166
saved switch file 189
scaling factor for graphical reports 28
scenario
multi-user access 158
scenarios 483
schema file for LDAP 355
schema.hcd.ldif 355
scrolling 60
SCTC 163
SDBM
RACF backend 347
search order at IPL 233
searching data bases 473
security considerations 339, 342
selection markers 64
selection methods in lists 61
sending an IODF 45
sensing 11, 397
sequence of migration 261
sequence to define a configuration 80
serial number
control unit 136
device 155
processor 99
switch 175
serial number for device 141
server started task log 466
Server Time Protocol
glossary definition 532
Server Time Protocol (STP) link 122
Server Time Protocol link support xxviii
setting up HCD 19
setup considerations for migration 16
severity of messages 167
shared channels, CTC connection 164
shared operation mode 112
sharing IODFs 39, 57, 343
shortcut keys 523
showing parameter/feature definitions 154
SHOWIODF command 45
SHOWMSG ACTIVATE command 222
SHOWMSG command 466
single IODF 39
single point of control 4, 224
single selection lists 62
single switch activation 191
SMP processor
glossary definition 532
term definition xxvii, 91
SMS considerations 344
SNA address 93
software only changes allowed 223
solving migration errors 302
space allocation 37
spanned channel path xxviii
spanned channel paths 112
spanned operation mode 112
specifying input data sets for migration 269
IODF, dialog 36
split screen 345
SS (subchannel set) 142
SS-related definitions for devices 142
stand-alone CTC adapter 163
stand-alone IOCP 208
standard DD names 337
starting
HCD 22
migration 269
the dialog, batch 308
status of configuration 226
STP
glossary definition 532
STP link 122
STP link support xxviii
strategy, migration 13
subchannel set 101, 142
glossary definition 532
support level 92
online description 93
provided functions 93
supported hardware 253
supported hardware report 401
supported migration paths 14
surrogate user ID for exporting IODFs 47
SWCONF statement 280
switch
blocking ports 185
changing data 175
changing data 175
compare report, example 442
confirming delete 178
collections to switches 178
dedicated connection 169
default connection 185
defining 172
defining connections 179
deleting 178
disabled marker 183
dynamic connection 169
dynamic switch 169
entry switch 169
graphical report 239

542 z/OS V1R11.0 HCD User's Guide
### Index

<table>
<thead>
<tr>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>187</td>
</tr>
<tr>
<td>189</td>
</tr>
<tr>
<td>279</td>
</tr>
<tr>
<td>176</td>
</tr>
<tr>
<td>169</td>
</tr>
<tr>
<td>174</td>
</tr>
<tr>
<td>169</td>
</tr>
<tr>
<td>175</td>
</tr>
<tr>
<td>389</td>
</tr>
<tr>
<td>256</td>
</tr>
<tr>
<td>279</td>
</tr>
<tr>
<td>169</td>
</tr>
<tr>
<td>174</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>175</td>
</tr>
<tr>
<td>389</td>
</tr>
<tr>
<td>256</td>
</tr>
<tr>
<td>280</td>
</tr>
<tr>
<td>191</td>
</tr>
<tr>
<td>175</td>
</tr>
<tr>
<td>186</td>
</tr>
<tr>
<td>443</td>
</tr>
<tr>
<td>183</td>
</tr>
<tr>
<td>189</td>
</tr>
<tr>
<td>280</td>
</tr>
<tr>
<td>172</td>
</tr>
<tr>
<td>169</td>
</tr>
<tr>
<td>191</td>
</tr>
<tr>
<td>175</td>
</tr>
<tr>
<td>184</td>
</tr>
<tr>
<td>188</td>
</tr>
<tr>
<td>187</td>
</tr>
<tr>
<td>280</td>
</tr>
<tr>
<td>228</td>
</tr>
<tr>
<td>224</td>
</tr>
<tr>
<td>225</td>
</tr>
<tr>
<td>224</td>
</tr>
<tr>
<td>175</td>
</tr>
<tr>
<td>176</td>
</tr>
<tr>
<td>162</td>
</tr>
<tr>
<td>169</td>
</tr>
<tr>
<td>169</td>
</tr>
<tr>
<td>169</td>
</tr>
<tr>
<td>181</td>
</tr>
<tr>
<td>279</td>
</tr>
<tr>
<td>447</td>
</tr>
<tr>
<td>449</td>
</tr>
<tr>
<td>344</td>
</tr>
<tr>
<td>224</td>
</tr>
<tr>
<td>225</td>
</tr>
<tr>
<td>224</td>
</tr>
<tr>
<td>532</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>448</td>
</tr>
<tr>
<td>449</td>
</tr>
<tr>
<td>224</td>
</tr>
<tr>
<td>456</td>
</tr>
<tr>
<td>271</td>
</tr>
<tr>
<td>457</td>
</tr>
<tr>
<td>458</td>
</tr>
<tr>
<td>235</td>
</tr>
<tr>
<td>377</td>
</tr>
<tr>
<td>122</td>
</tr>
<tr>
<td>60</td>
</tr>
<tr>
<td>209</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>277</td>
</tr>
<tr>
<td>60</td>
</tr>
<tr>
<td>469</td>
</tr>
<tr>
<td>471</td>
</tr>
<tr>
<td>471</td>
</tr>
<tr>
<td>467</td>
</tr>
<tr>
<td>109</td>
</tr>
<tr>
<td>53</td>
</tr>
<tr>
<td>347</td>
</tr>
<tr>
<td>340</td>
</tr>
<tr>
<td>103</td>
</tr>
<tr>
<td>292</td>
</tr>
<tr>
<td>310</td>
</tr>
<tr>
<td>48</td>
</tr>
<tr>
<td>295</td>
</tr>
<tr>
<td>61</td>
</tr>
<tr>
<td>307</td>
</tr>
<tr>
<td>533</td>
</tr>
<tr>
<td>28</td>
</tr>
<tr>
<td>259</td>
</tr>
<tr>
<td>457</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>45</td>
</tr>
<tr>
<td>162</td>
</tr>
<tr>
<td>215</td>
</tr>
<tr>
<td>345</td>
</tr>
<tr>
<td>390</td>
</tr>
<tr>
<td>183</td>
</tr>
<tr>
<td>132</td>
</tr>
<tr>
<td>217</td>
</tr>
<tr>
<td>217</td>
</tr>
<tr>
<td>217</td>
</tr>
<tr>
<td>217</td>
</tr>
<tr>
<td>217</td>
</tr>
<tr>
<td>214</td>
</tr>
<tr>
<td>217</td>
</tr>
<tr>
<td>543</td>
</tr>
</tbody>
</table>
X
XMP processor
  glossary definition 533
  term definition xvii, 91

Z
z9 EC processors xxiv
zooming graphical reports 243
Readers’ Comments — We’d Like to Hear from You

z/OS
Hardware Configuration Definition
User’s Guide
Version 1 Release 11

Publication No. SC33-7988-08

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.
Submit your comments using one of these channels:
- Send your comments to the address on the reverse side of this form.
- Send a fax to the following number: FAX (Germany): 07031+16-3456
  FAX (Other Countries): (+49)+7031-16-3456
- Send your comments via e-mail to: s390id@de.ibm.com

If you would like a response from IBM, please fill in the following information:

Name

Address

Company or Organization

Phone No.

E-mail address