z/VM
7.3

System Operation

IBM

SC24-6326-73

Note:

Before you use this information and the product it supports, read the information in “Notices” on page 275.

This edition applies to version 7, release 3 of IBM® z/VM® (product number 5741-A09) and to all subsequent releases and modifications until otherwise indicated in new editions.

Last updated: 2022-09-07

Contents

Chapter 10. Performing Data Management Services..211

Chapter 11. Running Utility Programs...233

Figures

Tables

About This Document

This document describes how to operate the IBM® z/VM® program. It provides information about the following tasks:

- Bringing up the z/VM system, including the production system and service virtual machines
- Running the z/VM system
- Controlling local devices
- Responding to z/VM errors
- Shutting down the z/VM system, including the production system and service virtual machines
- Collecting information about the system
- Performing data management services
- Running utility programs

Intended Audience

This document is intended for anyone who operates a z/VM system on a real processor. This document assumes that you understand the hardware controls and features of your installation. You should also have a general knowledge of basic virtual machine concepts, and you must have a basic knowledge of the production operating system you will be running in a virtual machine.

Where to Find More Information

For information about operating virtual machines, see *z/VM: Virtual Machine Operation*. For more information about z/VM functions, see the other documents listed in the "Bibliography" on page 279.

Links to Other Documents and Websites

The PDF version of this document contains links to other documents and websites. A link from this document to another document works only when both documents are in the same directory or database, and a link to a website works only if you have access to the Internet. A document link is to a specific edition. If a new edition of a linked document has been published since the publication of this document, the linked document might not be the latest edition.

How to Send Your Comments to IBM

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

To send us your comments, go to z/VM Reader's Comment Form (https://www.ibm.com/systems/campaignmail/z/zvm/zvm-comments) and complete the form.

If You Have a Technical Problem

Do not use the feedback method. Instead, do one of the following:

- Contact your IBM service representative.
- Contact IBM technical support.
- See IBM: z/VM Support Resources (https://www.ibm.com/vm/service).
- Go to IBM Support Portal (https://www.ibm.com/support/entry/portal/Overview).

Summary of Changes for z/VM: System Operation

This information includes terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations for the current edition are indicated by a vertical line (|) to the left of the change.

SC24-6326-73, z/VM 7.3 (September 2022)

This edition supports the general availability of z/VM 7.3. Note that the publication number suffix (-73) indicates the z/VM release to which this edition applies.

SC24-6326-03, z/VM 7.2 (May 2022)

This edition includes changes to support product changes provided or announced after the general availability of z/VM 7.2.

Miscellaneous updates for May 2022

References to IBM Z® Application Assist Processor (zAAP), which is not supported on IBM z13 and later models, are removed. The following topics are updated:

- “Finding Out about the Load on the System” on page 181
- “Finding Out a User's Scheduling Share” on page 62

SC24-6326-03, z/VM 7.2 (March 2021)

This edition includes changes to support product changes provided or announced after the general availability of z/VM 7.2.

[VM66173] 4 TB Real Memory Support

z/VM APAR VM66173 delivers support for up to 4 TB of real memory, allowing z/VM systems to address a full 4 TB of first-level (real) memory, doubling the previous supported limit of 2 TB. With advanced memory management capabilities available in the z/VM product, clients now have the ability to run workloads that exceed 4 TB of virtual memory across all hosted guest systems, depending on workload characteristics. In conjunction with z/VM support for 80 processors, IBM Z and LinuxOne servers can now host even more work in a single z/VM partition, or across multiple z/VM partitions on one system. APAR VM66173 also delivers various system command updates, such as automatic STANDBY memory for guests and an enhancement to the CP DEFINE STORAGE command.

With automatic STANDBY memory for guests, a system administrator can code a more generic DEFINE STORAGE command that does not need to be updated every time the guest's directory entry storage size changes.

The following information has been updated for this support:

- “Passing IPL Parameters” on page 13

SC24-6326-02, z/VM 7.2 (September 2020)

This edition includes changes to support the general availability of z/VM 7.2.

MSS Multi-Target PPRC Exploitation

z/VM supports now up to four subchannel sets that reflect the actual hardware configuration and are commonly used in a MSS Multi-Target PPRC environment. Logically z/VM groups all recognized devices into two device groups, with one being the active configuration and the other being the stand-by configuration. Most z/VM commands support only 4-digit device numbers and can refer only to devices in the active configuration. Support has been expanded to handle the real subchannel sets and 5-digit device numbers where the leading fifth digit reflects the hardware subchannel number.

Some operands have been updated for "Passing IPL Parameters" on page 13.

Removal of PAGING63 IPL Parameters

The PAGING63 IPL parameter is removed since this IPL parameter blocks use of newer paging technologies (for example, Encrypted, EAV, HyperPAV, and HPF Paging) and has not been recommended for use since z/VM 6.4.

The following IPL parameter has been deleted:

- PAGING63

SC24-6326-01, z/VM 7.1 (January 2020)

This edition includes changes to support product changes provided or announced after the general availability of z/VM 7.1.

Improved information about defining crypto resources to z/VM guests

A new chapter has been added to *z/VM: CP Planning and Administration* to provide improved information about how to define crypto resources to z/VM guests. See *Crypto Planning and Management* for more information.

Crypto terminology in z/VM has changed to eliminate confusion.

SC24-6326-00, z/VM 7.1 (September 2018)

This edition includes changes to support the general availability of z/VM 7.1.

Improvements to the z/VM Dump Process

Dump processing has been enhanced to reduce the time and space required to create, process, and transmit hard abend and snap dumps. The two largest dump components, the CP frame table and user PGMBKs, have been omitted for the majority of abends but can be included if requested by the SET DUMP or SNAPDUMP command. Excluding these components results in dumps that are considerably smaller, thus requiring less space in both the system SPOOL and CMS file system. Also, support to write hard abend and snap dumps to tape has been removed.

Chapter 1. Introduction

This chapter introduces you to z/VM. After reading this chapter, you can:

- Understand the role of system operation
- Describe the kinds of virtual machines you can run
- Describe the consoles you can use and what you will use them for
- Locate information about using a display under z/VM, using the hardware console for your processor complex, and using the commands for the production system you run
- Understand the major tasks involved in real system operation.

Role of System Operation

As the system operator, your job is to provide system resources for z/VM users and for guests running on z/VM, and to maintain the z/VM operating system.

z/VM Operations Environment

Unless otherwise specified, this book assumes you operate z/VM as follows:

- You operate z/VM on a real processor rather than in a virtual machine. This is also called running a "native" z/VM system.

  For information about the supported processors, see *z/VM: General Information*.

- You also operate a production system in a virtual machine. The production system may be any one of the following operating systems:

  - z/OS® (MVS™)
  - Linux®
  - z/VM
  - z/VSE®
  - z/TPF™

  Note: For additional information about running a production system, see *z/VM: Running Guest Operating Systems* or *z/VM: Getting Started with Linux on IBM Z*.

- The z/VM system also has several test and several service virtual machines running.

  This book assumes that, as a system operator, you are not responsible for running test systems in virtual machines. However, you may run one or more service virtual machines, such as:

  - Accounting virtual machine
  - Error recording (EREP) virtual machine
  - Symptom record recording virtual machine
  - RSCS virtual machine
  - VM/Pass-Through virtual machine
  - TCP/IP
  - Directory Maintenance Facility
  - Virtual Machine Resource Manager (VMRM)
  - Performance Toolkit for z/VM™

- As the operator, you use CP most of the time. However, some tasks may require you to work in the CMS environment.

- You are *not* running a programmable operator.
- You are *not* using RACF®. For information about running RACF, see *z/VM: RACF Security Server Security Administrator's Guide*.
- Production systems use z/VM's reserve/release support to share some of their DASD volumes with other systems, which run either in other virtual machines or on other processor complexes. For example, if you are running a z/OS production system, it may share some of its DASD volumes with z/OS test virtual machines on the same processor complex, or with a z/OS system running native on another processor complex, or both.

  This affects the way you control DASDs.

Virtual Machines You Run

As the z/VM operator, you run several kinds of virtual machines. Although there are probably as many ways to divide operator responsibilities as there are installations, this book assumes that you run the following kinds of virtual machines:

- z/VM operator's virtual machine
- Production system in a virtual machine
- Service virtual machines.

Running the z/VM Operator's Virtual Machine

When you bring up the z/VM system, CP automatically logs on the primary system operator at the z/VM primary system console. The primary system console is a display. Your installation defines its device number on the RDEVICE statement in the system configuration file.

Your installation defines the primary system operator's user ID on the SYSTEM_USERIDS statement in the system configuration file. Your installation may also have defined other user IDs for you to use when you perform special operations tasks. For example, an installation often has a spooling operator who controls the installation's printers. If you are the spooling operator, you often use another user ID at another console.

However, the primary system operator has special privileges. If you are responsible for controlling z/VM, you usually use the user ID of the primary system operator.

In addition to having special CP command privileges, the primary system operator's virtual machine has default settings that differ from those of other users. For example, the primary system operator's default setting for the TERMINAL MODE command is CP rather than VM. And the initial setting of the SET RUN command is ON rather than OFF.

Otherwise, the primary system operator's virtual machine is the same as any other virtual machine.

Running a Production System in a Virtual Machine

To run a production system in a virtual machine, you must:

1. Log on to z/VM using the production system's user ID.

   This creates the production system's virtual machine and gives it virtual storage, processors, and I/O devices to use as defined in the system directory.

   In many cases, the production system virtual machine is autologged and has a secondary operator definition (usually OPERATOR). This can free a display for other use and can help avoid accidental CP READ situations or logging off the production system.
2. Use CP commands to set up any special running environment the production system's virtual machine requires.
3. Load (IPL) the production operating system in the virtual machine.

Note: During the installation of z/VM, the profile exec or the user directory may have been customized to have the system automatically IPLed.

Once the production system comes up, running in a virtual machine is the same as running on a real processor, with few exceptions. In response to your commands, the production system performs the same functions in a virtual machine as it does on a real machine. It manages virtual machine storage, processors, and I/O devices the same way it manages real storage, processors, and I/O devices.

The difference between running a production system on a real processor and running it in a virtual machine is not the way you communicate with it. The difference is in the way you control the resources available to the operating system. A virtual machine has resources provided by CP. To stop, start, add, detach, or change virtual machine resources, you must communicate with CP.

Running Service Virtual Machines

Depending on what facilities, features, and products are enabled on your system, you may need to log on to other service virtual machines. For example, if accounting, error recording (EREP) and symptom record recording is active on z/VM, you need to log on to the accounting, error recording, and symptom record recording virtual machines. From these virtual machines, you enter CP and CMS commands to create accounting, EREP, and symptom record reports.

If you are using any of the following z/VM facilities and optional features, you must operate the service virtual machines they require:

- Directory Maintenance Facility for z/VM
- Performance Toolkit for z/VM
- RSCS Networking for z/VM
- TCP/IP for z/VM

For information about how to operate the virtual machines for z/VM facilities and features, see the appropriate publications listed in the “Bibliography” on page 279.

Some licensed programs may require you to operate service virtual machines. For example, if your installation has installed any of the following licensed programs, you must operate the service virtual machines they require:

- VM/Pass-Through Facility
- VTAM® Communications Network Applications (VCNA)
- VM/Virtual Telecommunications Access Method (VM/VTAM).

For information about when and how to stop these service virtual machines, see the appropriate licensed program documentation.

Finally, each time you bring up z/VM, CP automatically logs on a virtual machine called AUTOLOG1. The AUTOLOG1 userid is the default. You can change the AUTOLOG1 userid by using the STARTUP operand on the SYSTEM_USERIDS statement in the system configuration file. Your installation may use this service virtual machine to perform some system initialization tasks automatically. For example, AUTOLOG1 or the userid you specified on the SYSTEM_USERIDS statement in the system configuration file, can start up other virtual machines or dedicate devices to them.

Consoles You Use

As the z/VM operator, you can use several consoles. Consoles can be located on the same physical device (workstation, PC, display) or on separate devices. Because of the flexibility of z/VM, the number of consoles varies by customer installation. Depending on your installation, there might be one production system or many production systems. There can be hundreds of virtual machines, each running a production system. Regardless of the number of consoles, the following are the four types of consoles available and a description of each:

- Hardware Management Console
- Primary system operator console
- Virtual Console for each virtual machine

- Production system's system console when there is a production system running in a virtual machine.

Note: Although the number of consoles at each site depends on the installation, each installation has at least one Hardware Management Console, at least one primary system operator console, and one or more virtual consoles to control virtual machines. A production system often has its own system console.

Using the Hardware Management Console

A Hardware Management Console (HMC) communicates with each central processor complex (CPC). There can be one or more Hardware Management Consoles depending on your site's installation. This section describes only the specific tasks for the z/VM operator. For additional information on using the Hardware Management Console, see the processor complex hardware publications.

From the Hardware Management Console, you can:

- IPL (load) z/VM
- Monitor messages and send commands to z/VM (Operating System Messages panel)
- Perform basic z/VM system management functions
- Install, service, and operate z/VM using the integrated 3270 console.

Using the Operating System Messages Panel

From the Hardware Management Console, you can use the Operating System Messages task (panel) which is the SYSC device, to view z/VM messages. No setup is required to view the messages. You can also use the Operating System Messages panel as the system operator console to send commands z/VM. This communication between the Hardware Management Console and z/VM will be in line mode.

To use the Operating System Messages panel as the system operator console do one of the following:

- Specify the **CONS=SYSC** IPL parameter when using the Stand-Alone Program Loader (SAPL). See "Passing IPL Parameters" on page 13.
- Specify a load parameter of **CONSSYSC** to bypass the SAPL screen and have the z/VM operator come up on the Operator System Messages panel. See "Overriding Stand-Alone Program Loader Defaults" on page 10.
- Add **SYSTEM_CONSOLE** to the OPERATOR_CONSOLES system configuration statement. See *z/VM: CP Planning and Administration*.

To have emergency operating system message displayed, do the following:

- Add **SYSTEM_CONSOLE** to the EMERGENCY_MESSAGE_CONSOLES system configuration statement.

Performing basic z/VM system management functions

z/VM provides enabling interfaces that allow communication between the HMC and the operating system, allowing you to perform basic system management through the HMC. No additional configuration or products are required for this system management function.

For details on this system management function and how to use it, see the documentation for your HMC.

Setting Up and Using the Integrated 3270 Console

From the Hardware Management Console you can use the integrated 3270 console as the system operator's console. The integrated 3270 console is identified by z/VM as SYSG. This console provides you the ability to install, service, and operate z/VM without any attached 3270 devices.

To identify the integrated 3270 console to CP, do one of the following:

- Use the **CONS=SYSG** IPL parameter when using the Stand-Alone Program Loader (SAPL). See "Passing IPL Parameters" on page 13.
- Specify a load parameter of **SYSG** to bypass the SAPL screen and have the z/VM operator come up on the Integrated 3270 Console. See "Overriding Stand-Alone Program Loader Defaults" on page 10.

- Add **SYSTEM_3270** to the OPERATOR_CONSOLES system configuration statement. See *z/VM: CP Planning and Administration*.

To start the integrated 3270 console from the Hardware Management Console

From the Hardware Management Console, do the following to start an integrated 3270 console on the image (logical partition) where z/VM will be installed:

1. In the left panel under Systems Management, select the CEC containing your LPAR.
2. In the right panel, find your CPC Image (Logical Partition) and select it by checking the "select" box in the leftmost column.
3. Using either the pop-up menu obtained by clicking the boxed double-arrow next to the image name, or the lower right panel menu, select Recovery, then Integrated 3270 Console.
4. A 3270 full screen window should be displayed for the selected image.

To use the Load task on the Hardware Management Console to IPL z/VM

1. In the left panel under Systems Management, select the CEC containing your LPAR.
2. In the right panel, find your CPC Image (Logical Partition) and select it by checking the "select" box in the leftmost column.
3. Using either the pop-up menu obtained by clicking the boxed double-arrow next to the image name, or the lower right panel menu, select Recovery, then Load.
4. Enter: **SYSG** in the 'Load parameter' entry field. When the load task completes, the SAPL screen will appear on the integrated 3270 console. See Figure 1 on page 5.
5. Enter **CONS=SYSG** in the IPL parameters. This identify the integrated 3270 console as the system operator console.
6. From the integrated 3270 console, press F10 to load the specified z/VM system. The z/VM system should load and display startup messages.

```
 STAND ALONE PROGRAM LOADER: z/VM VERSION n RELEASE n.n

 DEVICE NUMBER:   01B8       MINIDISK OFFSET:   35         EXTENT:  -

 MODULE NAME:     CPLOAD     LOAD ORIGIN:       1000

 --------------------------------IPL PARAMETERS--------------------------------
cons=sysg


 -----------------------------------COMMENTS-----------------------------------
Secondary parm disks can be found on 1B8 at offset 185 and on 18A at offsets
35 and 185.


 ------------------------------------------------------------------------------



 9= FILELIST  10= LOAD  11= TOGGLE EXTENT/OFFSET
```

Figure 1. Stand-Alone Program Loader

Using the Primary System Operator Console

The primary system operator console is the console where CP automatically logs on the primary system operator virtual machine during IPL. This can be the Hardware Management Console, or a device specified during the IPL. If SYSC is specified, the Operating System Messages panel becomes the system operator console. If SYSG is specified, the integrated 3270 console becomes the system operator console. Your installation chooses what device or what Hardware Management Console function (SYSC or SYSG) is identified as the primary system operator console using one of the following:

- System configuration statement, OPERATOR_CONSOLES

- IPL parameter, CONS=*xxxx*
- Load parameter CONSxxxx

The primary system operator console is where you perform most of the tasks in this book. For example, you enter CP commands from the primary system console to perform the following tasks:

- Bringing up the z/VM CP
- Shutting down the z/VM CP
- Controlling z/VM devices
- Communicating with z/VM users
- Responding to z/VM errors
- Collecting information about z/VM operation.

After CP logs on the primary system operator, it controls the format of the primary system console's screen. If you are unfamiliar with this format, see the *z/VM: Virtual Machine Operation*.

Using Virtual Consoles for Other Virtual Machines

Other consoles you use are the virtual consoles for the virtual machines from which you create and initialize work or production systems. You use these consoles when:

- You log on a production system's virtual machine
- You set up any special running environment it requires
- You use the CP IPL command to load production systems in the virtual machine
- You need to recover a production system.

You use these consoles to control a production system's virtual machine. To control the production system itself, use the production system's system console.

When you log on a production system's virtual machine at this console, CP controls the format of the console's screen. If you are unfamiliar with this format, see the *z/VM: Virtual Machine Operation*.

Note: For z/OS guests, when the system console is responding to z/VM errors or IPLing the system, the virtual machine console can be used as an IPL and error-recording console. For more information see the *z/VM: Virtual Machine Operation* and the *z/VM: Running Guest Operating Systems*.

Using a Production System's System Console

When there is another operating system running work in a virtual machine, this is referred to as a production system. You use the production system's system console to control the work, just as if you were running the production system directly on the hardware. Some production systems have their system console on the same physical device as the virtual machine operator's console; other installations might use separate devices.

Operating z/VM

To operate z/VM, you must be able to:

- Bring up the z/VM system

  To bring up the z/VM system, you must be able to:

  – Bring up the z/VM Control Program (CP) on the real processor
  – Start service virtual machines to provide system services
  – Bring up the production system in a virtual machine
  – Check the initialization of system resources
  – Prepare the system for a possible failure
  – Respond to errors during bring-up, if necessary.

- Run the z/VM system

  To run the z/VM system, you must be able to:

  - Check the status of the system
  - Watch the system for changes and requests
  - Control local devices and other hardware
  - Control the system's service virtual machines
  - Communicate with users.
- Respond to z/VM errors
- Shut down the z/VM system, including the production system and the service virtual machines
- Collect information about the system, including information about:
  - System events
  - Performance
  - Accounting
  - Hardware errors.
- Perform data management services
- Run utility programs.

The chapters that follow tell you how to use z/VM to perform these operation tasks.

Chapter 2. Using the Stand-Alone Program Loader

The Stand-Alone Program Loader (SAPL) is a generic loader that loads stand-alone programs stored in CMS module form on a CMS minidisk. Using SAPL, you can use any stand-alone program that can be generated in the form of a CMS module.

Examples of such programs are:

- CP nucleus module (CPLOAD)
- Stand-Alone Program Loader Creation Utility (SALIPL)
- DASD Dump Restore Utility (DDR)
- Device Support Facilities Program (ICKDSF)
- IBM I/O Exerciser for System z® (ESAIO)

This chapter tells you how to use SAPL to load a stand-alone program that is generated in the form of a module.

This chapter also describes the IPL parameters supported by CP.

During the IPL process, various information is read from DASD devices. Here are some terms that will be used to refer to these devices:

- The **IPL device** is the device which is specified on the load screen and from which SAPL is loaded.
- The **SYSRES device** is the device from which SAPL loads the CP module.
- The **parm disk device** is the device on which the CP module finds the system configuration file.
- Other DASD devices are used but are not of interest for this discussion. They include the DASD devices used for the CP directory, warmstart, checkpoint, paging, spooling, and user minidisks.

Creating the Stand-Alone Program Loader

Before a DASD device can be IPLed, SAPL must be written to the device using the Stand-Alone Loader Creation Utility (SALIPL). For information on how you can write SAPL to the IPL device by using SALIPL from the CMS command line, see *z/VM: CP Commands and Utilities Reference*. For information on how to write SAPL to the IPL device by using SALIPL on the real processor, see “Running the Stand-Alone Program Loader Creation Utility (SALIPL)” on page 233.

Important Note!

As of z/VM V5.1, the number of blocks on FBA DASD (SCSI or not) that SALIPL uses for SAPL has increased. SALIPL writes to blocks 5-207 on CP-formatted DASD. In previous releases, SALIPL wrote to blocks 5-31. You must ensure that no other functions, such as CP directory, warmstart, checkpoint, paging, spooling, or minidisks, are using the area to which SALIPL writes.

On a CMS minidisk with a RECOMP area, SALIPL writes SAPL to blocks 5-207 of the RECOMP area. In previous releases, blocks 5-31 of the RECOMP area were used. In order to accommodate the larger size of SAPL, you may need to increase the size of the RECOMP area on the CMS minidisk.

The following data is in the blocks that SALIPL writes out:

Blocks 5-196	SAPL program
Blocks 197-199	Reserved for IBM Use
Blocks 200-207	SCSI boot records

When SALIPL writes SAPL to the IPL device, defaults for the following attributes that SAPL uses are established:

- The location of the CMS minidisk where modules can be found
- The name of the module that is to be loaded by SAPL
- The location in storage at which the module will be loaded, if it is relocatable
- The IPL parameters that are passed to the program being loaded
- Operator instructions that are displayed on the SAPL screen.

These defaults can be overridden as described below.

The default module is CP (default name CPLOAD), but it may be a stand-alone program in CMS module form. You can have many CP modules in your installation, allowing for different conditions or providing backup. You select the CP module to be used:

- When you run Stand-Alone Program Loader Creation Utility (SALIPL)
- During IPL by overriding the SAPL defaults on the SAPL screen
- During SHUTDOWN REIPL by using the MODULE operand.

Although the file name of a CP module is variable, the file type must be MODULE. CP modules typically reside on a minidisk that is identified as a parm disk in the volume allocation table, but you can select the DASD volume and offset/extent from which SAPL will read. Therefore, you can have CP modules on multiple minidisks on multiple DASDs.

Note: Although SAPL will accept a load origin when loading a CP module, it has no effect. CP will always relocate itself to location X'2000'.

Overriding the Console that CP Will Use

If you only need to override the console that CP will use as the system operator console during CP initialization, specify a load parameter of CONS*xxxx*. For example:

- Specify CONSSYSG to use the integrated 3270 console on the IBM Hardware Management Console (HMC)
- Specify CONSSYSC to use the operating system messages panel on the HMC
- Specify CONS*xxxx* to cause CP to use the 3270 at address *xxxx*.

If you only need to override the system configuration file that CP will use during system initialization, specify a load parameter of FN*xxxxxx*. For example, specifying FNSYS001 on the HMC will cause CP to search for a system configuration file to use at initialization with a file name of SYS001. The "*xxxxxx*" specifies a file name and must be between 1 and 6 characters in length. The file type will be CONFIG unless the FT= IPL parameter that was specified when SAPL was written to the configuration is more than 6 characters. In that case, you will need to access the SAPL screen using the method described in "Overriding Stand-Alone Program Loader Defaults" on page 10.

Overriding the Nucleus Device and PARM Disk

If you want to override the CP nucleus and PARM disk, specify the I*devno* or I*devno.p* load parameter. The first letter of this parameter is "I" (short for "IPL").

- *devno* specifies the 5-digit device number where the parm disk holding the CP nucleus resides. The first digit of that device number identifies the subchannel set containing the device.
- *.p* specifies the 1-digit extent number of that parm disk.

Overriding Stand-Alone Program Loader Defaults

If the defaults that were established by the SALIPL utility when it wrote SAPL to the IPL device are acceptable, all you need to do is to IPL from the IPL device. If you need to change any of the defaults, however, then you must supply a load parameter with the address of a console for SAPL to use. The console specified must accept 3270 CCWs and data streams and have at least 20 lines and 80 columns.

If you are running SAPL on a real processor, you specify the console address by entering the device number of the console into the load parameter field of the load window on your processor console. The SAPL screen will be displayed on the specified console.

If you are running SAPL in a virtual machine, you specify the console address using the LOADPARM option of the CP IPL command. For example:

```
IPL 18B LOADPARM EA0
```

In this example, 18B is the DASD volume to load from and EA0 is the device number of the console to be used.

To cause the SAPL screen to appear on the integrated 3270 console on the HMC, specify SYSG as a load parameter. The operating system messages panel on the HMC (SYSC) cannot be used by SAPL because it does not accept 3270 CCWs and data streams.

To IPL from a SCSI Disk

If you are IPLing CP in a virtual machine from a minidisk on an emulated FBA device (SCSI), just specify the minidisk device address as the IPL device on the IPL command as shown in the previous example.

To IPL from a SCSI disk on the real processor, access the load window on the processor console in order to specify additional information. To IPL from a SCSI disk in a virtual machine by using an FCP device, use the CP SET LOADDEV command to specify additional information that will be used by the CP IPL command:

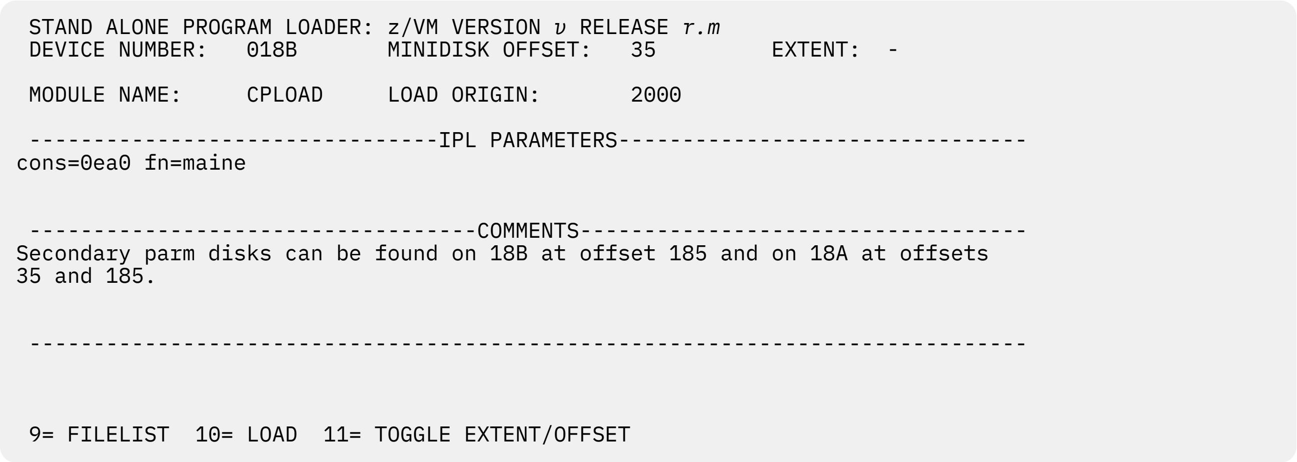
- Specify the load address of the FCP device you wish to use.
- Specify the world wide port name (PORT on the SET LOADDEV command)
- Specify the logical unit number (LUN)
- Specify 0 for the boot program selector (BOOTPROG)
- Specify C8 for the boot record logical block address (BR_LBA) if the IPL device has been CP-formatted by ICKDSF or CPFMTXA. The SALIPL utility writes the SCSI boot record to block 200 (X'C8') of a CP-formatted device. For a CMS-formatted device, SALIPL writes the SCSI boot record at block 200 (X'C8') into the RECOMP area of the minidisk.
- For the load parameter operand, optionally specify a console address on which you wish the SAPL screen to appear.
- For the OS specific load parameters (SCPDATA) operand, optionally specify a console address on which you wish the SAPL screen to appear.

  Note: As described earlier, the console specified must accept 3270 data streams. Specify SYSG for the integrated 3270 console on the HMC. The console may also be specified in the load parameter field of the load panel on the system console. Note that if both the OS specific load parameter SCPDATA and the load parameter fields are specified, SAPL will use the load parameter value.

If your configuration uses normal DASD devices (such as 3390), then the IPL device, SYSRES device, and parm disk device can all be the same device or different devices. **If the IPL device is an FCP device, then the IPL device, SYSRES device, and parm disk device must be the same SCSI disk.** This is because the FCP device, WWPN, and LUN combination specified on the load screen is the only DASD device path known during the IPL process.

The Stand-Alone Program Loader Screen

When the IPL completes and a valid console has been specified for the load parameter, you will see a menu screen similar to Figure 2 on page 12. The defaults will be filled in for some of the fields — all fields except the comments section can be altered.



```
 STAND ALONE PROGRAM LOADER: z/VM VERSION v RELEASE r.m
 DEVICE NUMBER:   018B       MINIDISK OFFSET:   35         EXTENT:  -

 MODULE NAME:     CPLOAD     LOAD ORIGIN:       2000

 -------------------------------IPL PARAMETERS-------------------------------
cons=0ea0 fn=maine


 ----------------------------------COMMENTS----------------------------------
Secondary parm disks can be found on 18B at offset 185 and on 18A at offsets
35 and 185.


 ----------------------------------------------------------------------------



 9= FILELIST  10= LOAD  11= TOGGLE EXTENT/OFFSET
```

Figure 2. Stand-Alone Program Loader

The Stand-Alone Program Loader screen contains the following fields:

DEVICE NUMBER

Identifies the DASD volume from which SAPL will read.

If the multiple subchannel set facility is installed, this field is five digits. The first digit specifies the subchannel set in which the device resides, while the remaining four digits specify the device number. If the DEVICE option was specified on the SALIPL command that created SAPL, the value of that option is displayed.

If the multiple subchannel set facility is not installed, this field is four digits and specifies the device number in subchannel set zero. If the DEVICE option was specified on the SALIPL command, the low-order four digits of the option value are displayed.

If the designated device does not exist, an error message is produced (if the SAPL panel is being displayed) or a disabled wait PSW is loaded.

If you IPLed from an FCP device (SCSI disk), the DEVICE NUMBER field will contain the FCP device address. The WWPN and LUN are not displayed on the SAPL screen and cannot be changed. If you IPLed from an FCP device, you can change the DEVICE NUMBER value to a DASD volume (3390). If you IPLed from a DASD volume, however, you cannot change to an FCP device.

MINIDISK OFFSET

Specifies the offset from the beginning of the DASD volume to the start of the CMS minidisk from which SAPL is to read. This number is cylinders for count key data devices and blocks for fixed block architecture devices.

EXTENT

Specifies the extent number from 1 to 9 (first to ninth). This field may be used only with DASD volumes that have been formatted using the Device Support Facilities Program (ICKDSF) and have an allocation map containing PARM extents written on them. When a valid extent is entered, the offset corresponding to the extent will be displayed in the MINIDISK OFFSET field.

Note: Only one of the fields, EXTENT or MINIDISK OFFSET, can be altered at a time. Use PF11 to toggle which field can be altered.

MODULE NAME

Specifies the name of a module on the minidisk that will be loaded by SAPL.

LOAD ORIGIN

Specifies the storage location (HEX address) into which SAPL will load a relocatable module. Specify a LOAD ORIGIN value of X'2000' if IPLing a CP module. As of z/VM V5.1, CP will always run with an origin of X'2000'. If a value other than X'2000' is specified, CP will initially load at the specified address and then relocate itself to location X'2000'.

IPL PARAMETERS

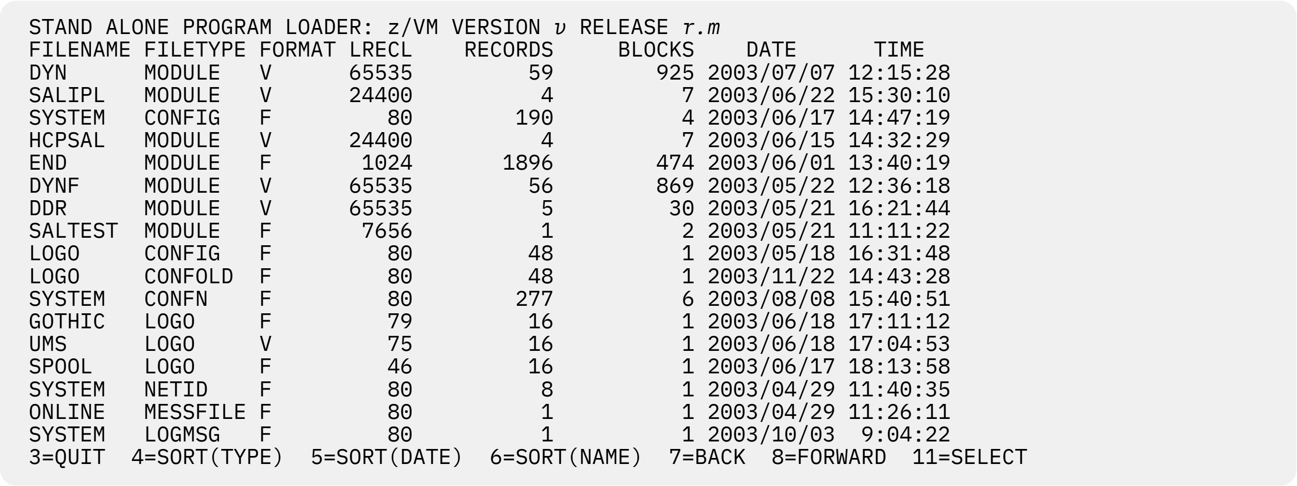
Specifies the data that will be passed to the loaded program by SAPL. The contents of this field depends entirely on the program loaded by SAPL. For parameters valid for CP, see "Passing IPL Parameters" on page 13. **If you are IPLing CP from an FCP device (SCSI disk), PDVOL=*addr* must be specified as an IPL parameter.** See the PDVOL= description below for more information.

COMMENTS

Contains comments such as operator instructions. This field may only be altered by running the SALIPL utility.

There is also a message line after the comments area. Any informational messages or error messages will appear on the message line.

There is a list of program function (PF) key numbers and their functions at the bottom of the screen. PF11 toggles which of the two fields, MINIDISK OFFSET or EXTENT, is active. PF10 causes SAPL to try to load a module using the current contents of the fields. PF9 displays a list (FILELIST) of files that are on the minidisk. The module you wish to use can be selected on that screen. Figure 3 on page 13 is an example of a Stand-Alone Program Loader file list screen.



```
STAND ALONE PROGRAM LOADER: z/VM VERSION v RELEASE r.m
FILENAME FILETYPE FORMAT LRECL    RECORDS     BLOCKS    DATE      TIME
DYN      MODULE   V      65535         59        925 2003/07/07 12:15:28
SALIPL   MODULE   V      24400          4          7 2003/06/22 15:30:10
SYSTEM   CONFIG   F         80        190          4 2003/06/17 14:47:19
HCPSAL   MODULE   V      24400          4          7 2003/06/15 14:32:29
END      MODULE   F       1024       1896        474 2003/06/01 13:40:19
DYNF     MODULE   V      65535         56        869 2003/05/22 12:36:18
DDR      MODULE   V      65535          5         30 2003/05/21 16:21:44
SALTEST  MODULE   F       7656          1          2 2003/05/21 11:11:22
LOGO     CONFIG   F         80         48          1 2003/05/18 16:31:48
LOGO     CONFOLD  F         80         48          1 2003/11/22 14:43:28
SYSTEM   CONFN    F         80        277          6 2003/08/08 15:40:51
GOTHIC   LOGO     F         79         16          1 2003/06/18 17:11:12
UMS      LOGO     V         75         16          1 2003/06/18 17:04:53
SPOOL    LOGO     F         46         16          1 2003/06/17 18:13:58
SYSTEM   NETID    F         80          8          1 2003/04/29 11:40:35
ONLINE   MESSFILE F         80          1          1 2003/04/29 11:26:11
SYSTEM   LOGMSG   F         80          1          1 2003/10/03  9:04:22
3=QUIT  4=SORT(TYPE)  5=SORT(DATE)  6=SORT(NAME)  7=BACK  8=FORWARD  11=SELECT
```

Figure 3. Stand-Alone Program Loader File List

The files displayed will initially be sorted in descending order by date and time. You can sort by file name/file type, file type/file name, and date/time by using the PF keys identified at the bottom of the screen. If you press PF11 (SELECT) when the cursor is on a line that lists a file, you will be returned to the main menu and the file name of the selected file will be placed in the MODULE NAME file.

Passing IPL Parameters

Using SAPL, you can pass up to 240 characters of parameters into CP at IPL time. The default contents of the 240-character parameter field can be selected when SALIPL is used to write SAPL to the IPL device. You can choose to override the defaults by using the full-screen option of SAPL.

During CP initialization, CP scans the IPL parameter list from left to right. If a parameter is specified more than once, CP uses the last valid occurrence of the parameter.

The following list describes the parameters that CP supports:

BETA=*string*

specifies a variable-length string that represents a left-to-right hexadecimal bit mask, in integral 1-byte entities. This string is an enablement mask for allowing functions that are otherwise disabled for beta test. IBM customer support provides the proper enablement mask for customers who have engaged in a beta test agreement with IBM.

CLEARPDR

is an IPL parameter that should only be used in emergency situations to recover from an IPL problem in an SSI cluster. It specifies that CP will clear the Persistent Data Record during initialization before when it is normally used. This parameter is intended to be used only when there are no other active members in the SSI cluster. If any other member of the SSI cluster is active, system integrity will be compromised, data corruption can occur and the active members will ABEND. The operator will be prompted at IPL time and be given the option of continuing or stopping the IPL in case any other member in the SSI cluster is active. After the Persistent Data Record is cleared, normal IPL continues.

CLEARPDR should be used only when the SSI cluster is down and the first member cannot join the cluster at IPL time because the PDR does not contain the correct current status of other members of the SSI. For example, this can occur during a disaster recovery scenario when copying the DASD containing the PDR while the other members are JOINED and then IPLing when the other members are DOWN or not present.

CLEARPDR is an easier alternative to the procedure of clearing a PDR by IPLing the SSI member in REPAIR mode and running the FORMSSI utility.

If the system is not a member of an SSI cluster, the CLEARPDR parameter is ignored. The CLEARPDR parameter should only be specified on the SAPL screen. It is only acknowledged during a system IPL and is not acknowledged during CP restart processing (SHUTDOWN REIPL or restarting after an abend).

CONS=*addr*
CON=*addr*

specifies the location of the operator console that CP is to use. Specify a 1- to 4-digit device address in subchannel set 0 for a 3270 console. Specify SYSC for the operating system messages panel on the Hardware Management Console (HMC). Specify SYSG for the integrated 3270 console on the HMC.

All OPERATOR_CONSOLES statements in the system configuration file are ignored. The console address specified will be added to the list of consoles defined on the EMERGENCY_MESSAGE_CONSOLES statement found in the system configuration file if it has not already been included in that list. For more information, see the description of the EMERGENCY_MESSAGE_CONSOLES statement.

FN=*filename*

specifies the file name of the system configuration file. If you do not specify this parameter, CP uses SYSTEM as the file name. If FN=*filename* is the only IPL parameter that you need to override, you could specify FN*xxxxxx* in the load parameter field of the load window of your processor console. By using the FN*xxxxxx* load parameter, the file name of your system configuration file is limited to 6 characters.

FT=*filetype*

specifies the file type of the system configuration file. If you do not specify this parameter, CP uses CONFIG as the file type.

IPLVAR=*value*

specifies the value of the CP.IPLPARMS.IPLVAR system environment variable. This is a special case of the environment variables that are normally set with the CP SET VARIABLE command or configuration file statement. The CP.IPLPARMS.IPLVAR variable can only be set by this IPL parameter and cannot be changed with the CP SET VARIABLE command or statement. This allows for one environment variable to be defined dynamically during system initialization.

The value for IPLVAR= is more restrictive than the other environment variables in order to satisfy the restrictions required for parsing the IPL parameters. The length must be 1 to 233 characters, no embedded blanks are allowed, and lowercase letters are automatically converted to uppercase. Valid characters are A to Z, 0 to 9, underscore(_), and period(.).

When using the SALIPL utility, make sure the characters used in the variable value are not line-editing symbols. If the value contains a character that is used as a line-editing symbol, the CP TERMINAL command should be issued first to define a different line-editing symbol or to turn off the line-editing function. Issue the CP QUERY TERMINAL command to view the characters used as line-editing symbols.

NOEXITS

is intended to assist you in a situation where your dynamic additions to CP are somehow wrong and are preventing CP from completing system initialization. Use the NOEXITS parameter to tell CP not to process any of these requests from your system configuration file:

- CPXLOAD statement
- ENABLE EXIT statement
- EXTERNAL_SYNTAX statement
- ASSOCIATE MESSAGES statement
- The ENABLE operand of the ASSOCIATE EXIT statement

Except for examining the statement for syntactic correctness, CP will ignore the statement or the operand. If it was one of your dynamic additions to CP (for example, a CP exit) that was causing CP to not initialize, using NOEXITS may help you to IPL CP so that you can correct the problem.

NOHCD

specifies that CP should ignore any IODF statement in the system configuration file. The system initializes as if no IODF statement were present and HCD is not used for I/O configuration.

An example of when the NOHCD IPL parameter would be useful is if you encounter a disabled wait state because of problems with the IODF statement or problems with the file specified on the IODF statement. Use the NOHCD parameter to IPL the system without HCD and then change the IODF statement, replacing the IODF or otherwise investigate the wait state condition.

PDNUM=*n*

specifies the number of the parm disk that you want CP to access to find the system configuration file. CP reads the allocation map on the IPL device or the DASD volume indicated on the PDVOL= parameter and skips to parm disk *n*. If the allocation map does not contain at least *n* parm disks, CP loads a disabled wait.

This parameter usually needs to be specified or changed if the PDVOL= parameter is specified or changed. See the PDVOL= parameter for more information.

PDOFF=*offset*

specifies the offset at which a minidisk resides on the IPL device or the DASD volume indicated on the PDVOL= parameter that contains the system configuration file. If the DASD volume does not have at least *offset* cylinders or blocks, or if there is no CMS-formatted minidisk at the specified offset, CP loads a disabled wait.

This parameter usually needs to be specified or changed if the PDVOL= parameter is specified or changed. See the PDVOL= parameter for more information.

PDVOL=*addr*

specifies a 4-digit real device number in the subchannel set CP is IPLing from. It is the volume on which the parm disk resides that you want CP to access to find the system configuration file. If there is no device at address *addr* or if the device is not a DASD volume, CP loads a disabled wait.

When the PDVOL= parameter is specified or changed, the PDNUM= or PDOFF= parameter usually needs to be specified or changed, because the layout of the new parm disk volume might be different from the IPL volume. If neither PDNUM= nor PDOFF= is specified, CP uses the offset (cylinder or block) of the minidisk that contained the CP module, which is probably not correct for the new parm disk volume. If CP cannot find the parm disk, CP loads a disabled wait.

If you are IPLing CP from an FCP device (SCSI disk), PDVOL=*addr* must be specified as an IPL parameter. Any device address that is not in the I/O configuration of the system to be IPLed can be specified for *addr*. CP requires this parameter in order build the EDEVICE control block for the SYSRES/parm disk device during CP initialization. For more information, see the EDEVICE configuration statement in *z/VM: CP Planning and Administration*.

PROMPT

specifies that you wish to have CP prompt you about start-up options at IPL time. If you specify this parameter, CP ignores the AUTO_WARM_IPL and AUTO_IPL operands on the FEATURES statement in the system configuration file.

The PROMPT parameter is only valid for the SAPL screen. This parameter is not acknowledged during CP bounce processing (SHUTDOWN REIPL or restarting after an abend).

REPAIR

specifies that SSI cluster initialization and management functions be bypassed. This parameter is intended to be used only when a running system is required to correct a problem and there are no other active members in the SSI cluster. If any other member of the SSI cluster is active, system integrity will be compromised and data corruption can occur. An operator prompt is issued at IPL time to give the operator the option of continuing or stopping the IPL in case any other member in the SSI cluster is active. Spool file initialization occurs based on the contents of the FEATURES statement in the system configuration file, unless the PROMPT parameter is used, in which case CP prompts you about startup options at IPL time. When the REPAIR parameter is specified, the NOAUTOLOG and DISABLE IPL options are automatically invoked, whether or not they are entered.

If the system is not a member of an SSI cluster, the REPAIR parameter causes CP to bypass automatic logon of virtual machines as if NOAUTOLOG was specified and to disable terminal devices as if the DISABLE IPL option was specified.

selector:value(IPL parameter substring)

provides conditional IPL parameters to address system configuration requirements that differ based on environmental conditions. The *selector* designates a parameter that if equal to *value* causes the IPL parameter substring to be used to configure the control program. If the *selector* does not equal *value*, the IPL parameter substring is ignored. The *selector* must be followed immediately by a colon and the *value* with no intervening blanks. The following *selectors* are supported:

FN

designates the configuration file name specified by either the last FN= parameter in the IPL parameter string, excluding conditional parameters, or the default value of SYSTEM.

The IPL parameter string is evaluated from left to right. Parameters that are not conditional can appear anywhere in the string, before or after conditional ones, and the order of parameters is preserved.

For example, if the IPL parameters are:

```
CONS=020 FN:SITE1(PDVOL=1111) NOEXITS FN:SITE2 (PDVOL=2222)
```

and z/VM is IPLed with a load parameter of:

```
FNSITE2
```

then the parameter:

```
FN=SITE2
```

will be appended to the IPL parameters by the Stand-Alone program Loader to produce:

```
CONS=020 FN:SITE1(PDVOL=1111) NOEXITS FN:SITE2 (PDVOL=2222)
FN=SITE2
```

which will in turn result in the IPL parameters:

```
CONS=020 NOEXITS PDVOL=2222 FN=SITE2
```

being used to configure the z/VM control program.

An FN= parameter within a conditional IPL parameters substring is ignored, except that syntax errors in such parameters is reported. In addition, only the last FN= parameter encountered is honored, so that for example:

```
FN=SITE1 FN:SITE1(PDVOL=1111) FN:SITE2(PDVOL=2222) FN=SITE2
CONS=020
```

would produce:

```
PDVOL=2222 FN=SITE2 CONS=020
```

not:

```
PDVOL=1111 FN=SITE2 CONS=020.
```

A new value for the parameter corresponding to the *selector* is ignored and only the last value of such a parameter in the unconditional portion of the IPL parameters is used. Conditional IPL parameters cannot be nested.

STORE=*nnnnu*

specifies the amount of permanent storage that you want CP to initialize at IPL. The value should be greater than or equal to the minimum amount of storage required to operate z/VM, as described in *z/VM: CP Planning and Administration*. If the value is not a multiple of the storage-increment size, it is rounded up to the next higher multiple. The value is adjusted downward if it exceeds the maximum storage size supported by z/VM or the sum of the `Initial + Reserved` storage allocations in the image profile for the logical partition, whichever is less.

nnnn is a 1- to 4-digit decimal number and *u* is the storage unit suffix:

M

megabytes

G

gigabytes

T

terabytes

P

petabytes

E

exabytes

STORE=INITIAL

indicates that, at IPL, CP should initialize an amount of permanent real storage equal to the amount of configured storage that CP sees at system initialization. This amount might not be the same as the partition's image profile INITIAL amount.

Usage Notes

1. After IPL, if standby storage is available, you can dynamically add permanent real storage, reconfigurable real storage, or both to the z/VM configuration by using the SET STORAGE command. After reconfigurable storage is defined to the configuration, it can be removed using the SET STORAGE command. For more information, see *z/VM: CP Commands and Utilities Reference*.
2. It is recommended you use the STORAGE statement in the system configuration file to define your real storage configuration rather than the STORE=IPL parameter. The STORAGE statement provides the flexibility to define permanent and reconfigurable storage at IPL.
3. The IPL parameter STORE= overrides the KEEP option on the STORAGE AFTER_SHUTDOWN_REIPL and STORAGE AFTER_RESTART statements in the system configuration file.

Chapter 3. Using the IBM I/O Exerciser for System z (ESAIO)

The IBM I/O Exerciser for System z (ESAIO) is a stand-alone tool that is used to verify a System z I/O configuration without having to run a production workload. This tool is intended to help identify possible cabling or definition errors by validating that all the paths defined to each device actually connect to the same physical device. Additionally, a data transfer stress test is performed to help verify the quality of the cabling infrastructure. These tests are especially useful after major infrastructure upgrades (for example, push/pull of a new processor, data center move, replacing storage devices). By running ESAIO before IPLing an operating system and starting a production workload, clients can avoid disruptions to production and insure that the correct set of devices are put in use.

ESAIO can be installed to run in either a native System LPAR or in a z/VM environment. The installation and activation procedures for each system are outlined below. Regardless of how it is installed and activated, ESAIO operates as described in "Using the Tool" on page 21.

This chapter describes how to install and use the ESAIO stand-alone tool.

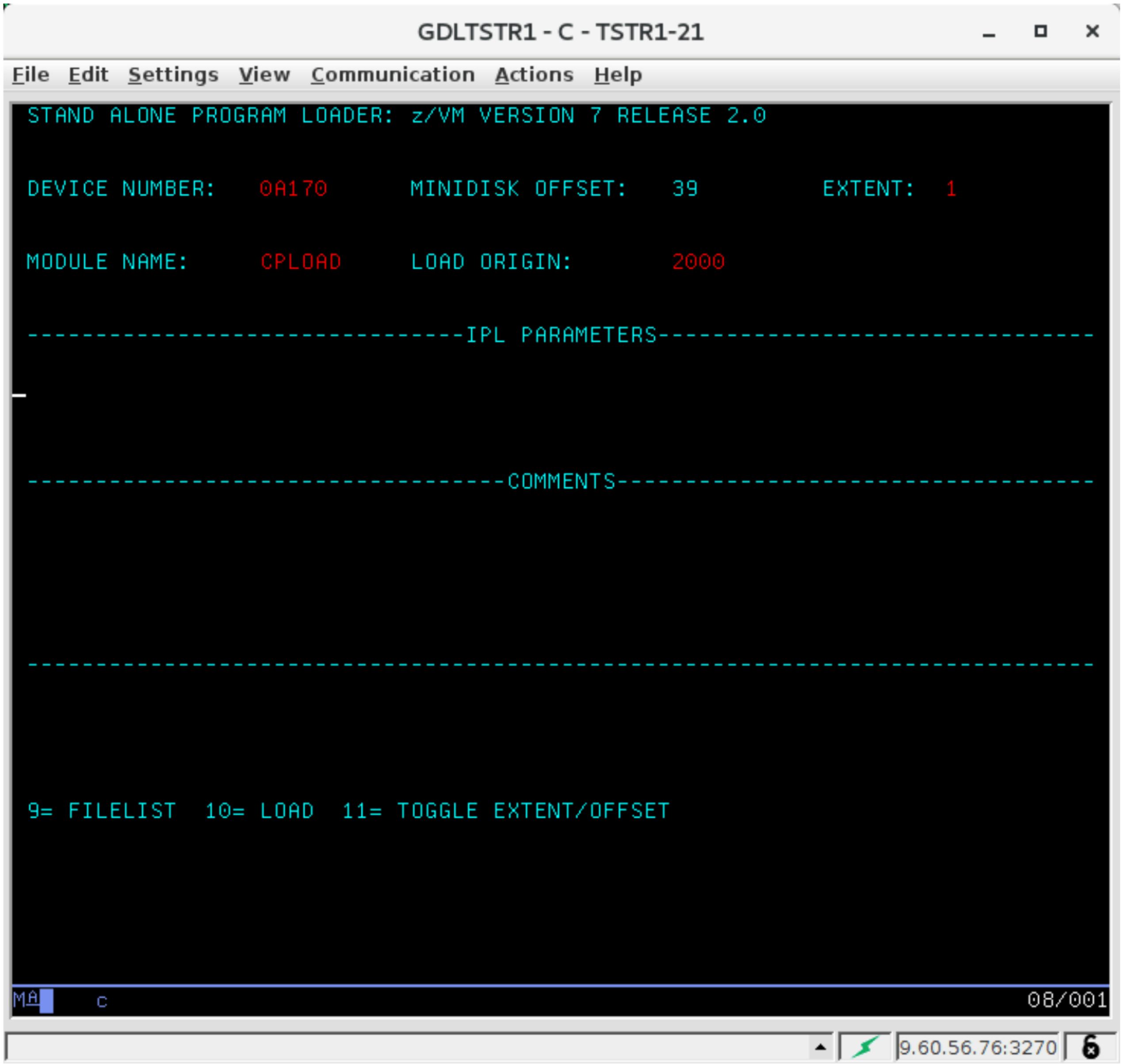
Installing the ESAIO Tool

For z/VM, the I/O path verification tool consists of one part – ESAIO MODULE – delivered as part of the original install on PARM disk 1 of the IPL pack. This is also the MAINT CF1 minidisk. The tool is loaded from a PARM disk using SAPL, the z/VM Stand Alone Program Loader. In order for SAPL to load the tool, ESAIO MODULE must be copied to a PARM disk.

Copy ESAIO MODULE to a PARM disk (for example, user MAINT's CF1 minidisk).

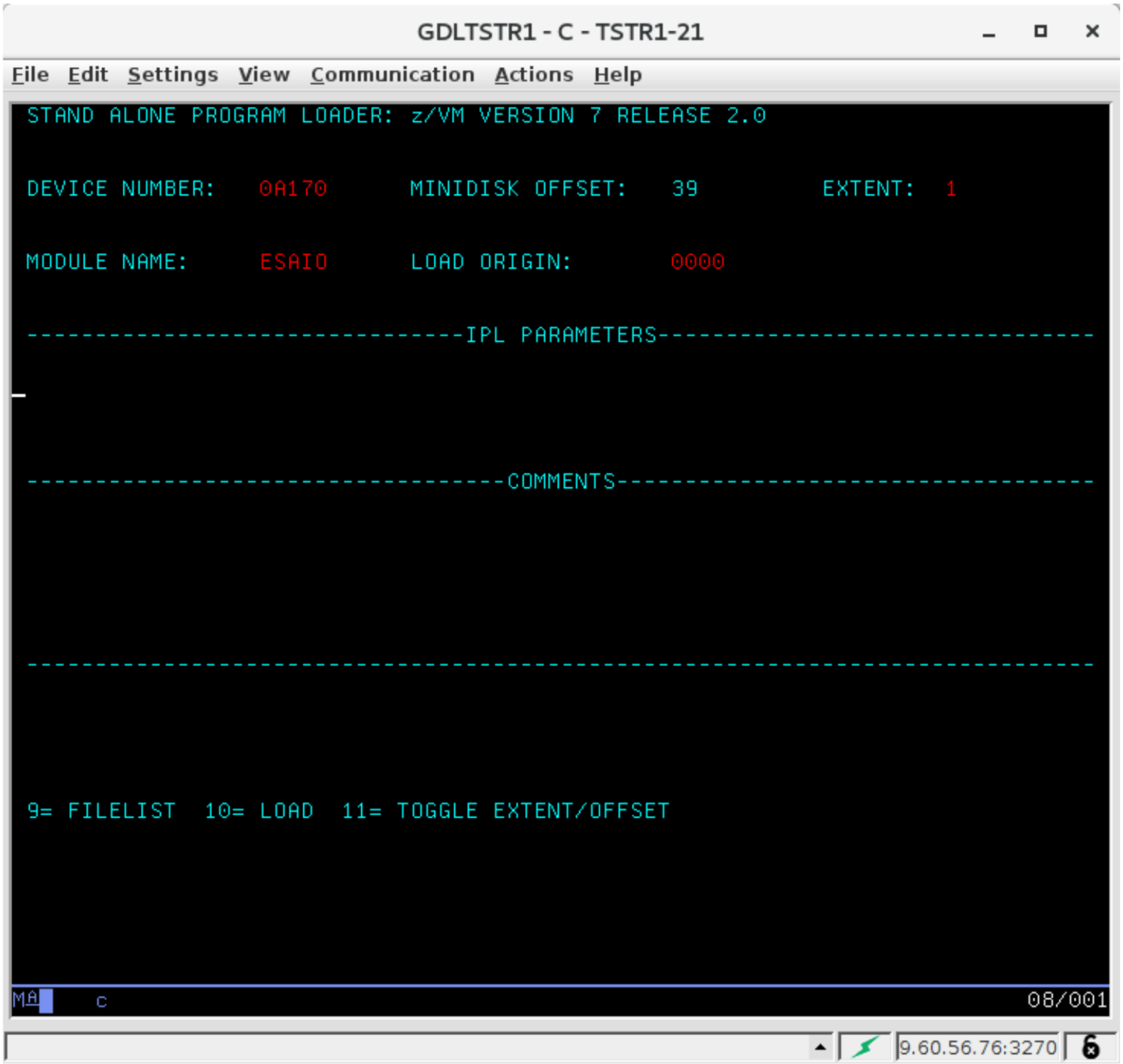
Activating the Loader

IPL any volume where SAPL is installed and specify the 3270 console device number as the IPL Load Parameter. This will bring up the SAPL screen.



```
STAND ALONE PROGRAM LOADER: z/VM VERSION 7 RELEASE 2.0

DEVICE NUMBER:    0A170      MINIDISK OFFSET:    39         EXTENT:  1

MODULE NAME:      CPLOAD     LOAD ORIGIN:        2000

--------------------------------IPL PARAMETERS--------------------------------



-----------------------------------COMMENTS-----------------------------------




------------------------------------------------------------------------------



9= FILELIST  10= LOAD  11= TOGGLE EXTENT/OFFSET
```

Select the PARM disk where ESAIO MODULE resides, specifying its device number and either its extent number or offset. Specify ESAIO as the name of the module to be loaded and set the load origin to 0.



```
STAND ALONE PROGRAM LOADER: z/VM VERSION 7 RELEASE 2.0

DEVICE NUMBER:    0A170      MINIDISK OFFSET:   39         EXTENT:  1

MODULE NAME:      ESAIO      LOAD ORIGIN:       0000

--------------------------------IPL PARAMETERS--------------------------------



-----------------------------------COMMENTS-----------------------------------




------------------------------------------------------------------------------



9= FILELIST  10= LOAD  11= TOGGLE EXTENT/OFFSET
```

Press PF10 to load ESAIO into memory.

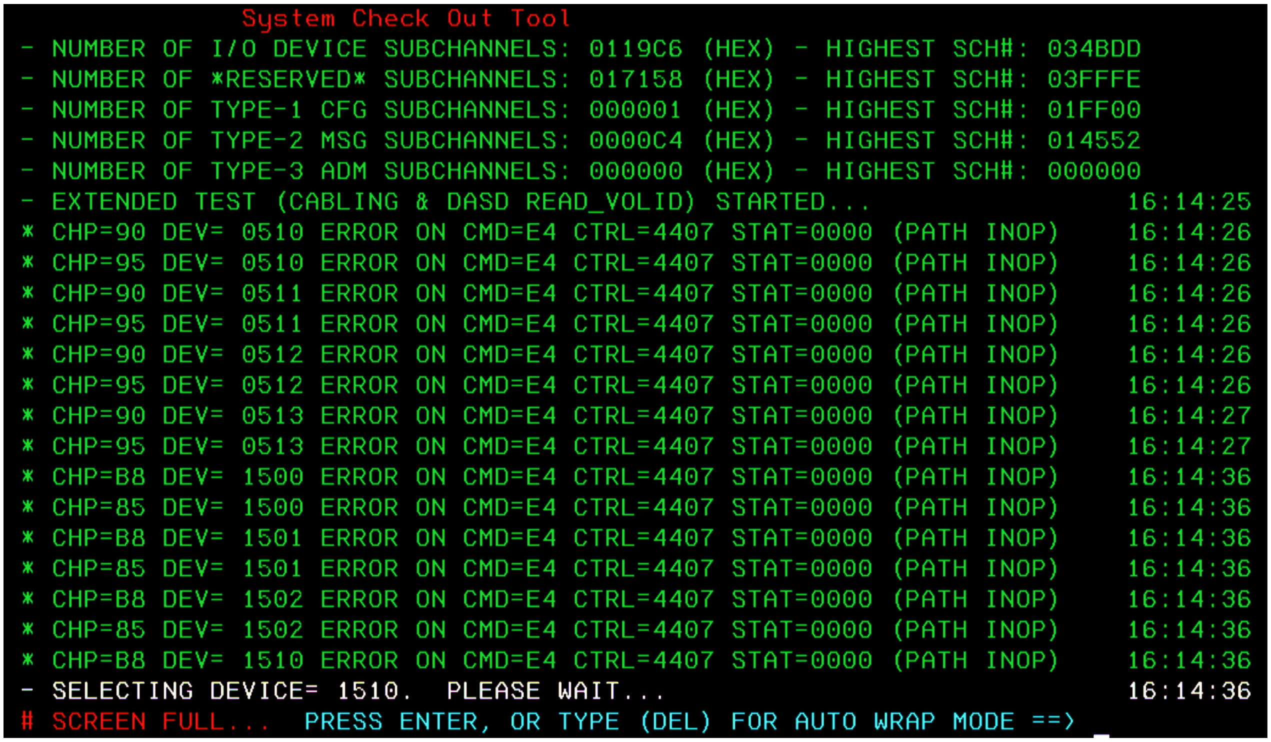
For more information on input fields, see "The Stand-Alone Program Loader Screen" on page 11.

Note: If running in a virtual machine, press PA1 to clear screen and drop into CP READ. Enter #CP ATT(N) to send attention to ESAIO to display tool's console.

Using the Tool

After IPLing the ESAIO tool, hit ENTER on a 3270 terminal to cause the tool to start testing paths to every device to which the LPAR has access. The first test verifies that all the paths defined are actually connected to the same device. If any configuration errors are found, messages are displayed on the console. Hit ENTER again to scroll after the screen fills.

This next screen image is while checking all of the paths for connectivity. The errors below show that not operational conditions were found. This can occur because the I/O configuration was over defined, meaning devices were defined but do not exist or devices have been powered off or the SAN connections are faulty or blocked.

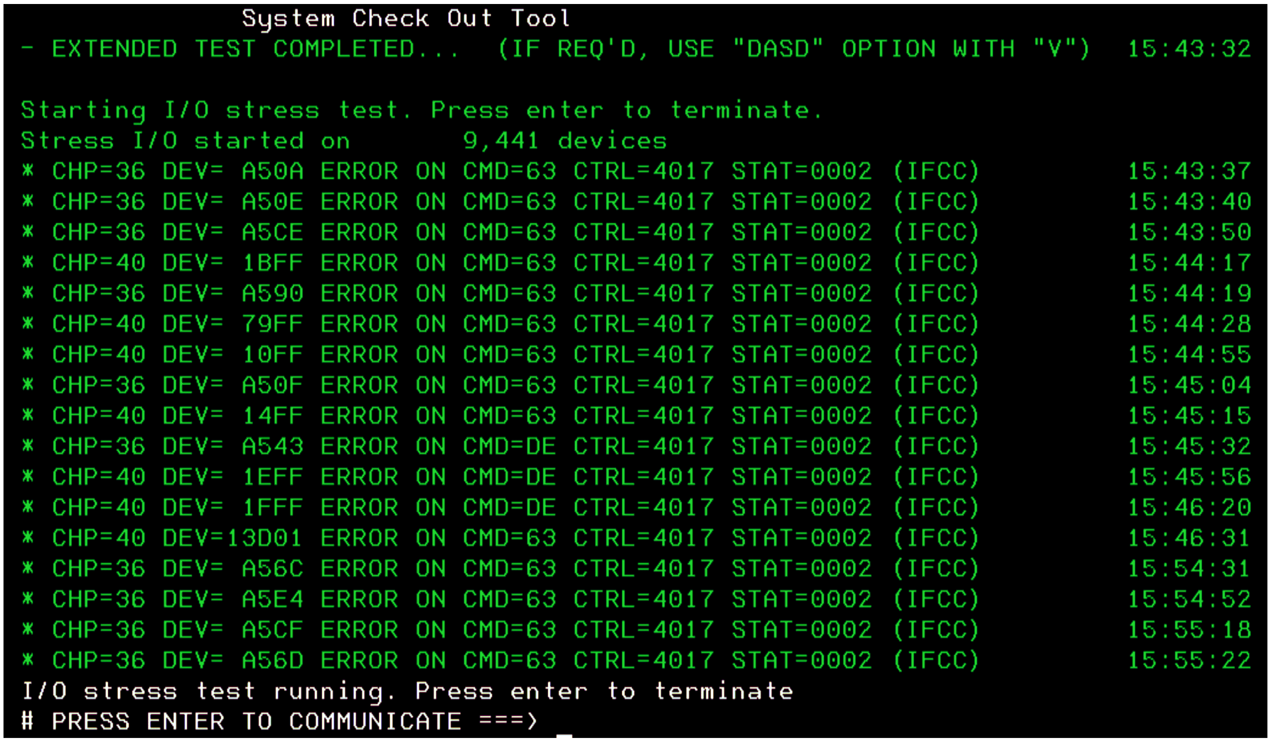


```
               System Check Out Tool
- NUMBER OF I/O DEVICE SUBCHANNELS: 0119C6 (HEX) - HIGHEST SCH#: 034BDD
- NUMBER OF *RESERVED* SUBCHANNELS: 017158 (HEX) - HIGHEST SCH#: 03FFFE
- NUMBER OF TYPE-1 CFG SUBCHANNELS: 000001 (HEX) - HIGHEST SCH#: 01FF00
- NUMBER OF TYPE-2 MSG SUBCHANNELS: 0000C4 (HEX) - HIGHEST SCH#: 014552
- NUMBER OF TYPE-3 ADM SUBCHANNELS: 000000 (HEX) - HIGHEST SCH#: 000000
- EXTENDED TEST (CABLING & DASD READ_VOLID) STARTED...              16:14:25
* CHP=90 DEV= 0510 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:26
* CHP=95 DEV= 0510 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:26
* CHP=90 DEV= 0511 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:26
* CHP=95 DEV= 0511 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:26
* CHP=90 DEV= 0512 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:26
* CHP=95 DEV= 0512 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:26
* CHP=90 DEV= 0513 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:27
* CHP=95 DEV= 0513 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:27
* CHP=B8 DEV= 1500 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:36
* CHP=85 DEV= 1500 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:36
* CHP=B8 DEV= 1501 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:36
* CHP=85 DEV= 1501 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:36
* CHP=B8 DEV= 1502 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:36
* CHP=85 DEV= 1502 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:36
* CHP=B8 DEV= 1510 ERROR ON CMD=E4 CTRL=4407 STAT=0000 (PATH INOP)    16:14:36
- SELECTING DEVICE= 1510.  PLEASE WAIT...                              16:14:36
# SCREEN FULL...  PRESS ENTER, OR TYPE (DEL) FOR AUTO WRAP MODE ==> _
```

After the path tests complete a stress test is performed. Multiple concurrent I/O operations are started repeatedly to every DASD, tape, and CTC device. For DASD, the channel program used reads the first ten tracks on the volume. For tape and CTC, the channel program used issues ten Sense ID commands to each device.

Note: The stress test really will stress the I/O subsystem, so if devices are shared with other partitions, care should be taken to avoid interfering with workload that is running in those partitions. FBA DASD are not supported and will display (CMD-REJECT) status on the stress test.

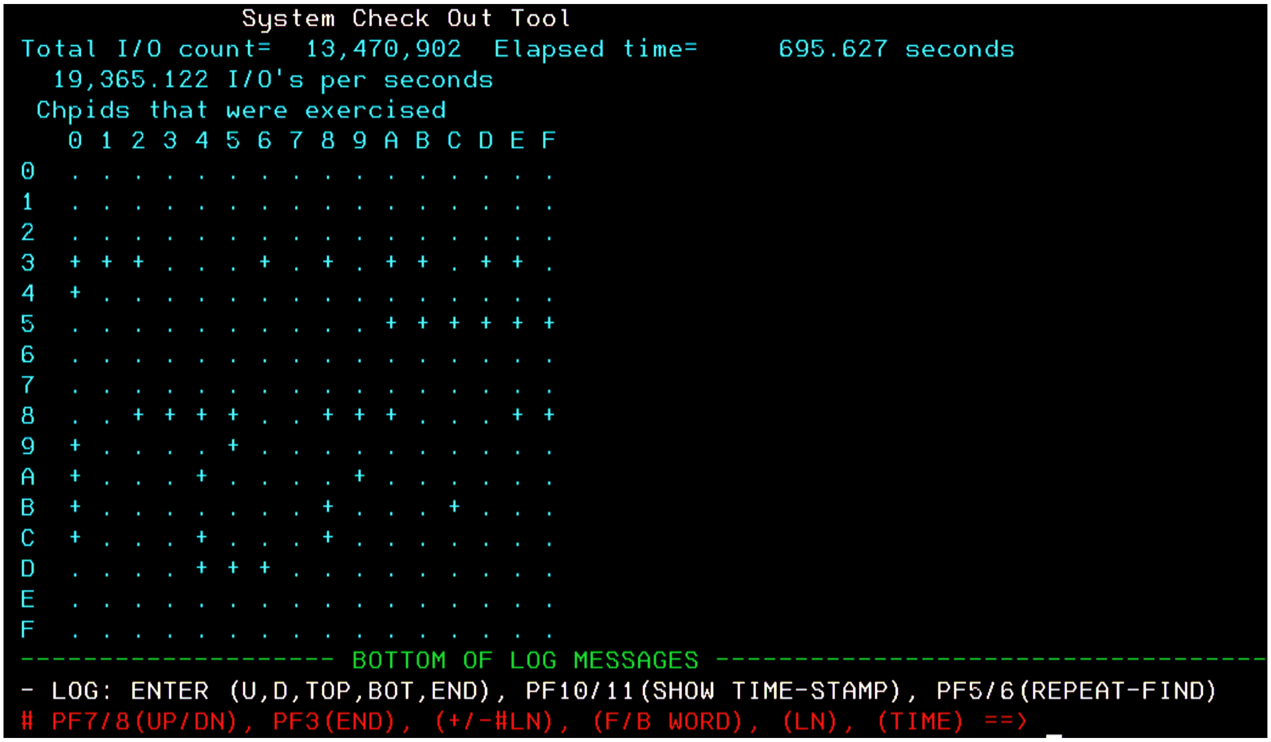
This next screen image is displayed after checking all of the paths and before starting the I/O stress test. We now see some interface control check errors (IFCC's) occurring on different paths/devices. IFCC's can occur because of poor quality fiber connections. Dirt can accumulate on the optics if the dust covers are left off. Defects can occur if the cables are bent or pulled to stretch around corners, for example. IFCC errors may be intermittent.



```
              System Check Out Tool
- EXTENDED TEST COMPLETED...  (IF REQ'D, USE "DASD" OPTION WITH "V")  15:43:32

Starting I/O stress test. Press enter to terminate.
Stress I/O started on       9,441 devices
* CHP=36 DEV= A50A ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:43:37
* CHP=36 DEV= A50E ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:43:40
* CHP=36 DEV= A5CE ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:43:50
* CHP=40 DEV= 1BFF ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:44:17
* CHP=36 DEV= A590 ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:44:19
* CHP=40 DEV= 79FF ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:44:28
* CHP=40 DEV= 10FF ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:44:55
* CHP=36 DEV= A50F ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:45:04
* CHP=40 DEV= 14FF ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:45:15
* CHP=36 DEV= A543 ERROR ON CMD=DE CTRL=4017 STAT=0002 (IFCC)         15:45:32
* CHP=40 DEV= 1EFF ERROR ON CMD=DE CTRL=4017 STAT=0002 (IFCC)         15:45:56
* CHP=40 DEV= 1FFF ERROR ON CMD=DE CTRL=4017 STAT=0002 (IFCC)         15:46:20
* CHP=40 DEV=13D01 ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:46:31
* CHP=36 DEV= A56C ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:54:31
* CHP=36 DEV= A5E4 ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:54:52
* CHP=36 DEV= A5CF ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:55:18
* CHP=36 DEV= A56D ERROR ON CMD=63 CTRL=4017 STAT=0002 (IFCC)         15:55:22
I/O stress test running. Press enter to terminate
# PRESS ENTER TO COMMUNICATE ===>
```

The I/O stress test runs until ENTER is pressed on the console. The tool then reports the amount of activity it generated and shows which channel paths were exercised before entering the LOG function and allowing one to scroll through the console output.

When the ENTER key is pressed the stress test stops and a summary is presented, as shown below.



```
              System Check Out Tool
Total I/O count=  13,470,902  Elapsed time=     695.627 seconds
  19,365.122 I/O's per seconds
 Chpids that were exercised
   0 1 2 3 4 5 6 7 8 9 A B C D E F
0  . . . . . . . . . . . . . . . .
1  . . . . . . . . . . . . . . . .
2  . . . . . . . . . . . . . . . .
3  + + + . . . + . + . + + . + + .
4  + . . . . . . . . . . . . . . .
5  . . . . . . . . . . + + + + + +
6  . . . . . . . . . . . . . . . .
7  . . . . . . . . . . . . . . . .
8  . . + + + + . . + + + . . . + +
9  + . . . . + . . . . . . . . . .
A  + . . . + . . . . + . . . . . .
B  + . . . . . . . + . . . + . . .
C  + . . . + . . . + . . . . . . .
D  . . . . + + + . . . . . . . . .
E  . . . . . . . . . . . . . . . .
F  . . . . . . . . . . . . . . . .
-------------------- BOTTOM OF LOG MESSAGES ------------------------------------
- LOG: ENTER (U,D,TOP,BOT,END), PF10/11(SHOW TIME-STAMP), PF5/6(REPEAT-FIND)
# PF7/8(UP/DN), PF3(END), (+/-#LN), (F/B WORD), (LN), (TIME) ==>
```

Chapter 4. Bringing Up the System

This chapter tells you how to bring up a z/VM system and a production system in a virtual machine. It also tells you where to find out how to start up any required service virtual machines.

If you need only a brief summary of the tasks in this chapter, see "Summary of Bringing Up the System" on page 52.

After reading this chapter, you can:

- Bring up a z/VM system
- Locate information about how to start up any required service virtual machines
- Log on the production system's virtual machine
- Bring up a production operating system in a virtual machine
- Check to make sure the system has been properly initialized and make adjustments where needed
- Prepare the system for a possible failure
- Respond to errors during bring-up.

Before You Begin

When you bring up a native z/VM system, you are initializing the configuration that your installation defined during system generation. (*z/VM: CP Planning and Administration* describes, in detail, the system generation process.)

Before bringing up this system, make sure that:

1. Your processor is partitioned correctly.
2. The real processor is ready for use.
3. The system residence volume is ready for use.
4. The volumes containing the checkpoint and warm start cylinders or pages are ready for use, if they are not included on the system residence volume.
5. Any volumes containing user minidisks are ready for use.
6. The system paging and spooling volumes are ready for use.
7. The other devices in the configuration are on, ready, and available to the processor you are using.
8. The z/VM system console is available.
9. Your required storage is available.
10. If you are using a processor that requires it, you have selected the correct I/O configuration data set (IOCDS).
11. If you are planning to perform a cold start, you have mounted the tape that contains the spool files and system data files you previously backed up using the SPXTAPE command, if any.

If your installation has changed the z/VM system configuration statements that define the new I/O configuration to the Control Program (CP), make sure that:

1. Your installation has run the Input/Output Configuration Program (IOCP) to create a new IOCDS.
2. You select the new IOCDS.
3. If you are planning to use the dynamic I/O capabilities of z/VM, you must ensure that:
   - In the system configuration file, the FEATURES statement has enabled **dynamic_i/o**.
   - From the system hardware console, you have:
     - Specified an IOCP extension percentage to allow for additional HSA (Hardware System Area) space

  - Set the 'Allow Dynamic Modification' bit

  These are required to exploit this capability.

For information about running the IOCP, see the appropriate publication listed in the bibliography.

Finally, make sure you know the following device numbers:

- The real device number of the device on which z/VM resides.

  To bring up z/VM, you need to specify this real device number on the processor complex's system console.
- The virtual device number your installation assigns to the device on which your production system resides.

  To bring up your production system, you need to specify this virtual device number on the IPL command.

Now you are ready to load CP into storage.

Bringing Up z/VM

To bring up z/VM, you must:

1. Load CP into storage
2. Select the type of start you want to perform (WARM, FORCE, COLD, or CLEAN)
3. Set the time-of-day clock
4. Help CP synchronize the time-of-day clocks in your processor complex.

Loading CP into Storage

To load CP into storage, use the Load task on the Hardware Management Console or IPL the DASD device. See Chapter 2, “Using the Stand-Alone Program Loader,” on page 9.

Selecting the Type of Start You Want to Perform

As soon as you load z/VM into storage, CP becomes active and:

- Displays messages about errors encountered in IPL parameter processing.
- Displays messages about errors encountered in configuration file processing.
- Displays a message for each CP-owned volume that is offline.

  Be sure all required volumes are mounted and online before continuing. If you continue initialization and there are spool volumes offline, you will receive messages indicating the volumes that are offline and the impact their absence will have. You then have an opportunity to end initialization without causing a loss of spool files or to continue initialization.
- Asks you to specify the type of start you want to perform.

In response you may:

- Specify a warm start, force start, cold start, or clean start.

  If an automatic IPL has been enabled using a FEATURES statement in the system configuration file, CP attempts a start without changing the TOD clock. If the TOD clock is not set or the specified type start cannot be performed, CP will prompt the operator for more information. If an automatic start is not enabled CP goes through all the usual prompts in the IPL process.
- Cancel the process of bringing up z/VM.
- Modify the type of start with the DRAIN or DISABLE options.
- Bring up the system without the user directory by specifying the NODIRECT option.

- Bring up the system without autologging any users beyond the operator virtual machine by specifying the NOAUTOLOG option.

If you do not receive CP's prompt for the type of start you want to perform, check the status of any DASD you share with another processor complex. If an operating system running on another processor complex has reserved a shared DASD, CP does not finish initializing until the other system releases the device. To release the device, contact the operator of the other system and follow your installation's normal procedures.

Warm Starts, Force Starts, Cold Starts, and Clean Starts

z/VM provides you with four basic types of starts: warm, force, cold, and clean. The difference among them is in how much of the system's environment CP restores, with a warm start restoring the most and a clean start the least. You save the system's environment by entering the SHUTDOWN command. You restore it by bringing up CP with a warm start. If a warm start fails, contact your system support personnel before you use a force or cold start.

In order to restore the system's environment, you must save it first. As mentioned, you do this by entering a SHUTDOWN command. When you enter this command, CP saves the following parts of the system's environment:

- Any accounting, EREP, and symptom records in storage.

  Accounting, EREP, and symptom record recording service virtual machines normally transfer these records from storage to disk. If these virtual machines are not retrieving, the records remain in storage. The SHUTDOWN command will write any records that are still in storage into the checkpoint area on DASD so that during the next IPL the records can be returned to storage for retrieval.
- System log message.

  The system log message communicates information to users when they log on or reconnect.
- Spool file queues.

  Spool files are collections of data on disk waiting to be processed by real or virtual readers, printers, or punches. The SHUTDOWN command saves information about the queues that locate these spool files.

  Note: Spool files may not be restored in the same order following a warm or force start.
- The system data file queues.

  The system data file queues hold the following collections of data, called **system data files**: named saved systems (NSSs), discontiguous saved segments (DCSSs), image libraries, user class restructure (UCR) files, message repository files, and system trace files.

  Note: UCR files might still exist but are no longer supported and will not be processed by CP.

  - A **named saved system (NSS)** is a copy of an operating system that a user has named and retained in a system data file. The user can load the operating system by its name, which is more efficient than loading it by device number.
  - A **discontiguous saved segment (DCSS)** refers to one or more pages of storage that a user has named and retained in a system data file. When a discontiguous saved segment, defined as shareable, is loaded, more than one user can access it.
  - An **image library** is a set of modules, contained in a system data file, that define the spacing, characters, and copy-modification data that a printer uses to print a spool file.
  - **Message repository** files contain z/VM messages and responses translated into a national language.
  - **System trace** files contain records of events that occur within the system. Use these files to determine the source of problems in the system.

  The SHUTDOWN command saves information about the queues that locate these system data files.
- The status of unit record devices, displays, and 3270 printers.

When you enter the SHUTDOWN command, you save all these parts of the system's environment. Then, when you bring the system back up with a warm start, CP restores them.

If you enter SHUTDOWN REIPL, CP attempts to do an automatic start of the specified module.

See the *z/VM: CP Commands and Utilities Reference* for other restrictions when you re-IPL another module.

Unless you have a reason to do otherwise, specify a warm start. Keep in mind that if you do not shut down the z/VM system with the SHUTDOWN command, you may not be able to bring it back up with a warm start. The information that the SHUTDOWN command saves is what allows a warm start to work.

During a force start, CP tries to restore most of what it restores during a warm start, but it may not be able to do so completely. Just as in a warm start, CP tries to restore, in this order:

1. Accounting, EREP, and symptom records in storage
2. System log message
3. Spool file and system data file queues.

However, if CP encounters an error during the first step, it immediately goes on to recover the spool file and system data file queues. Unless you specified FEATURES ENABLE LOGMSG_FROM_FILE in the system configuration file, the system log message is lost. Any remaining accounting, EREP, and symptom records are lost. Further, if CP encounters an error while recovering a spool file, it immediately goes on to the next file in the queue, and the file on which the error occurred is scheduled to be deleted. When the preliminary phase of spooling initialization is complete, a message providing spool file summary status is displayed, and the operator can then stop system initialization without the loss of any spool files.

Also, a force start does not restore any of the unit record device characteristics, such as the class to be processed or the image library. Instead, a force start tries to start the unit record devices with the default characteristics specified on the RDEVICE statement in the system configuration file. A force start does try to restore the status of all displays and 3270 printers.

Therefore, after a force start, CP may not have completely recovered all of the system's environment. Use a force start only when authorized by the system support personnel after a warm start fails.

If a force start fails, you must perform a cold start. During a cold start, CP tries to recover *only* the system data files. This means that when you perform a cold start, you lose all of your spool files, all accounting, EREP, and symptom records in storage. The system log message is lost if you did not specify the FEATURES ENABLE LOGMSG_FROM_FILE statement in the system configuration file. The status of your unit record devices, displays, and 3270 printers may also differ from their status before the restart. Use a cold start only when both a warm start and a force start fail.

The essential points to grasp about warm starts, force starts, cold starts, and clean starts are:

- To save the system's environment before you restart it, use the SHUTDOWN command.
- To restore the system's environment, bring the system back up with a warm start.
- If a warm start fails,
  - Check the DASD volumes.
  - Contact system support personnel.
  - When repeated attempts to perform a warm start fail, try a force start to restore as much of the system's environment as possible.
- If you decide to continue with system initialization even though there are files scheduled to be deleted, remember that those files cannot be recovered.
- If a force start fails, try a cold start.
- A clean start IPLs the system without attempting to recover spool files and system data files that existed prior to system shutdown.

The next section explains how to specify each of the basic types of starts.

Specifying a Warm Start

When CP asks you to specify the type of start you want to perform, it issues the following prompt:

```
Start ((Warm|Force|COLD|CLEAN) (DRain) (DIsable) (NODIRect) (NOAUTOlog))
      or (SHUTDOWN)
```

Specify a warm start by doing one of the following:

- Enter:

  ```
  warm
  ```

- Or press ENTER.

Either of these responses causes CP to perform a warm start. After the warm start completes, the system's environment is the same as it was before you shut down the system with the SHUTDOWN command.

If no warm start data is available, CP prompts the operator for more information:

```
Invalid warm start data encountered.
No files have been deleted yet.
To change to a FORCE start, enter FORCE.
To stop processing, enter STOP.
```

If you are performing a warm start and encounter spooling errors, CP advises you of these errors after the preliminary phase of spooling initialization is complete. A message is displayed that gives you the spool file summary status. CP then issues a prompt similar to the following:

```
Spool file errors encountered during initialization.
No files have been deleted yet.
To continue restart and delete the files, enter GO.
To stop so you can check spool volumes, enter STOP.
```

You must now decide whether to continue system initialization (losing the indicated files) or to stop system initialization (without losing any files).

To continue system initialization, enter:

```
go
```

To stop system initialization, enter:

```
stop
```

CP may not be able to perform a warm start for a number of reasons. I/O errors, incorrect data in the warm-start area, or invalid data in the checkpoint area may prevent a warm start from completing successfully.

When a warm start fails, you may be able to correct the problem that caused the failure. You can then perform a warm start successfully without losing any data that might have been lost because of the warm start failure.

A common reason for a warm start failure is that a volume containing spool files that was present during system termination may now be offline. You can attempt to correct this problem by using the following procedure:

1. Reload z/VM into storage.
2. Check the error messages. (Before CP prompts you for the type of start you want to perform, it issues an error message for each CP volume defined in the system configuration file that is not currently online.)
3. If you receive an error message for a volume that should be mounted, you respond to the start prompt by entering:

   ```
   shutdown
   ```

4. Make sure the appropriate spooling volume is mounted and ready.
5. Reload z/VM into storage and retry the warm start.

The checkpoint and warm-start data that was saved during the last system termination or shutdown is not changed when the warm start fails. If a hardware error has caused a warm start to fail one or more times, and you are able to correct the hardware error and reload the system, the warm-start procedure still restores the system's environment.

When repeated attempts to perform a warm start fail, contact your installation's support personnel before you proceed. After the support personnel have performed any necessary diagnosis, make sure all spooling volumes are mounted and ready. Then, go back to the processor complex's system console and reload z/VM into storage.

Specifying a Force Start

When the system is reloaded, CP again prompts you for the type of start you want to perform. If a warm start fails and you are authorized by the system support personnel, enter:

```
force
```

This response causes CP to perform a force start. After the force start completes, much of the system's environment may be the same as it was before you shut down the system with the SHUTDOWN command. But, CP may have lost some data, as explained before.

Also, if CP encounters an error during the force start, it may not automatically start your unit record devices or enable your displays. If you need any of these devices, you must explicitly start or enable them. (To find out how to start or enable a device, see Chapter 6, “Controlling Local Devices,” on page 81.)

If you are performing a force start and encounter spooling errors, CP advises you of these errors after the preliminary phase of spooling initialization is complete. A message is displayed that gives you the spool file summary status. CP then issues a prompt similar to the following:

```
Spool file errors encountered during initialization.
No files have been deleted yet.
To continue restart and delete the files, enter GO.
To stop so you can check spool volumes, enter STOP.
```

You must now decide whether to continue system initialization (losing the indicated files) or to stop system initialization (without losing any files).

To continue system initialization, enter:

```
go
```

To stop system initialization, enter:

```
stop
```

Specifying a Cold Start

When the system is reloaded, CP again prompts you for the type of start you want to perform. If a force start failed and you are authorized by the system support personnel, enter:

```
cold
```

This response causes CP to perform a cold start.

When you are performing a cold start and the preliminary phase of spooling initialization is complete, CP displays a message that gives you the spool file summary status. This message is always displayed. CP then issues a prompt similar to the following:

```
Spool files will be deleted because of COLD start.
No files have been deleted yet.
To continue COLD start and delete files, enter GO.
To stop COLD start without deleting files, enter STOP.
```

If you want to continue cold-start processing, enter:

```
go
```

If you want to end system initialization (without losing any spool files), enter:

```
stop
```

Attention: The cold-start procedure purges:

- All spool files
- All accounting, EREP, and symptom records in storage
- The system log message.

CP tries to recover the system data files.

Specify a cold start, therefore, only if you *cannot* bring up the system with a warm start or force start.

Specifying a Clean Start

When the system is reloaded, CP prompts you for the type of start you want to perform. CP issues the following prompt:

```
Start ((Warm|Force|COLD|CLEAN) (DRain) (DIsable) (NODIRect) (NOAUTOlog))
      or (SHUTDOWN)
```

Specify a clean start by entering:

```
clean
```

This response causes CP to perform a clean start. When clean start is selected, you receive the following prompt:

```
CLEAN start has been selected.  This will cause all
spool files and System Data Files (NSS, DCSS, TRF, IMG,
UCR, NLS) to be deleted.

No files have been deleted yet.
To continue CLEAN start and delete files, enter GO.
To stop CLEAN start without deleting files, enter STOP.
```

You must now decide whether to continue system initialization (deleting the indicated files) or to stop system initialization (without deleting any files).

To continue system initialization, enter:

```
go
```

To stop system initialization, enter:

```
stop
```

Attention: The clean start procedure purges:

- All spool files
- All system data files
- All accounting, EREP, and symptom records in storage
- The system log message.

A clean start is desirable in the following scenarios:

- IPLing after replacing existing spool volumes with new volumes.
- First time IPLing with a new warmstart area defined (spool file recovery information is stored in the warm-start area).
- You do not want to restore previously existing system data files.

Canceling the Bring-Up Process

If you need to cancel the process of bringing up z/VM, enter the following response to the start prompt:

```
shutdown
```

You can reload the Control Program when you are ready to bring up the system again. Entering SHUTDOWN to cancel the process of bringing up the system does not affect the existing checkpoint and warm-start data.

Modifying the Type of Start (DRAIN, DISABLE)

When you bring up z/VM, CP reactivates the unit record devices and displays that were active when you last shut down the system. If you do not want to reactivate these devices, you can indicate that you want CP to do one or both of the following:

- Suspend spooling activity to all of the unit record devices that are attached to the system
- Keep the displays that are attached to the system from being enabled.

Because CP does not restore spool files during a cold or clean start, you may want to keep the displays from being enabled. This gives you a chance to restore the files from tape before users log on and try to use the files.

Suspending Spooling Activity to the Unit Record Devices

To suspend spooling activity to the unit record devices, respond to CP's start prompt with the DRAIN option and either WARM, FORCE, COLD, or CLEAN. For example, if you want CP to perform a warm start and suspend all spooling activity to the unit record devices, enter:

```
warm drain
```

To start spooling activity to some or all of these unit record devices, enter the appropriate START commands. (To find out how to start a device, look under the device in Chapter 6, “Controlling Local Devices,” on page 81.)

Keeping Displays from Being Enabled

If you want to keep the displays from being enabled, respond to CP's start prompt with the DISABLE option and either WARM, FORCE, COLD, or CLEAN. For example, if you want to perform a warm start and keep the displays from being enabled, enter:

```
warm disable
```

When you are ready to enable some or all of these displays, enter the appropriate ENABLE commands. (To find out how to enable a display, see “Making a Display, a 3270 Printer, or Communication Lines Available for User Access” on page 86.)

Similarly, if you want to perform a warm start, suspend spooling activity to the real unit record devices, and keep the displays from being enabled, enter:

```
warm drain disable
```

When you are ready to start or enable some of the devices, enter the appropriate START or ENABLE commands.

Note:

1. If you include the DISABLE option, enter the following command to reenable the z/VM system console before you do anything else:

   ```
   enable rdev
   ```

where *rdev* is the real device number of the display you are using as the z/VM system console. This command lets you log back on if you get disconnected while you are bringing up z/VM.

If you do not enter this command and you get disconnected, you must reload the Control Program to regain control of the z/VM system console.

2. The DISABLE option also keeps the 3270 printers from being enabled. To reenable them later, enter the appropriate ENABLE commands. See “Making a Display, a 3270 Printer, or Communication Lines Available for User Access” on page 86.
3. If you do not include the DISABLE option, any display, 3270 printer, or communication lines that were enabled the last time you shut down the system are automatically enabled when you finish bringing it back up.

Bringing Up the System Without Logging on Users

Usually, CP autologs these seven virtual machines after the operator's virtual machine has been initialized:

- Two accounting virtual machines
- Two EREP virtual machines
- Two symptom record virtual machines
- The start up virtual machine (usually AUTOLOG1).

You can specify the NOAUTOLOG option if you wish to have CP initialize only the operator virtual machine, so you can make some changes to the system before continuing with normal initialization.

Bringing Up the System Without a User Directory (NODIRECT)

If there is a problem with the z/VM user directory (for example, its data is incorrect), CP may not be able to perform a start. When this happens, contact your installation's support personnel before you proceed.

After your installation's support personnel have performed any required diagnosis, go back to the processor complex's system console and reload z/VM into storage.

For Non-SSI Systems

When the system is reloaded, CP again prompts you for the type of start you want to perform. To bring up the system without accessing the user directory, respond to CP's start prompt with the NODIRECT option and either WARM, FORCE, COLD, or CLEAN. For example, if you want to perform a warm start without using the directory, enter:

```
warm nodirect
```

This response prevents CP from trying to access the user directory. When the system comes up, you must recover the user directory before you do anything else. To find out how to recover the user directory, see “Recovering the User Directory” on page 47.

For SSI Systems

SSI systems must be handled differently than non-SSI systems when recovering the directory. You must follow these steps when reloading the system:

1. IPL the system with a load parameter, which will cause the Stand Alone Program Loader (SAPL) screen to be displayed.
2. On SAPL, specify REPAIR in the IPL PARAMETERS section. This will cause your system to be IPLed in non-SSI mode. Use the PF10 Load function to continue the IPL. Respond to the rest of the IPL prompts as described in “For Non-SSI Systems” on page 33. In addition, you will get additional warnings related to the REPAIR option. Read these warnings and respond to the prompts appropriately.
3. After you have successfully recovered the directory, reIPL your system normally (that is, without the REPAIR option). Your system will now be running as an SSI system.

Setting the Time-of-Day Clock

After you indicate the type of start you want CP to perform, CP does one of three things:

- If you are bringing up the system for the first time since the last initial microprogram load (IML) or power-on reset, or if the time-of-day (TOD) clock is not within the years 1942 - 2041, CP sends you the following prompt so you can set the time-of-day (TOD) clock, enter one of these:

  ```
  Set date MM/DD/YY or MM/DD/YYYY or YYYY-MM-DD (valid years are 1942 - 2041)
  ```

  Enter the current date in the format that appears in the prompt (a *YY* of 42 to 99 corresponds to 1942 - 1999, a *YY* of 00 to 41 corresponds to 2000 - 2041). For example, if the date is January 10, 2001, enter one of these:

  ```
  01/10/01
  01/10/2001
  2001-01-10
  ```

  CP then sends you this prompt:

  ```
  SET TIME HH:MM:SS
  ```

  If your system configuration file contains TIMEZONE_DEFINITION and TIMEZONE_BOUNDARY statements, CP may determine that one of the following is true:

  – The time and date combination you have entered is impossible.

    If your system configuration file indicated that on the date you entered, the time zone changes so that at 2:00 a.m. the clock changes to 3:00 a.m. and you entered 2:30 a.m. as the time, CP issues an error message and prompts you for date and time again.

  – The time and date combination you have entered is ambiguous.

    If your system configuration file indicates that on the date you entered, the time zone changes so that at 2:00 a.m. the clock changes back to 1:00 a.m. and you entered 1:30 a.m., CP displays a list of possible time zones and asks you to choose the timezone that 1:30 a.m. corresponds to.

  After you press ENTER, CP sends you the following prompt:

  ```
  NOW 16:15:30 tmz day-of-week 1998-02-06
  CHANGE TOD CLOCK (YES|NO)
  ```

  Where:

  tmz
  specifies the time zone your installation specified on the TIMEZONE_DEFINITION statement in the system configuration file during system generation.

  day-of-week
  specifies the day of the week

  If the TOD clock value is correct, press ENTER or enter NO, and CP continues initializing. If the TOD clock value is not correct, enter YES. CP then repeats the sequence for setting the TOD clock.

- If you are not bringing up the system for the first time since the last IML or power-on reset, CP sends you this prompt:

  ```
  NOW hh:mm:ss tmz day-of-week yyyy-mm-dd
  CHANGE TOD CLOCK (YES|NO)
  ```

  Where:

  hh:mm:ss
  specifies the time

  tmz
  specifies the time zone your installation specified on the TIMEZONE_BOUNDARY statement in the system configuration file during system generation

day-of-week
specifies the day of the week

yyyy-mm-dd
specifies the date

CP uses the time zone offset in TIMEZONE_DEFINITION to convert the time you enter into Coordinated Universal Time (UTC) and stores it as the TOD clock.

Unless your installation specified a different time zone on the TIMEZONE_DEFINITION statement in the system configuration file, CP interprets the time you specify for this prompt as UTC.

If the time and date are correct, press ENTER or enter NO, and CP continues initializing. If either value is wrong, enter YES. CP then prompts you for the correct data.

- If your hardware is enabled and configured for Server Time Protocol (STP) and this feature has been enabled in your configuration file (using the STP_TIMESTAMPING or STP_TZ features statements), you will not receive any prompts to change the TOD clock. Instead, STP is automatically used to initialize the TOD clock with the STP-coordinated TOD value. When the TOD clock is initialized, you will receive message HCP986I "TOD Clock Synchronized via STP."

Note:

1. During system generation, unless your installation specified a different time zone on the TIMEZONE_DEFINITION statement in the system configuration file, CP interprets the time you specify as Coordinated Universal time (UTC).

What CP Does during Initialization

After you finish setting any TOD clocks on the processor complex, CP continues initializing and logs on the primary system operator. CP also logs on the AUTOLOG1 virtual machine, which your installation may use to perform such initialization tasks as starting up other virtual machines, attaching devices to them, and restoring spool files and system data files from tape. AUTOLOG1 is the default user id. You can change the AUTOLOG1 user ID by using the STARTUP operand on the SYSTEM_USERIDS statement in the system configuration file.

If a problem with the z/VM user directory prevents CP from initializing that directory, CP searches all other system volumes until it locates a valid directory. If there is no alternate directory or CP cannot initialize the new directory, you must reload the system and specify the NODIRECT option. To find out how to do this, see “Bringing Up the System Without a User Directory (NODIRECT)” on page 33.

Even if a problem with the z/VM user directory does not prevent CP from initializing, it may prevent CP from logging on the primary system operator using the normal directory definition. In this case, CP logs on the primary system operator under the user ID specified on the OPERATOR operand on the SYSTEM_USERIDS statement in the system configuration file. If you are the primary system operator, you must recover the user directory before you do anything else. To find out how to recover the user directory, see “Recovering the User Directory” on page 47.

During initialization, CP reads the volume identifiers of all the volumes that are mounted and compares them with the system and user minidisk volumes your installation specified during system generation. If any of the volumes that were requested are not mounted (and therefore could not be attached), CP sends you a message. If any of the missing volumes are needed by the system or contain user minidisks, mount and attach them to the system as soon as possible. See “Attaching a System Volume to the System” on page 111.

After CP finishes initializing, other users can log onto z/VM. You can also now start up required service virtual machines and bring up the production system.

Note: If you have just performed a cold start and not all the system data files (NSSs, DCSSs, image libraries, message repository files, or system trace files) are re-created, use the SPXTAPE command to restore the files to the system as soon as possible. For more information on saving and restoring these files, see “Backing Up Standard Spool Files and System Data Files” on page 214.

Starting Service Virtual Machines

After CP is initialized, you can start the service virtual machines. These virtual machines provide system services and include the:

- Accounting virtual machine
- Error recording (EREP) virtual machine
- Symptom record recording virtual machine
- Virtual machines required by z/VM facilities and optional features
- Virtual machines required by supported licensed programs
- DFSMS/VM service virtual machine.

Starting Accounting, Error, and Symptom Record Recording Virtual Machines

If accounting, error, or symptom record recording is active on z/VM, you may need to log on and start the accounting, error, or symptom record recording virtual machines. However, installations usually set up the user directory entries and CMS PROFILE EXECs for these machines so that they start automatically.

For information about how to start the accounting, error, or symptom record recording virtual machines, see "Collecting Accounting Information" on page 183, "Collecting Hardware Error Information" on page 190, or "Collecting Symptom Record Information" on page 202.

Starting Virtual Machines for z/VM Facilities and Optional Features

If you are using any of the following z/VM facilities and optional features, you need to start the service machines they require:

- DirMaint™
- Performance Toolkit for z/VM
- RSCS
- TCP/IP
- Virtual Machine Resource Manager

For information about when and how to start these service virtual machines, see the publications listed in the "Bibliography" on page 279.

Starting Licensed Program Virtual Machines

Some licensed programs require you to operate service virtual machines. For example, if your installation has installed any of the following licensed programs, you need to start the service virtual machines they require:

- VM/Pass-Through
- VTAM® Communications Network Applications (VCNA)
- VM/Virtual Telecommunications Access Method (VM/VTAM).

For information about when and how to start these service virtual machines, see the appropriate publications for these programs.

Using Link Modes to Ensure Minidisk Stability for Service Virtual Machines

DFSMS/VM is the foundation for system-managed storage in the z/VM environment. A DFSMS/VM service machine uses the Interactive Storage Management Facility (ISMF) to reduce the amount of manual work that must be performed when data is migrated to newly installed DASD. ISMF formats new minidisks, updates user directories, and automates other functions, providing both speed and predictability during data migration activities. You can use the DFSMS/VM virtual machine to provide information about storage management tasks. Certain applications are dependent on the stability of a given minidisk. For example, a virtual machine used for moving data from one system to another or for managing data bases requires this minidisk stability.

Enhancements to the CP LINK command, consisting of link access modes, guarantee data security and stability. When any one of these enhanced link modes is in effect for a given minidisk, no other user ID on the executing system, or on any other system sharing access to the minidisk in a single system image (SSI) cluster or through cross-system link (XLINK), can acquire write access to the minidisk. The following link modes are provided in addition to read/write access:

- Stable read (SR).

  An authorized user who obtains this link mode locks the target minidisk against all write link requests from all users on the executing system. A user attempting to establish W, M, MR, or MW links receives a *not linked* message, indicating the minidisk is locked against all write access requests until the user releases the minidisk.

  When there is a critical need to establish a stable link, use the stable mode to access the minidisk.
- Stable write (SW,SM).

  An authorized user who obtains this link mode locks the target minidisk against all write link requests from all users on the executing system. A user attempting to establish W, M, MR, or MW links receives a *not linked* message, indicating the minidisk is locked against all write access requests until the user releases the minidisk.

  When there is a critical need to establish a stable link, use the stable mode to access the minidisk.
- Exclusive READ (ER).

  An authorized user who has obtained this link mode locks the target minidisk against all link requests from all users on the executing system. A user attempting to establish R, RR, W, WR, M, MR, or MW links fails and receives a message indicating the minidisk is temporarily locked against all requests for access.
- Exclusive WRITE (EW).

  An authorized user who has obtained this link mode locks the target minidisk against all link requests from all users on the executing system. A user attempting to establish R, RR, W, WR, M, MR, or MW links fails and receives a message indicating the minidisk is temporarily locked against all requests for access.

Note:

1. Full-pack minidisks are a special case. A read/write full-pack minidisk and a read/write user's A-disk can share the same volume on an assigned DASD and, at the same time, can be held in EW mode. In this case, SW and EW modes do not stop overlapping and conflicting accesses.

   To prevent conflicting access and to provide full data integrity, you must take the following precautions:

   - Do not define overlapping minidisks.
   - Do not grant DEVMAINT authority to allow fullpack overlays.
   - Do not use DEFINE MDISK.
2. If an external security manager (ESM) is installed on your system, you may not be authorized to use the LINK command for all minidisks and all access modes. The ESM may downgrade certain requests for write access to read access. For additional information, contact your security administrator.

Bringing Up a Production System in a Virtual Machine

After you start any service virtual machines that need to be started, you are ready to bring up a production system in a virtual machine.

Although there are many ways to set up the consoles, this book assumes that you have access to the following separate consoles:

- Processor complex's system console
- z/VM primary system console
- Console for the production system's virtual machine
- Production system's system console.

In the following explanation, bringing up the production system requires these consoles:

- Console for the production system's virtual machine.

  You use this console to control the production system's virtual machine. Log on the production system's virtual machine, set up its running environment, and enter an IPL command.
- Production system's system console.

  This is the system console from which you control the production system. It is dedicated to the production system's virtual machine.
- z/VM system console.

  If you need to dedicate devices to the production system's virtual machine before you bring it up, you enter the appropriate CP commands from this console.

Note:

1. To eliminate a console, your installation may configure the consoles so that the console for the production system's virtual machine and the production system's system console are the same physical display. You then operate the production system in full-screen mode. See the *z/VM: Virtual Machine Operation* for an explanation of how to enter commands in this environment.
2. For Linux guests, you might want to automate the loading of Linux whenever its virtual machine is logged on. See *z/VM: Getting Started with Linux on IBM Z* for more information.

To bring up the production system, follow these steps:

1. Log on the production system's virtual machine.
2. Load (IPL) the production system in its virtual machine.
3. Finish bringing up the production system.
4. Optionally, disconnect the production system's virtual machine.

These steps are described in the sections that follow.

Logging on a Production System's Virtual Machine

To log on a production system's virtual machine to z/VM:

1. Make sure the console you will use to log on the production system's virtual machine is enabled.

   To find out the console's status, enter:

   ```
   query rdev
   ```

   where *rdev* is the real device number of the display you are using as the console for the production system's virtual machine.

   If the status indicates that the display is enabled, go to step “2” on page 39.

   If the status indicates that the display is disabled, enter:

   ```
   enable rdev
   ```

   If the status indicates that the display is offline, enter:

   ```
   vary on rdev
   enable rdev
   ```

2. Go to the console you plan to use to log on the production system's virtual machine. If the device is enabled, you see the z/VM logo. If you do not see the logo, go back to step "1" on page 38.
3. On the z/VM logo, enter the user ID of the production system's virtual machine. The cursor is positioned at the correct place for you to enter the user ID.
4. When CP prompts you, enter the logon password.

   CP responds with the system log message, messages about devices that are being attached, and a message that the production system's virtual machine is logged on.
5. Watch the CP messages to see that all the devices you need to run the production system are attached. If any devices are missing, go back to the z/VM system console and enter the appropriate CP commands to attach them to the production system's virtual machine.

   Normally, your installation dedicates quite a few devices to the production system's virtual machine. It does this by putting DEDICATE statements in the user directory entry for the production system's virtual machine. During logon, CP issues messages that tell you when these dedicated devices are not attached. You should watch these messages to see that all the devices you need to run the production system are attached. If they are not, you must attach them by entering CP commands. You can find information on these commands in Chapter 6, "Controlling Local Devices," on page 81.
6. If necessary, go back to the console where you logged on the production system's virtual machine. If your installation requires you to set up a special running environment for the production system's virtual machine, enter the appropriate CP commands now.

   For example, your installation may require you to enter the SET RUN ON command before you bring the production system up in its virtual machine. For more information about setting up appropriate running environments for particular production operating systems, see the *z/VM: Running Guest Operating Systems*.

Loading (IPLing) a Production System in a Virtual Machine

You have just logged on a production system's virtual machine at a console and set up its running environment. To load (IPL) the production system, do the following:

1. Decide which of the following ways is most appropriate for you to load the production system:

   - If your production system is a VSE system, enter:

     ```
     ipl vdev clear
     ```

     where *vdev* is the virtual device number of the device on which the VSE system resides.
   - If you want to pass a console address to the Stand-Alone Program Loader (SAPL) or to specify an alternate z/OS nucleus, enter:

     ```
     ipl vdev loadparm parameter
     ```

     where *vdev* is the virtual device number of the device on which the production system resides, and *parameter* is the 1 - 8 character load parameter.

     Example—Specifying a Console Address for SAPL: The details of the following example are relevant only if you are IPLing a volume on which SAPL has been installed and you want to specify a console address to activate the fullscreen option of SAPL.

     If you are IPLing the virtual device B00 on which SAPL has been installed and you have your virtual console defined at virtual address 009, you can enter:

     ```
     ipl B00 loadparm 9
     ```

     SAPL determines that a console address has been specified with the LOADPARM option and displays a fullscreen menu on your console. You can choose what object you want SAPL to load and where in storage it should be placed by modifying fields on this screen.

Example—IPLing an Alternate z/OS Nucleus: The details of the following example are relevant only if you are IPLing z/OS and wish to specify an alternate nucleus.

If you are IPLing z/OS from virtual device 292 and you want to use an alternate nucleus called IEANUC04, rather than the default nucleus IEANUC01, you enter:

```
ipl 292 loadparm 4
```

This passes 4 as data to the z/OS Control Program that you are IPLing. z/OS takes this data and appends it to IEANUC01 to form the alternate nucleus name, IEANUC04.

- If none of the above apply, enter:

  ```
  ipl vdev
  ```

  where *vdev* is the virtual device number of the device on which the production system resides.

2. Go to the production system's system console and wait for the production system to come up on it. *Do not type anything* on the console where you entered the IPL command until you see the first message on the production system's system console.

Note: For z/OS guests, when the system console is responding to z/VM errors or IPLing the system, the virtual machine console can be used as an IPL and error-recording console. For more information see the *z/VM: Virtual Machine Operation*.

Finishing Bringing Up the Production System in a Virtual Machine

After you see the first message on the production system's system console, bring up the production system normally. From this point on, you enter CP commands from the z/VM system console and production system commands from the production system's system console.

Unless z/VM abnormally terminates, you do not use the console where you logged on the production system's virtual machine and entered the IPL command again. If you want to use the console for some other virtual machine, you can disconnect the production system's virtual machine by entering at that console:

```
#cp set run on
#cp disconnect
```

The production system continues to run normally.

Note that in the preceding commands, the pound sign (#) represents the logical line-end symbol in effect for your virtual machine. If you have redefined the logical line-end symbol, substitute your symbol for the #. If you do not know the line-end character, use the CP QUERY TERMINAL command to display it.

The SET RUN ON command keeps the production system's virtual machine running and helps prevent it from being logged off by CP. The DISCONNECT command disconnects the console from the virtual machine and allows you to use the console for another purpose. For more information about the interaction of these two commands, see the *z/VM: CP Commands and Utilities Reference*.

Checking the Initialization of System Resources

At this point, both z/VM and all production systems are running. Now make sure that all resources are available as your installation requires. This includes checking your system's devices and programs, as follows:

1. Make sure that all devices are allocated to the right virtual machines.

   To check the status of real devices attached to the system, enter one of CP's QUERY commands for real devices from the z/VM system console. For more details, see "Checking the Status of Real Devices" on page 41.

2. Make sure that all devices allocated to each production system's virtual machine are available to that production system.

   To do this, enter the appropriate commands from the production system's system console.

3. Make sure that all z/VM service virtual machines that are supposed to run are running.

   To check the status of the accounting, error recording (EREP), and symptom record recording virtual machines, enter the CP QUERY RECORDING command. For more details, see "Finding Out the Status of Accounting" on page 185, "Finding Out the Status of Error Recording" on page 193, or "Finding Out the Status of Symptom Record Recording" on page 203.

   To check the status of the service virtual machines required by z/VM facilities and features, such as TCP/IP, DirMaint, and RSCS, use the commands appropriate to these facilities and features. For example:

   - To display information about the devices and links defined for the TCP/IP virtual machine, issue the NETSTAT DEVLINKS command from the TCPMAINT user ID. To display information about the gateways known by the TCP/IP server, issue the NETSTAT GATE command. If TCP/IP is not started, you will receive an IUCV error message. For information about the NETSTAT command, see *z/VM: TCP/IP User's Guide*.
   - To display information about the RSCS system, issue the RSCS QUERY SYSTEM command. There are several variations of this command; if you do not specify any additional operands, the default is QUERY SYSTEM LINKS. If you issue the command from a user ID other than the RSCS console operator, you must preface the RSCS command with SMSG *rscsid*, where *rscsid* is the name of the RSCS server. For information about RSCS QUERY commands, see *z/VM: RSCS Networking Operation and Use*.

   To check the status of the licensed program service virtual machines, use the commands appropriate to those products. Supported licensed programs that require service virtual machines include VM/Pass-Through.

4. Make sure that all production system applications that are supposed to run are running.

   To do this, enter the appropriate commands from the production system's system console.

5. Make sure that all remote lines are operating.

   To check remote lines to z/VM, use RSCS commands as appropriate.

   To check remote lines to the production system, enter the appropriate production system commands.

Checking the Status of Real Devices

You can use a CP QUERY command to check the status of your real devices. For example, you can find out which:

- Devices are in use, are available for use, or are offline
- Devices are in use by CP or are dedicated to a virtual machine
- DASDs are system volumes
- Printers are associated with a security label.

To check the status of all of your real devices of a specific type, enter one of the following commands:

```
query dasd all
query tapes all
query ur all
query graf all
query lines all
query ctca all
query swch all
query osa all
```

CP responds with the status of all of your real DASD, tape drives, unit record devices, displays, 3270 printers, communication lines, channel-to-channel adapters, dynamic switching devices, and open systems adapters, respectively. Unit record devices include card readers, printers, and card punches.

Note: Since open systems adapter (OSA) devices cannot come online until the OSA hardware is fully initialized, there may be a delay between the completion of system IPL and the availability of the OSA devices.

The status CP displays for a device depends on its type. For example, CP displays different kinds of status for a printer than for a DASD volume. But CP always displays the device's type (such as DASD or tape), its real device number, and one or more keywords that describe its status. Additional information may follow the keywords.

A keyword indicates the specific status of a device. A device also falls into one of three general groups: active, free, or offline. These general groups correspond to operands you may specify on the QUERY command: ACTIVE, FREE, or OFFLINE. An *active* device is either in use by CP or is dedicated to a virtual machine. A *free* device is available to CP, but is not now in use by either the system or a specific virtual machine. An *offline* device is unavailable to CP.

Note: The BOXED operand indicates if the device is unavailable. If the device is unavailable, it is usually because of a hardware condition. The System Operator should physically isolate the device from the I/O subsystem. When the condition has been repaired, use the VARY ONLINE command to initiate recovery.

Table 1 on page 43 lists the keywords that appear in responses to QUERY commands for real devices. Use the table to find out the meaning of a keyword and the types of devices for which CP displays it. The table also shows whether the keyword indicates:

- An active, free, or offline device
- A dedicated device or a device in use by CP.

To find out how to change the status of a device, see Chapter 6, “Controlling Local Devices,” on page 81.

The ALL operand, in combination with a device type operand, allows you to check the status of all your real devices of a specific type. You can also use other operands on the QUERY commands for real devices. For example, if you want to find out the status of all of the active tape drives, enter:

```
query tapes active
```

If you want to find out the status of all the devices within a specific range of device numbers, enter:

```
query rdev1-rdevn
```

where *rdev1* is the real device number of the first device in the range and *rdevn* is the real device number of the last device in that range.

If you want to know the status of only one device, enter:

```
query rdev
```

where *rdev* is the real device number of the device.

If you want to know the status of the DASD containing a particular volume, enter:

```
query dasd volid
```

where *volid* is the volume identifier. If there is more than one volume with the same identification, CP displays the status of all of the DASDs that contain the volume.

Note: In the following discussions, responses will include cylinders for CKD and ECKD™ devices and pages for FBA devices.

To display a list of CP-owned system volumes, enter:

```
query alloc all
```

In response, CP lists all of the CP-owned volumes that contain temporary disk, directory space, paging, or spooling space. For each volume, CP displays the number of cylinders or pages that are allocated, in use, and available for temporary disk, directory space, paging, and spooling space. CP also displays whether the volume contains the active CP directory.

To display the list of all CP-owned system volumes containing paging space, enter:

```
query alloc page all
```

In response, CP lists all of the CP-owned volumes that contain paging space. For each paging extent on each volume, CP displays the starting cylinder or page and ending cylinder or page, the total number of pages, the number of pages in use, the highest page in use, and the percentage of pages in use.

To display the list of all CP-owned system volumes containing spooling space, enter:

```
query alloc spool all
```

In response, CP lists all of the CP-owned volumes that contain spooling space. For each spooling extent on each volume, CP displays the starting cylinder or page and ending cylinder or page, the total number of pages, the number of pages in use, the highest page in use, and the percentage of pages in use.

To display the list of all CP-owned system volumes containing temporary disk space, enter:

```
query alloc tdisk all
```

In response, CP lists all of the CP-owned volumes that contain temporary disk space. For each temporary disk extent on each volume, CP displays the starting cylinder or page and ending cylinder or page, the total number of cylinders or pages, the number of cylinders or pages in use, the highest cylinder or page in use, and the percentage of cylinders or pages in use.

To display the list of all CP-owned system volumes containing directory space, enter:

```
query alloc drct all
```

In response, CP lists all of the CP-owned volumes that contain directory space. For each allocation map extent on each volume, CP displays the starting cylinder or page and ending cylinder or page, the total number of cylinders or pages, the number of cylinders or pages in use, the highest cylinder or page in use, and the percentage of cylinders or pages in use.

To display the allocation mapping of all CP-owned system volumes, enter:

```
query alloc map all
```

In response, CP lists all of the CP-owned volumes that contain temporary disk, directory, paging, or spooling space. For each extent on each volume, CP displays the starting cylinder or page and ending cylinder or page, the total number of cylinders or pages in use, the highest cylinder or page in use, the percentage of cylinders or pages in use, and the allocation type of the extent.

Table 1. Status of Real Devices

Device Status Keyword	Keyword Meaning	Device Type	Type of Status
ASSIGNED SYSTEM	The device is available for a dump, an SPXTAPE function, or a TRSAVE function.	Tape drive	Active (CP)
ATTACHED TO	The device is dedicated to the indicated virtual machine.	Any	Active (dedicated)
BOXED	The device is unavailable to users.	Any	Boxed
CP OWNED	The device is used by the system for paging, directory, temporary disk space, and spooling activity.	DASD	Active (CP)
CP SYSTEM	The device contains user minidisks.	DASD	Active (CP)

Table 1. Status of Real Devices (continued)

Device Status Keyword	Keyword Meaning	Device Type	Type of Status
DIALED TO	The device is dedicated to the indicated virtual machine through the DIAL command.	3270 display, communication lines	Active (dedicated)
DISABLED	The device is unavailable to users.	Display, 3270 printer, communication lines	Free
DRAINED	The device is unavailable for processing spool files.	Unit record device	Free
DRAINING	The device will become unavailable for processing spool files after finishing the current file.	Unit record device	Active (CP)
ENABLED	The device is available to users but is not in use.	Display, 3270 printer, communication lines	Free
FREE	The device is available but is not in use.	DASD, CTCA, tape drive	Free
INT REQ	The device requires operator intervention.	Unit record device	Active (CP)
INT REQUIRED	The device requires operator intervention.	Any	Active (CP or dedicated)
LOGON AS	A user has logged on to the device.	Display, communication lines	Active (CP)
MOUNT REQ	A printer or card punch is waiting for you to change the paper or cards (mount a new form).	Card punch, printer	Active (CP)
OFFLINE	The device is unavailable to CP.	Any	Offline
PRINTING	The device is printing a spool file.	Printer	Active (CP)
PUNCHING	The device is punching a spool file.	Card punch	Active (CP)
READING	The device is reading a card deck.	Card reader	Active (CP)
SEC	Specifies the security label of files that can be printed on the printer. This is used only when an ESM is installed and security label checking is enabled. If a security label is not assigned to the printer, the SEC field is followed by the word NONE. No files can be printed until the printer is STARTed with a valid security label.	Printer	Active (CP)
SETUP REQ	An impact printer is waiting for you to check the alignment of the paper on the printer.	Printer	Active (CP)

Table 1. Status of Real Devices (continued)

Device Status Keyword	Keyword Meaning	Device Type	Type of Status
STARTED	The device is available for processing spool files.	Unit record device	Active (CP)
SUBCHOFF	The subchannel for this device is disabled.	Any	Offline
Volume identifier	The device is available but is not in use.	DASD	Free

Preparing the System for a Possible Failure

The final thing you need to do to finish bringing up the z/VM system is to prepare the system for a possible failure. This includes the following tasks:

- Setting up the system abend dump environment
- Making sure CP tracing is active
- Getting ready to take a CP stand-alone dump
- Making sure error recording is on
- Making sure symptom record recording is on
- Making sure CP is logging activity at your operator's console
- Getting ready to recover the production system.

Setting Up the System Abend Dump Environment

Unless your installation requires you to do otherwise, you should make sure z/VM's abend dump environment is set up so that:

- A dump device is defined.
- CP tries to restart z/VM after an abend occurs.

To find the current dump environment settings, enter:

```
query dump
```

CP's response indicates:

- The type of dump device and its real device number(s), or that the dump device is undefined (SET OFF)
- Whether CP tries to restart the system after it takes the dump (IPL or NOIPL)
- How much spool space is allocated for a system dump (PAGES).
- The settings for other operands that are allowed on the SET DUMP command.

Make sure the dump device is defined. If the dump SET OFF, if the wrong device is defined, or if the response says NOIPL, you need to define the correct device or appropriate options.

To set the dump device to a DASD, enter:

```
set dump dasd ipl
```

To set the dump device to multiple DASDs, be sure to use the correct device numbers for your installation. Enter:

```
set dump rdev1 rdev2 rdevn ipl
```

where *rdev1*, *rdev2*, and *rdevn* are the real device numbers of the DASDs your installation uses for dump devices.

If you need more information on the options you can specify on the SET DUMP command, see the *z/VM: CP Commands and Utilities Reference*.

Making Sure CP Tracing Is Active

Unless your installation directs you to do otherwise, make sure CP tracing is active.

When CP tracing is active, CP records information about various system events in one or more CP trace tables in storage. When a system abend occurs, CP includes the trace tables in the abend dump. The detailed history of system activity in the CP trace table helps support personnel diagnose the cause of the abend.

When CP tracing is not active, CP does not record this information, and it is more difficult to diagnose the cause of a system failure. Therefore, you should make sure CP tracing is active.

When you bring up z/VM, CP tracing is active. To make sure that CP is tracing all events, enter:

```
query cptrace for all
```

If CP is tracing all events, it displays the following response:

```
FOR NONSPECIFIC
ALLCODES ENABLED
NO SPECIFIC TRACE SET
```

If CP displays something else, you can cause CP to trace all events by entering:

```
set cptrace for specific reset
set cptrace for all on
```

For more information about the QUERY CPTRACE and SET CPTRACE commands, see the *z/VM: CP Commands and Utilities Reference* and the *z/VM: Diagnosis Guide*.

Getting Ready to Take a CP Stand-Alone Dump

Make sure your installation has installed the z/VM stand-alone dump utility on DASD. You should know on which volumes it is installed. You should also know which device is the IPL device and which are additional dump devices. If you do not know, contact the person at your installation who last generated the stand-alone dump program.

For information about using the stand-alone dump utility, see *z/VM: CP Planning and Administration*.

Making Sure Error Recording Is Active

CP creates environmental error record editing and printing (EREP) records to record hardware errors. These records can help support personnel diagnose hardware problems. Accordingly, when you bring up z/VM, make sure that error recording is active.

To find out how to check the status of error recording and what to do if it is inactive, see “Finding Out the Status of Error Recording” on page 193.

Making Sure Symptom Record Recording Is Active

CP includes symptom records in every dump. The symptom record service machine can place these records in a separate CMS file. This can help support personnel identify known problems and their fixes without first reading the dumps. Accordingly, when you bring up z/VM, make sure that symptom record recording is active.

To find out how to check the status of symptom record recording and what to do if it is inactive, see “Collecting Symptom Record Information” on page 202.

Making Sure CP Is Logging Operator Console Activity

When you bring up z/VM, CP normally records activity at the primary system operator's console in a console log. The console log contains all the commands you enter and all the responses you receive while you are logged on to the operator's virtual machine. It also contains any error or informational messages that CP sends you. If the system abends, CP saves the console log in a spool file, which you can print later. Information in the console log can help support personnel determine the cause of the abend. Therefore, make sure CP is recording activity at your operator's virtual console in a console log.

To find out whether CP is logging your console's activity, enter:

```
query virtual console
```

On the first line of CP's response, you will see either START or STOP. If it says START, CP is already logging your console activity.

If it says STOP, enter:

```
spool console start
```

CP then starts logging the activity at your operator's console.

For more information on starting and printing a console log, see "Collecting Information about System Events" on page 179.

Note: Some installations may not define a virtual console for the primary system operator's virtual machine. If you enter the QUERY VIRTUAL CONSOLE command, and your virtual machine has no virtual console, CP responds that it cannot find your console. In this case, you cannot start a console log unless you enter:

```
define console as vdev
```

where *vdev* is the virtual device number you want to assign to the virtual console.

Once you have defined a virtual console, you can check to make sure CP is logging its activity by using the QUERY VIRTUAL CONSOLE command.

Getting Ready to Recover a Production System

If CP abnormally terminates, you must know how to recover your production systems. If you do not know how to recover a production system, see Chapter 7, "Responding to Errors," on page 167.

You are now finished bringing up the z/VM system.

Responding to Errors during Bring-Up

This chapter has already discussed most of the errors you are likely to encounter. Being unable to perform a warm start is one example. But one other error you may run into requires you to recover the z/VM user directory.

Recovering the User Directory

If there is a problem with the z/VM user directory, for example, data is incorrect, CP may not be able to initialize that directory. When this happens, contact your installation's support personnel before you proceed. Then bring up the system without the user directory. See "Bringing Up the System Without a User Directory (NODIRECT)" on page 33. After you do this, CP logs on the primary system operator without trying to use the operator's user directory definition. Instead, CP uses a default definition of the operator's virtual machine.

If the problem with the user directory does not prevent CP from initializing that directory, it may still prevent CP from using the directory definition to log on to the primary system operator. Again, CP uses a default definition of the operator's virtual machine.

If you are the primary system operator, CP logs you on under these conditions as the default system operator. Because you are the default system operator, your virtual machine has the following characteristics:

- Your user ID is the one your installation specified on the OPERATOR operand on the SYSTEM_USERIDS statement in the system configuration file.
- You have all command privilege classes, A through G.
- You have 2047 megabytes (MB) of virtual storage.
- Your virtual console is a virtual 3270 with virtual device number 001.

If you are the default system operator, you are authorized to perform all the tasks you normally perform. However, you are not linked to other virtual machines' minidisks, where data you may need to access is stored. For example, you may need to access the EREP records stored on the error recording virtual machine's A disk.

As the default system operator, you must recover the user directory before you do anything else.

Preparing to Recover the User Directory

Preparing to recover the user directory involves the following tasks:

- Getting a printed copy of the directory entries for the following minidisks:
  - The minidisk containing the CMS system disk
  - The minidisk containing the directory source file
  - The minidisk containing the system configuration file.
- Preventing new users from linking to these minidisks
- Removing existing links to the minidisks
- Setting up your virtual machine.

Before you start to recover the directory, you must:

1. Get a printed copy of the directory entry for:
   - The CMS system disk (usually MAINT 190)
   - The minidisk containing the CMS file that is used to update the directory.

   You must also find out the volume identifier and the virtual device number assigned to the DASD pack that contains the object directory. This information is specified on the DIRECTORY control statement (which is the first statement in the directory source file) and is necessary to recover the user directory.
2. Find out the real device number of your virtual console. Enter:

   ```
   query userid
   ```

   where *userid* is your user ID. CP responds with your user ID and the real device number of your virtual console.
3. Prevent any new users from logging on. Enter:

   ```
   disable all
   ```

4. Reenable your virtual console:

   ```
   enable rdev
   ```

   where *rdev* is the real device number of the display you are using for your virtual console.

   If you do not enter this command and somehow you are disconnected, you cannot log back on. You will have to reload the system to log on again.

5. Using information in the printed copy of the directory, find out the real device numbers of the DASDs that contain the following minidisks:

   - Minidisk containing the CMS system disk (usually MAINT 190)
   - Minidisk containing the directory source file
   - Minidisk containing the system configuration file.

   For each volume, enter:

   ```
   query dasd volid
   ```

   where *volid* is the volume identifier of each volume.

   CP's response indicates the real device number, the volume identifier, and the number of links to minidisks contained on the volume. Record the real device numbers and note whether there are links to any of the volumes.

6. If there are no links to any of the volumes, proceed to step “9” on page 49.

   Otherwise, find out which users are linked to minidisks on the volumes. Later in this procedure, enter the DEFINE MDISK command for each of the volumes. If there are any links to the volumes, the DEFINE MDISK command does not complete successfully.

   To find out which users are linked to minidisks on a volume, enter:

   ```
   query system rdev
   ```

   where *rdev* is the real device number of the DASD on which the volume resides.

   CP's response lists all of the users who are linked to minidisks on the volume and the virtual device numbers each user assigned to the minidisks. Note both the users and the virtual device numbers.

7. Ask all users to detach the minidisks from their virtual machines. For example, for each volume, enter:

   ```
   warning all problem with the user directory
   warning all please detach link to volid
   ```

   where *volid* is the volume identifier of the selected volume.

   Enter QUERY SYSTEM commands to determine when all users have detached their minidisks on the three volumes.

8. If users do not detach their minidisks in a reasonable amount of time, try to log off each of the users by entering:

   ```
   force userid
   ```

   where *userid* is the ID of a user who still has a link to one of the minidisks.

   If you are unable to log off the users with the FORCE command, contact your support personnel. At this point, you must perform a system restart from your processor complex's system console. A system restart produces a dump that your support personnel may wish to use to diagnose the problem with the FORCE command.

9. Enter the following command to ensure that your virtual console is operating in VM mode:

   ```
   terminal mode vm
   ```

10. Make sure you have enough storage to IPL CMS and to update the user directory.

    Later in the procedure, IPL CMS (as 190). You must have 20 MB of virtual storage to do so. This should also be enough storage to allow you to use the DIRECTXA command successfully. If you IPL a named system, (CMS) 16 MB of storage may suffice. However, if you IPL 190, then you will need 20 MB of storage.

    To find out how much storage you have in your virtual machine, enter:

```
query virtual storage
```

If you want to change this amount to 20 MB, enter:

```
define storage 20m
```

Note: If you cannot define enough storage for your virtual machine, reload the system with the NODIRECT option. See "Bringing Up the System Without a User Directory (NODIRECT)" on page 33. The NODIRECT option gives you (as the default system operator) 2047 MB of virtual storage, which should be enough to update the directory.

You are now ready to start the directory update process.

Updating the User Directory

Updating the user directory requires you to:

- Define required minidisks
- Bring up CMS in your virtual machine
- Change, if necessary, the directory source file
- Use the DIRECTXA command
- Restore normal system operation.

1. Define the required minidisks.

   Using the information contained in the printed copy of the directory entries you obtained earlier, define:

   - The minidisk containing the CMS system disk
   - The minidisk containing the directory source file
   - The minidisk containing the system configuration file (this is where the new directory is placed).

   Note: If you have problems defining any of the minidisks, go back to step "5" on page 49.

   To define the minidisk containing the CMS system disk, enter:

   ```
   define mdisk as 190 start count volid
   ```

   where *start* is the decimal number that defines the cylinder or block on the real disk that corresponds to the first cylinder or block of the CMS system minidisk; *count* is a decimal number that specifies the number of cylinders or blocks that make up the minidisk; and *volid* is the volume identifier of the real DASD volume on which the minidisk resides. (Your copy of the directory entry for the CMS system disk indicates the values you must specify for *start*, *count*, and *volid*.)

   To define the minidisk containing the directory source file, enter:

   ```
   define mdisk as 191 start count volid
   ```

   where *start* is the decimal number that specifies the cylinder or block on the real disk that corresponds to the first cylinder or block of the minidisk containing the directory source file; *count* is a decimal number that specifies the number of cylinders or blocks that make up this minidisk; and *volid* is the volume identifier of the real DASD volume on which the minidisk resides. (The directory entry for the minidisk containing the CMS file for updating the directory indicates the values you must specify for *start*, *count*, and *volid*.)

   To define the minidisk containing the user directory, enter the following command:

   ```
   define mdisk as vdev start count volid
   ```

   where *vdev* is the virtual device number that is assigned the DASD pack on which the directory resides; *start* is the decimal number that defines the cylinder or block on the DASD that is to be considered cylinder/block 0 for this minidisk. *count* is the total number of cylinders or blocks this pack contains; and *volid* is the volume identifier of this pack. (The values you specify for both *vdev* and *volid* must match the device number and volume identifier specified on the DIRECTORY statement in the directory.)

2. Bring up CMS in your virtual machine.

   After you have defined all three minidisks, enter the following command to bring up CMS in your virtual machine:

   ```
   ipl 190
   ```

3. Change, if necessary, the directory source file.

   If you think there is a problem with the CMS files containing the directory source statements, use the CMS XEDIT command to update this source file. For information about changing the directory source statement files, see the *z/VM: CP Planning and Administration*.

   If you think the directory was somehow destroyed and only needs to be recreated, go directly to the next step.

4. Use the DIRECTXA command.

   When you are satisfied that the directory source file is correct, enter:

   ```
   directxa filename
   ```

   where *filename* is the name of the CMS file containing the source statements for the directory. This file should have a file type of DIRECT. If it does not, specify the file type as well as the file name when you enter the DIRECTXA command.

   Note:

   a. By default, the DIRECTXA utility is located on the cross release utilities disk (PMAINT 551). This disk must be accessed to run the DIRECTXA utility.

   b. When writing a user directory, DIRECTXA does not overwrite the current directory but does write an alternate user directory. The directory pointer in cylinder 0 or block 1 is then changed to reference this new directory. It is advisable, therefore, to enter the DIRECTXA command twice. This assures that the directory you wish to use is available; otherwise, you may lose your directory again.

5. Restore normal system operation.

   Once you recover the directory, you should do one of the following:

   - Log off and log back on using your normal user ID
   - Enter the SHUTDOWN command to shut down the system and then follow normal procedures for bringing it back up.

   In either case, once you are logged on under your normal user ID, enter the following command to enable other users to log on:

   ```
   enable all
   ```

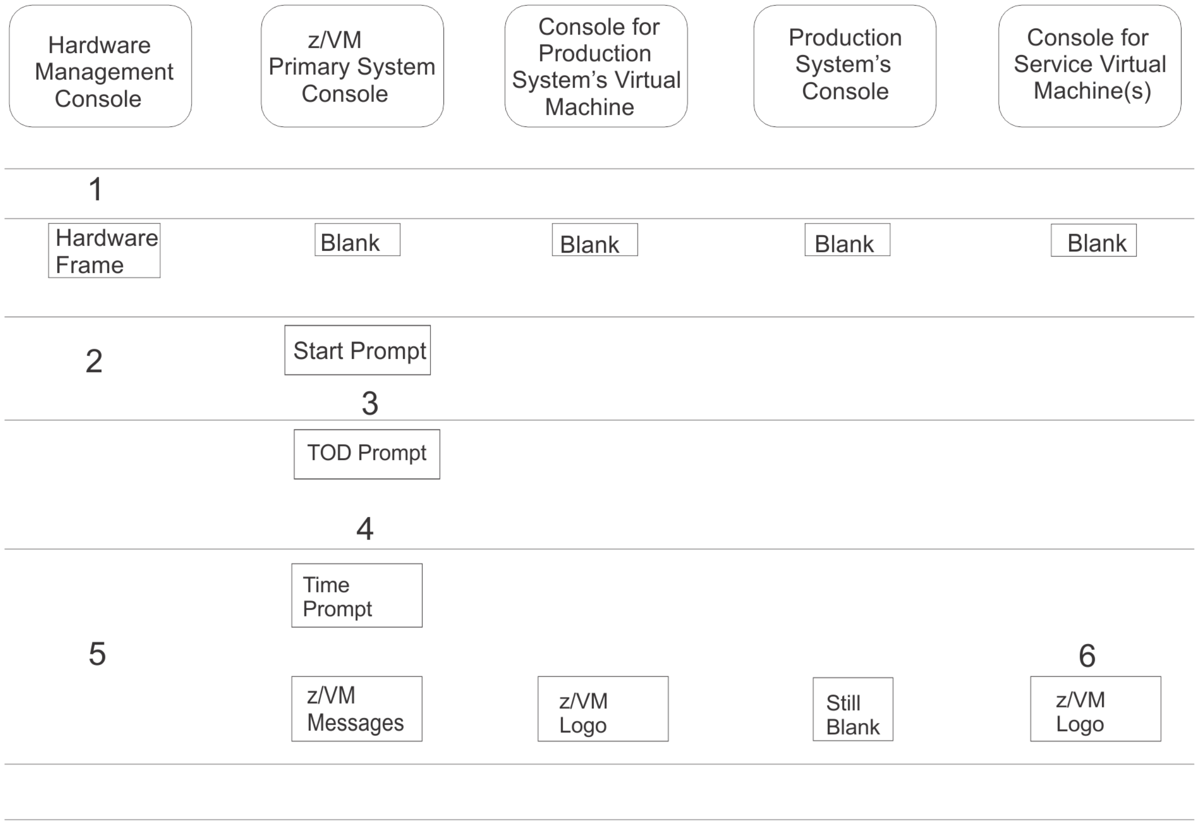
   You must enter this command because you disabled all devices in Step “3” on page 48.

Note:

1. You can use this process to recover the z/VM user directory at any time. However, if the directory source file is all right, simply enter the DIRECTXA command after you enter the IPL 190 command.
2. If, after you recover the directory, you determine there are still problems with the source file (or if there are problems that need to be fixed before you can recover the directory), you may have to update the directory source file. For information on how to do this, see the *z/VM: CP Planning and Administration*.

Summary of Bringing Up the System

Figure 4 on page 52 through Figure 7 on page 55 summarize the procedures for bringing up the z/VM system.


Hardware Management Console
z/VM Primary System Console
Console for Production System's Virtual Machine
Production System's Console
Console for Service Virtual Machine(s)
1
Hardware Frame
Blank
Blank
Blank
Blank
2
Start Prompt
3
TOD Prompt
4
5
Time Prompt
z/VM Messages
z/VM Logo
Still Blank
6
z/VM Logo

To bring up a z/VM system:

1 Check everything listed in “Before You Begin” at the beginning of this chapter.

2 From the processor complex's system console, load the z/VM console program (CP) into storage.

3 Unless you have a specific reason to do otherwise, specify a warm start. When CP prompts you, enter:

```
warm
```

If you encounter spooling errors, CP advises you of those errors. You must decide whether to: continue system initialization, enter:

```
go
```

stop system initialization, enter:

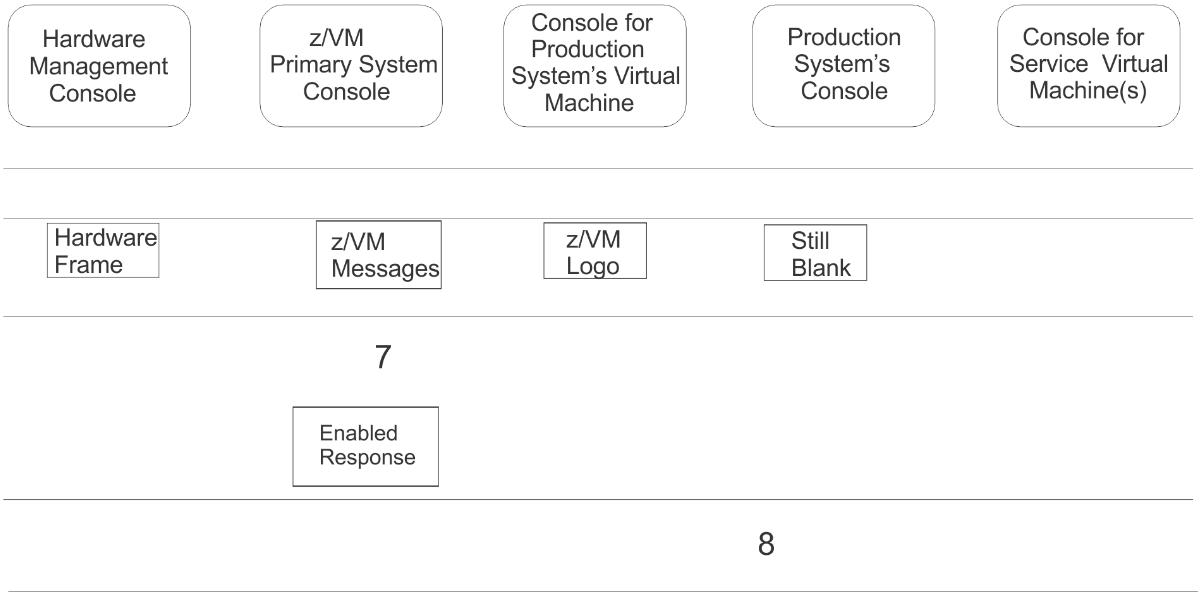
```
stop
```

4 (May not be prompted if AUTO_WARM_IPL is enabled).

5 If necessary, update the time.

6 If necessary, start up any z/VM service virtual machines that your installation runs.

Figure 4. Summary of Bringing Up the System (1 of 4)


Hardware Management Console
z/VM Primary System Console
Console for Production System's Virtual Machine
Production System's Console
Console for Service Virtual Machine(s)
Hardware Frame
z/VM Messages
z/VM Logo
Still Blank
7
Enabled Response
8

To bring up a z/VM system:

7 Make sure the console you plan to use to log on the production system's virtual machine is enabled by entering:

```
query rdev
```

where *rdev* is the real device number at the display you are using as the console for the production system's virtual machine.

If the status indicates that the display is enabled, go to step 8.
If the status indicates that the display is disabled, enter:

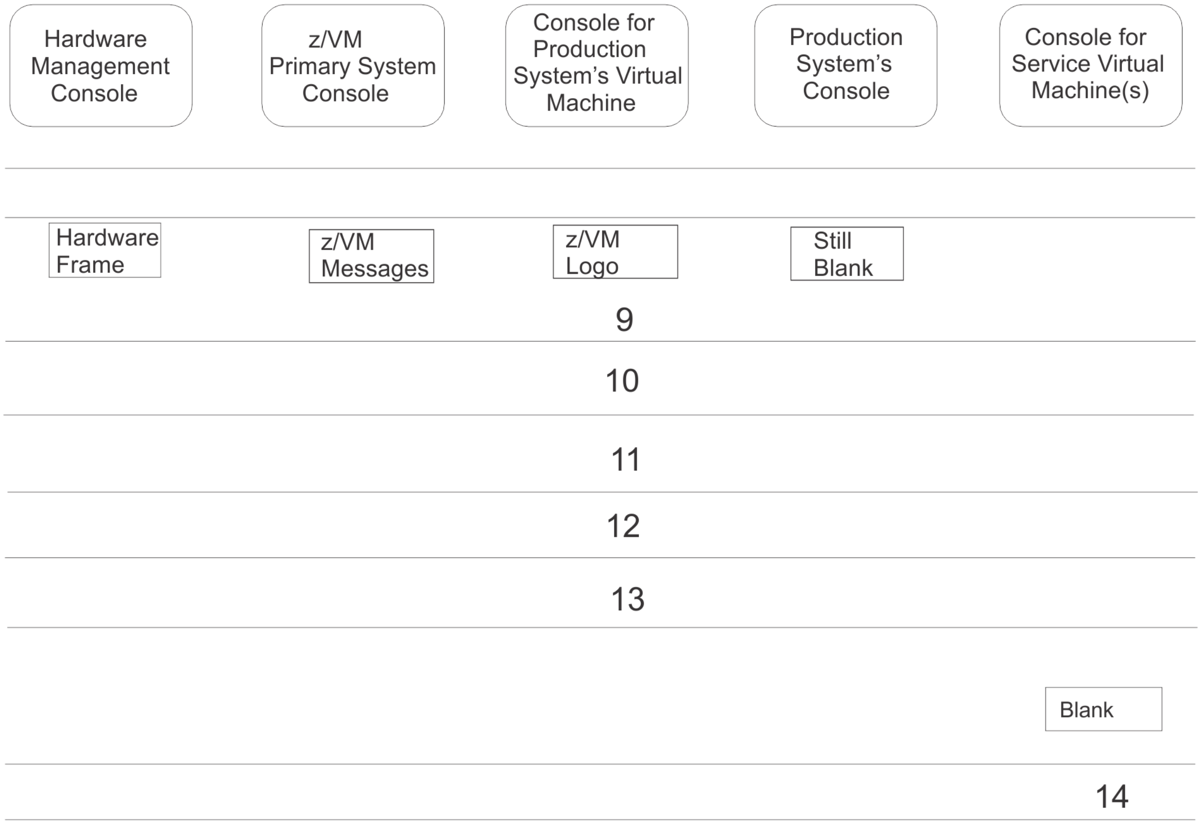
```
enable rdev
```

If the status indicates that the display is offline, enter the following commands:

```
vary on rdev
enable rdev
```

8 Go to the console you plan to use to log on the production system's virtual machine. If the device is enabled, you see the z/VM logo. If you do not see the logo, go back to step 7.

Figure 5. Summary of Bringing Up the System (2 of 4)


Hardware Management Console
z/VM Primary System Console
Console for Production System's Virtual Machine
Production System's Console
Console for Service Virtual Machine(s)
Hardware Frame
z/VM Messages
z/VM Logo
Still Blank
9
10
11
12
13
Blank
14

9 On the z/VM logo, log on the production system's virtual machine by entering the logon ID. When CP prompts you, enter the password.

10 Watch the messages CP sends you about the production system's devices. If necessary, go back to the z/VM system console and attach any missing devices to the production system's virtual machine.

11 If you moved to the z/VM console in the previous step, return to the console where the production system's virtual machine is logged on.

12 If necessary, enter any CP commands your installation requires to set up the running environment of the production system's virtual machine.

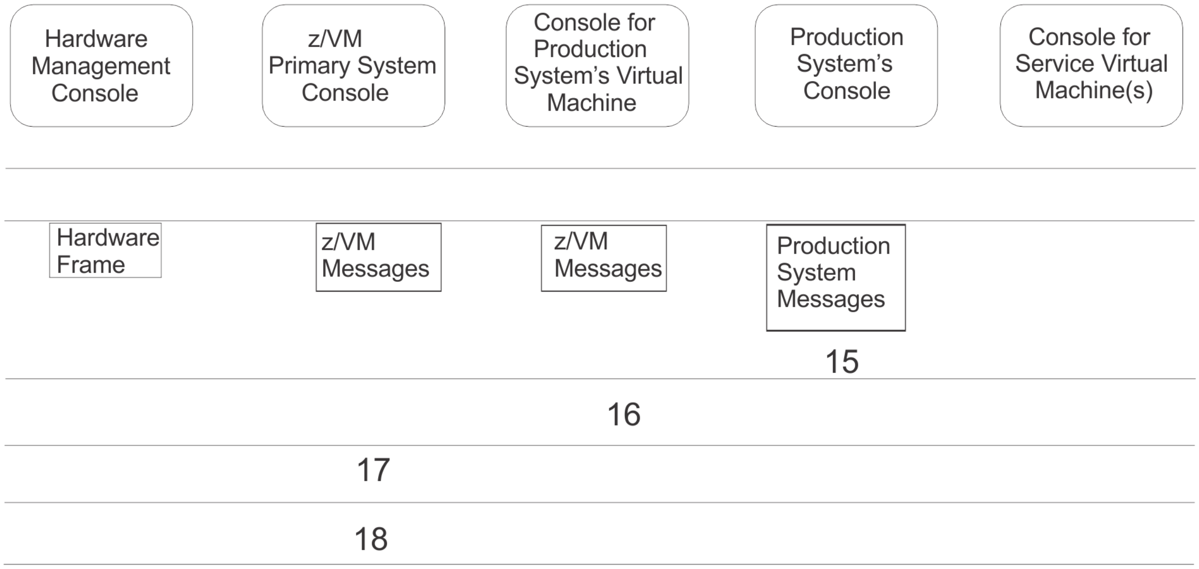
13 Load (IPL) the production system in the virtual machine. For example:

IPL *vdev*

where *vdev* is the virtual device number of the device on which the production system resides.

14 Go to the production system's system console and wait for the production system to come up on it.

Figure 6. Summary of Bringing Up the System (3 of 4)


Hardware Management Console
z/VM Primary System Console
Console for Production System's Virtual Machine
Production System's Console
Console for Service Virtual Machine(s)
Hardware Frame
z/VM Messages
z/VM Messages
Production System Messages
15
16
17
18

15 When the first message appears on the production system's system console, bring up the production system as you normally do.

16 If you want to use the console on which you IPLed the production system for something else, go to it and enter:

```
#cp set run on
#cp disconnect
```

17 Check to make sure the system has initialized properly.

This includes checking to see that devices are properly allocated, service virtual machines are running properly, and remote lines are up.

18 Prepare the system for a possible failure.

This includes making sure the dump environment is properly defined, CP tracing is active, taking a stand-alone dump, if necessary, and creating a console log and knowing how to recover your production system.

Figure 7. Summary of Bringing Up the System (4 of 4)

Chapter 5. Running the System

This chapter tells you how to run the system under normal conditions. It describes all the tasks required to run the system except controlling local devices. Information about controlling local devices is located in Chapter 6, “Controlling Local Devices,” on page 81.

If you need only a brief summary of the tasks in this chapter, see “Summary of Running the System” on page 77.

After reading this chapter, you can:

- Locate information about checking system status
- Check the commands your ID is authorized to use
- Check the status of system users
- Watch the system for changes and requests
- Locate information about controlling local and remote devices
- Control processors and storage
- Query the cryptographic hardware
- Locate information about how to control the system's service virtual machines
- Communicate with users of the system.

Checking System Status

As a z/VM operator, you are responsible for making sure that the system's resources are available to users. One of the first things you need to know, then, is what the status of the system is.

If you just brought up the system, you should have a good idea. “Checking the Initialization of System Resources” on page 40 discusses checking the status of the system when you first bring it up.

If the system is already running when you come on shift, you need to find out about particular aspects of the system. These include its:

- CP accessed minidisks
- Performance
- Processors
- Crypto Express adapters
- Storage
- Devices
- Service virtual machines
- Users.

You may also want to know the level of z/VM you are running.

Most of these items are discussed either later in this chapter or elsewhere in this book. The items discussed elsewhere are:

- System performance (see “Collecting Performance Information” on page 180)
- The status of the system's local devices (see “Checking the Status of Real Devices” on page 41).

The items discussed in this chapter are:

- The status of the system's real processors (see “Controlling Processors” on page 64)
- The status of the system's Crypto Express adapters (see “Finding out the Status of the Crypto Resources” on page 66)

- The status of the system's real storage (see “Controlling Storage” on page 66)

The sections that follow tell you how to check the:

- Release level of CP you are running
- Commands and DIAGNOSE codes your ID can use
- Status of the service virtual machines
- Status of the single system image (SSI) cluster
- Status of the system users
- Virtual device number of any virtual device.

Checking the Release Level of CP

To find out information about the level of CP you are running, enter:

```
query cplevel
```

In response, CP displays the software release and service level numbers, the date and time when CP was generated, and the date and time when CP was last IPLed.

Checking Where CP Was Loaded from and What Parm Disk Was Used

To find the location of the CP module and the parm disk that CP used at IPL time, enter:

```
query cpload
```

In response, CP tells you the name and location of the CP module and the location of the parm disk used at IPL time. For more information about the QUERY CPLOAD command, see the *z/VM: CP Commands and Utilities Reference*.

Finding Out What Commands and DIAGNOSE Codes Your ID Can Use

To find out what commands and DIAGNOSE codes your ID is authorized to use, enter:

```
commands
```

In response, CP displays a list of commands and DIAGNOSE codes that your ID has been authorized to use.

Checking the Status of the System's Service Virtual Machines

To check the status of the accounting and error recording (EREP) virtual machines, use the CP QUERY RECORDING command. For more details, see “Finding Out the Status of Accounting” on page 185 or “Finding Out the Status of Error Recording” on page 193.

You can exempt service virtual machines from free storage limit detection by using the SET STGEXEMPT command. For more information about the SET STGEXEMPT command, see the *z/VM: CP Commands and Utilities Reference*.

To check the status of the service virtual machines used by z/VM facilities and optional features, such as TCP/IP, DirMaint, and RSCS, use the commands appropriate to these facilities and features.

To check the status of the licensed program service virtual machines, use the commands appropriate to those products. Supported licensed programs that require service virtual machines include RSCS and VM/Pass-Through.

Checking the Status of the Single System Image (SSI) Cluster

If your system is a member of an SSI cluster, and you want to find out the state and heartbeat information for each system in the cluster, enter:

```
query ssi status
```

If you want to find out the status of the connections from this system to the other systems in the SSI cluster, enter:

```
query ssi connect
```

Checking the Status of System Users

Checking the status of system users includes determining:

- How many users are logged on
- The maximum number of users who can log on
- Who is logged on
- The system resources CP has allocated to a user
- A user's scheduling share.

Finding Out the Number of Users

To find out how many users are logged on to the z/VM system, enter:

```
query users
```

The response from CP indicates how many users are logged on and how many have entered the DIAL command. (A user enters the DIAL command to gain access to an operating system another user is running in a virtual machine.)

In an SSI cluster, total counts for the cluster are displayed.

To find out how many users are logged on to the local member of the SSI cluster, enter:

```
query users at *
```

To find out how many users are logged on to a specific member of the SSI cluster, enter:

```
query users at sysname
```

To find out how many users are logged on to each member of the SSI cluster, enter:

```
query users at all
```

A response will be issued from each member.

Finding Out the Maximum Number of Users Who Can Log On

To find out the maximum number of users who can log on to the z/VM system, enter:

```
query maxusers
```

The response from CP indicates the maximum number of users allowed to log on to the system at any one time.

Setting the Maximum Number of Users Who Can Log On

To set the maximum number of users who can log on to the z/VM system, enter:

```
set maxusers nnnnn
```

where *nnnnn* is the maximum number of users allowed.

Listing the User IDs of Logged-on Users

To list the user IDs of all users who are logged on to the z/VM system, enter:

```
query names
```

The response from CP displays the user ID of each logged-on user and the real or logical device number where the user is connected. If the user is disconnected, the response indicates DSC.

In an SSI cluster, all of the logged-on users are displayed. If a user is logged on to a remote member of the cluster, "SSI" is displayed instead of a device number.

To display only the users on the local member of the SSI cluster, enter:

```
query names at *
```

To display the users on a specific member of the SSI cluster, enter:

```
query names at sysname
```

To display the users on each member of the SSI cluster, enter:

```
query names at all
```

A response will be issued from each member.

Note: Displaying all of the users in the SSI cluster could tie up your console with many thousands of names.

Finding Out If a Specific User Is Logged On

To find out whether a specific user is logged on, enter:

```
query userid
```

or

```
query users userid
```

If the user is logged on, CP responds with the user ID you specified and the real or logical device number where the user is logged on. If the user is disconnected, the response indicates "DSC".

In an SSI cluster, if the specified user is logged on to a remote member of the cluster, "SSI" is displayed instead of the device number.

To obtain additional logon information about a user, such as the member of the SSI cluster where the user is logged on, enter:

```
query userid extended
```

or

```
query users userid extended
```

To obtain information about all of the logon instances for a multiconfiguration virtual machine, enter:

```
query userid at all
```

or

```
query users userid at all
```

A response will be issued from each member of the SSI cluster where the user is logged on.

Checking a User's System Resource Usage

To find out what system resources CP has allocated to a user, enter:

```
indicate user userid
```

The response from CP for this command indicates:

- The user's ID (USERID).
- The user's virtual machine mode (MACH), which indicates the virtual machine architecture:

  ESA
  The virtual machine is processing according to ESA/390 architecture (31-bit). ESA virtual machines are also capable of processing according to z/Architecture®® (64-bit) if switched into that mode by a guest operating system.

  XA
  This designation is supported for compatibility. XA virtual machines are functionally equivalent to ESA virtual machines. Some applications may check the virtual machine designation and may require the application to be running in an XA virtual machine.

  XC
  The virtual machine is processing according to ESA/XC architecture.

  Z
  The virtual machine is processing according to z/Architecture (64-bit).
- The user's virtual machine storage size (STOR).
- The virtual device number or name of the named saved system specified on the last IPL command the user entered (IPLSYS).
- The number of devices attached to the user (DEVNUM).
- The number of virtual storage pages resident in real storage (RES), in the working set (WS), locked (LOCKEDREAL), or reserved (RESVD).
- The number of pages, owned by the user, residing on DASD.
- The number of page reads (READS) and page writes (WRITES) for this user since logon.
- The number of the virtual processor.
- The user's connect time (CTIME), virtual time (VTIME), and virtual and simulation time (TTIME).
- The number of I/O requests the user has made (IO).
- The number of virtual cards the user has read (RDR) and punched (PCH), and the number of virtual lines the user has printed (PRT) since logon or since entering the last ACNT command.

Finding Out If a User ID is Subject to Free Storage Limit Detection

To find out if a particular user ID is subject to free storage limit detection, enter:

```
query stgexempt userid
```

where *userid* is the user ID of the virtual machine you are checking on.

The response will indicate that either the virtual machine is subject to or exempt from free storage limit detection. You can change this condition by using the SET STGEXEMPT command. There may be an additional response saying that free storage limit detection is off for the system. The SET STGLIMIT command can be used to turn free storage limit detection on for the system.

To list the user IDs of all users currently exempt, enter:

```
query stgexempt all
```

For more information about the QUERY STGEXEMPT ALL command, see *z/VM: CP Commands and Utilities Reference*.

Finding Out a User's Scheduling Share

You may need to find out how much of the system's processing power is assigned to a user (the user's scheduling share). To do this, enter:

```
query share userid
```

The response from CP indicates the following information:

- Share values for all processor types (CP, zIIP, IFL, and ICF)
- Kind of share assigned (ABSOLUTE or RELATIVE)
- Amount of the share (an absolute percentage or relative number)
- Number of processors dedicated to the user, if any

The kind of share assigned might be one of the following:

- ABSOLUTE, which indicates the user's share is a percentage of all the processing power that remains after taking out any dedicated processor and the resources required for CP overhead
- RELATIVE, which indicates that CP assigns a processing share to the user that takes into account the shares of other relative users. CP assigns the relative shares after absolute shares, and absolute shares after dedicated processors and CP overhead
- *m* DEDICATED CPU[S], which specifies the number of virtual CPUs of type CP or IFL that this user has that have a real processor dedicated to them. This string does not appear if the user has no dedicated processors. Because only processors of the primary real processor type can be dedicated, this string can appear only on the response line for virtual CPUs of type CP or IFL.

A user's maximum share of CPU resources might be limited by the user's assignment to a resource pool, which limits the amount of CPU resources (in terms of physical CP or IFL processors) that a group of users are allowed to consume in aggregate. If you know the name of the resource pool, you can get information about the type and setting of the CPU limit (and verify the user's assignment) by issuing the following command:

```
query respool pool poolname members
```

If you do not know the name of the resource pool, or you are uncertain of the user's assignment to any resource pool, you can get the information by issuing the following command:

```
query respool user userid
```

If a user has individual CPU limits as defined by the SET SHARE command, and the user is assigned to a resource pool, both limits will be checked each time the user's CPU usage is evaluated, and the more restrictive limit will be applied.

For more information about the QUERY SHARE and QUERY RESPOOL commands, see *z/VM: CP Commands and Utilities Reference*.

Changing the Virtual Device Number of Any Virtual Device

You can change the number of any virtual device in the configuration by using the REDEFINE command. When you use REDEFINE to alter the configuration of the virtual machine, the changes made are temporary and are in effect only for the current session. Responses are sent to the display console to confirm the configuration change request has taken place. If you have entered SET IMSG OFF, these responses do not appear on your console.

Note: Virtual paging subsystem devices should not be redefined. The integrity of the user's virtual paging subsystem may be lost if it is redefined.

To redefine the virtual device number, enter:

```
redefine vdev1 as vdev2
```

where

vdev1
is the virtual device number of an existing virtual device

vdev2
is the new virtual device number.

For example, if 01CF is the current device number of a 3278, the command:

```
redefine 01CF as 01EF
```

redefines device number 01CF as 01EF.

The following response confirms the virtual device number has been changed:

```
type vdev DEFINED
```

Watching the System for Changes and Requests

You will spend much of your time watching the system for changes in system availability and for requests from users for resources. In general, you need to watch the:

- z/VM system console
- Production system's system console.

Watching the z/VM System Console

On the z/VM system console, you receive messages from the operating system and from z/VM users.

The operating system sends you messages about hardware errors, software errors, and resource shortages. It also sends you informational messages about such matters as when users log on and log off.

z/VM users who need to have tapes or DASD volumes mounted may send messages to make their requests.

Watching the z/VM system console lets you respond quickly to system errors and z/VM user requests.

Watching the Production System's System Console

You will probably spend most of your time watching the production system's system console. But this depends on the operating system you are running and its workload. It also depends on whether your production system is part of a larger network of operating systems.

Watching the production system's system console allows you to keep tabs on the part of the system that is doing the most work.

Controlling Local Devices

A large part of your job as a z/VM system operator is controlling local devices. The next chapter, Chapter 6, "Controlling Local Devices," on page 81, contains information about controlling the following types of local devices:

- Dynamic switching devices
- 3270 devices and communication lines
- DASDs
- Tape drives
- CTCAs
- OSAs
- Printers

It also tells you how to control:

- Channel paths to local devices
- SNA/CCS terminals.

Controlling Remote Devices

If your installation has installed the VM/Pass-Through Facility, users at remote locations can use 3270 displays to log on to z/VM. They may also use 3270 printers to print hard copy at their remote sites.

If your installation has installed VM/VTAM or VTAM Communications Network Application (VCNA), users of Systems Network Architecture/Console Communication Services (SNA/CCS) terminals can log on to z/VM. To control these devices, you can use the following types of commands:

- CP
- VCNA
- VSCS
- VTAM.

Because you can use SNA/CCS terminals as either local or remote devices, more information about them is in Chapter 6, “Controlling Local Devices,” on page 81.

For more information on using the commands, see the appropriate library.

If your production system's configuration includes remote devices, use appropriate production system commands to control them.

Controlling Processors

Sometimes you may need to use CP commands to control the real processors available to z/VM. For example, you may need to:

- Find out which processors are active
- Find out how CP is using processors
- Change the availability of processors to the system
- Change the scheduling of processors for users.

Finding Out Which Processors Are Active

To find out which of your real processors are active, enter:

```
indicate load
```

In an SSI cluster, if you want the response from another member of the cluster, enter:

```
indicate load at sysname
```

The response from CP includes the percentage of time each processor is running. The processors listed in the response are the active processors.

Finding Out How CP Is Using Processors

CP uses a real processor in one of the following ways:

- As the *master* processor. The master processor is the processor on which certain CP system work is required to run. It is usually the processor on which you last IPLed the system.
- As an *alternate* processor. In a multiprocessor system, an alternate processor is a processor that is neither the master processor nor a dedicated processor. A uniprocessor system has only one processor, the master processor.

- As a *dedicated* processor. A dedicated processor is a processor that CP reserves for the sole use of a virtual processor belonging to a virtual machine.

To find out how CP is using each real processor in your configuration, enter:

```
query processors
```

In response, CP displays each online processor and its use as a master, alternate, or dedicated processor. If a processor is dedicated, CP also displays the user ID of the virtual machine and the address of the virtual processor to which the real processor is dedicated.

Changing the Availability of Processors to the System

Under normal circumstances, you bring up and run the system with all the processors in the processor complex or partition. But if support personnel at your installation instruct you to do so, you may use the VARY ONLINE PROCESSOR or VARY OFFLINE PROCESSOR commands to control whether a processor is available to z/VM.

For more information on these commands, see the *z/VM: CP Commands and Utilities Reference*.

Changing the Scheduling of Processors for Users

By scheduling and dispatching their work, CP makes sure that all virtual machines get to use the real processor resources. Your installation can influence the scheduling of a user's virtual machine in several ways:

- By changing the share of the system's processing power assigned to the user
- By changing the temporary boost CP gives users who are working at displays (the system's interactive bias)
- By changing how much time CP gives each user to use real processor resources before CP reassigns the processor to another user (the dispatching time slice)
- By dedicating a processor to a virtual processor belonging to a user (or removing it).

Usually, your installation sets up the scheduling shares for each virtual machine during system generation. CP supplies default values for the system's interactive bias and for the dispatching time slice, which your installation may use. Thus, when you bring up the z/VM system, you usually do not have to adjust any of the scheduling controls.

However, if support personnel at your installation instruct you to do so, you may use one of the following commands:

- SET SHARE—to change a user's scheduling share
- SET SRM IABIAS—to change the system's interactive bias
- SET SRM DSPSLICE—to change the size of the dispatching time slice CP assigns each user
- SET SRM STORBUF—to view host storage as being partitioned when moving users from the eligible to dispatch lists
- SET SRM LDUBUF—to partition the system's paging resources when moving users from the eligible to dispatch lists
- SET SRM MAXWSS—to set maximum working size for users
- DEDICATE—to dedicate a processor to a virtual processor belonging to a virtual machine
- UNDEDICATE—to remove sole use of a processor from a virtual CPU belonging to a virtual machine
- DEFINE RESPOOL—to create a resource pool that limits the amount of CPU resources that a group of user's are allowed to consume in aggregate
- SCHEDULE—to assign a user to a resource pool or remove a user from a resource pool
- SET RESPOOL—to change the CPU limit setting for a resource pool

For more information on these commands, see the *z/VM: CP Commands and Utilities Reference* or the *z/VM: CP Planning and Administration*.

Finding out the Status of the Crypto Resources

To determine if Crypto Express adapters are installed on this system enter:

```
query crypto
```

The response tells you whether or not the instructions performed by Crypto Express adapters are available.

To find out the status of crypto resources on Crypto Express adapters enter:

```
query crypto domains
```

The response tells you the device status, configuration state, and device usage of each crypto resource available on the system.

For more information on these commands, see *z/VM: CP Commands and Utilities Reference*.

For more information on planning and managing crypto resources on a z/VM system, see Crypto Planning and Management in *z/VM: CP Planning and Administration*.

Controlling Storage

Sometimes you may need to query or control your processor complex's storage. You can use CP commands to:

- Check and increase the amount of real storage used by z/VM
- Check the allocation of real storage
- Check the status of system execution space pages
- Reserve pages for a particular virtual machine in host storage
- Lock or release virtual machine pages in host storage
- Examine or change values in host storage

Free Storage Limit Detection

The free storage limit detection function keeps track of free storage so that a guest will not be able to bring the system down.

Use the QUERY STGLIMIT command to check the state of free storage limit detection for the system. Use the SET STGLIMIT command to control the state of free storage limit detection for the system.

Use the QUERY STGEXEMPT command to check the state of free storage limit detection for virtual machines. Use the SET STGEXEMPT command to control the state of free storage limit detection for virtual machines.

Exempting a user ID will ensure that the user ID is not subject to being stopped or forced due to the amount of free storage it causes CP to use. Exempting a user ID is recommended for special purpose user IDs that are vital to the installation, for example:

- User IDs running trusted code
- User IDs that should never be forced off the system.

If no action is taken, free storage limits will be enforced for all users.

Changing the Size of Real Storage

Use the QUERY STORAGE command to check the current real storage values:

```
query storage
```

The response to this command indicates the following storage sizes:

Field
Meaning

STORAGE
Online real storage (the amount of storage used by z/VM)

CONFIGURED
Configured real storage (the amount of storage configured for the logical partition in which z/VM is running)

INC
Real storage increment

STANDBY
Real storage in standby state

RESERVED
Real storage in reserved state

If standby storage is available, you can use the SET STORAGE command to dynamically increase the amount of online real storage. If the current online storage is less than the configured storage, the remaining configured storage also is available. To specify the new online storage size, enter:

```
set storage size
```

Or, to increase the online real storage by a specific amount, enter:

```
set storage +additional
```

Note:

1. See the SET STORAGE command for information about the required format for the *size* or *additional* value.
2. Whether you specify a storage size or an additional amount of storage, the new storage size will be set (rounded up if necessary) to a multiple of the real storage increment.

Attention

Good change management procedures suggest that you should avoid simultaneous storage configuration operations. Before issuing the SET STORAGE command:

1. Ensure that all other storage reconfigurations have completed. This includes logical partition activation or deactivation, storage configuration actions by any other operating system, and any other SET STORAGE command.
2. Issue the QUERY STORAGE command as necessary to ensure that the reported STANDBY and RESERVED values are stable.
3. Ensure that the desired increase in storage size is available:

   ```
   increase <= STANDBY + (CONFIGURED - STORAGE)
   ```

4. Only then issue the SET STORAGE command.

If you need to reset the online real storage to a lower value, you can issue the SHUTDOWN command with the REIPL operand and use the IPLPARMS option to specify STORE=*nnnnu* as an IPL parameter.

For more information about the QUERY STORAGE, SET STORAGE, and SHUTDOWN REIPL commands, see *z/VM: CP Commands and Utilities Reference*.

Checking the Allocation of Real Storage

Use the QUERY FRAMES command to get information about how your system's real storage is allocated:

```
query frames
```

The response to this command indicates the size and allocation of host real storage, in 4 KB frames:

- Initialized at IPL (Configured)
- Installed (Real)
- Usable (Usable)
- Offline (Offline)
- Available for paging (Pageable)
- Not initialized above 2 GB (NotInitialized)
- Cleared to 0 and available for use by CP, the total for all processors (GlobalClearedAvail)
- Cleared to 0 and available for use by CP, the sum of the lists for each processor (LocalClearedAvail)
- Not cleared to 0 and available for use by CP, the sum of the lists for each processor (LocalUnclearedAvail)

The second part of the response indicates the real storage usage **below 2 GB**, in frames:

- Not cleared to 0 and available for use by CP, the total for all processors (GlobalUnclearedAvail)
- Available for paging (Pageable)
- Allocated for host logical free storage (LogicalFreeStorage)
- Allocated for host real free storage (RealFreeStorage)
- Locked by CP (LockedRS)
- Locked as the result of CP LOCK and UNLOCK commands (LockedCmd)
- In use for minidisk track cache (MinidiskCache)
- In use to back the CP nucleus pages and prefix pages (Nucleus/Prefix)
- In use for trace tables, total for all processors (Trace)
- In use for any other purpose not included above (Other)

The third part of the response indicates the real storage usage **above 2 GB**, in frames:

- Not cleared to 0 and available for use by CP, the total for all processors (GlobalUnclearedAvail)
- Available for paging (Pageable)
- Allocated for host logical free storage (LogicalFreeStorage)
- Allocated for host real free storage (RealFreeStorage)
- Locked by CP (LockedRS)
- Locked as the result of CP LOCK and UNLOCK commands (LockedCmd)
- In use for minidisk track cache (MinidiskCache)
- In use for any other purpose not included above (Other)

Checking the Status of System Execution Space Pages

CP control blocks, the CP nucleus, and files loaded with the CPXLOAD command or configuration statement are all mapped in the system execution space and are addressed by host logical storage addresses. You can use the QUERY SXSSTORAGE command to display the host logical storage address ranges that are in use or usable by the system for the system execution space. The frame table and the system execution space page management table reside above 2 GB; the rest of the system execution space resides below 2 GB.

To display the status of system execution space pages below 2 GB, enter:

```
query sxspages
```

The response to this command lists the following system execution space status:

- All pages (Total, Backed below 2 GB, Backed above 2 GB, Unbacked)
- Available pages (Total, Backed below 2 GB, Backed above 2 GB, Unbacked)
- In use pages (Total, Alias, Nucleus/Prefix, FreeStorage, Other)
- Free storage pages (Backed below 2 GB, Backed above 2 GB, Reserved)
- Locked pages (Total, By command)
- Deferred page requests waiting for:
  - Available page (Alias, Backed below 2 GB, Backed by any)
  - Available frame (Frame below 2 GB, Frame any)
  - Total

Reserving Pages

If support personnel instruct you to do so, you can use the SET RESERVED command to solve certain performance problems.

A virtual machine, especially one running a large operating system, needs a certain minimum number of pages to be resident in host storage to run efficiently. Your installation can use the SET RESERVED command to ensure that CP reserves that minimum number for the virtual machine.

If you need to find information about reserved pages, you can use the QUERY RESERVED command.

For more information about the SET RESERVED command and the QUERY RESERVED command, see the *z/VM: CP Commands and Utilities Reference*.

Locking and Releasing Pages in Storage

If support personnel instruct you to do so, you can use the LOCK command to solve certain performance problems.

If CP is paging certain pages frequently, your installation can lock into host storage selected pages that belong to a virtual machine. Locking pages in storage prevents CP from paging them out.

ATTENTION: Locking too many pages in storage can degrade system performance for other users or cause system termination. Accordingly, you should use this command only when support personnel instruct you to do so.

To release locked pages, use the UNLOCK command.

For more information about the LOCK and UNLOCK commands, see the *z/VM: CP Commands and Utilities Reference*.

Examining and Changing Values in Host Storage

If support personnel instruct you to do so, you can use the LOCATE (Storage), DISPLAY (Host Storage), DUMP (Host Storage), and STORE (Host Storage) commands to locate, examine, dump, and change values in host storage, respectively.

⚠ **Attention:** Exercise extreme caution in entering the STORE (Host Storage) command. Careless storage alterations can cause abnormal terminations of your system, as well as security and integrity problems. Accordingly, you should use this command only when support personnel instruct you to do so.

Note: You may not be authorized to issue the STORE (Host Storage) command if an external security manager (ESM) is installed on your system. For additional information, contact your security administrator.

For more information about these commands, see the *z/VM: CP Commands and Utilities Reference*.

Controlling the System's Service Virtual Machines

Your installation may run one or more of the following service virtual machines:

- Accounting virtual machine
- Error recording (EREP) virtual machine
- Symptom record recording virtual machine
- Virtual machines required by z/VM facilities and optional features
- Virtual machines required by supported licensed programs
- Batch virtual machine.

For information about operating the accounting, error recording, and symptom record recording virtual machines, see “Collecting Accounting Information” on page 183, “Collecting Hardware Error Information” on page 190, or “Collecting Symptom Record Information” on page 202.

For information about operating the CMS batch virtual machine or supported licensed programs in their virtual machines, see the appropriate publication listed in the bibliography.

Communicating with Users

A final task essential to operating a z/VM system is communicating with users. As the primary system operator, to communicate with users you must be able to:

- Answer questions users ask.

  You must be prepared to answer user questions about system availability and whether particular programs or devices are up. Users may also ask you about things that represent their work, such as jobs, spool files, and user IDs.

  You need to enter appropriate CP commands or production system commands to answer these questions.
- Inform users of scheduled changes to system availability.

  To plan their use of the system, users need to know what the installation's planned changes are. Examples include what the weekend schedules are, when migration to new devices will take place, and when new software will be installed.

  You can communicate your installation's planned changes by updating the z/VM system log message.
- Inform users of unscheduled changes to system availability.

  When you must shut down the system without notice, as when a hardware error occurs, you should inform both your z/VM and production system users.

  You can communicate such unscheduled changes to z/VM users by using the CP WARNING command. On the production system's system console, use appropriate production system commands to communicate with your production users.
- Remind users of an impending shutdown.

  Again, you can use the CP WARNING command to tell z/VM users that you are about to shut down the system. On the production system's system console, use appropriate production system commands to communicate with your production users.
- Send informational messages to users.

  To send messages without the urgency of a warning, you can use the CP MESSAGE command to communicate with z/VM users. On the production system's system console, use appropriate production system commands to communicate with your production users.

To communicate with z/VM users, you must use z/VM commands to:

- Create and change the system log message
- Send messages or warnings to users.

To communicate with production system users, you must use production system facilities similar to the log message and message commands.

Creating and Changing the System Log Message

All z/VM users should know certain information about the system and its availability. Putting this information in the system log message brings it to the attention of all users as soon as they log on or reconnect.

If you specified FEATURES ENABLE LOGMSG_FROM_FILE in your system configuration file, CP searchs for log message information on CP-accessed minidisks. Otherwise, you have to specify the contents of the log message using the SET LOGMSG command.

Creating and Changing Log Message Files on CP-Accessed Minidisks

When a user logs on to the system, CP searches for specific files on the minidisks that are accessed at that time. You can specify which minidisks CP should access by placing CP_ACCESS statements in the system configuration file or executing CPACCESS commands.

Depending on the features enabled in the system configuration file and the capabilities of the terminal used by the user that is logging on, CP searches for the following files on the lowest CP-accessed minidisk.

Table 2. Log Messages

Types of Log Message	SBCS File	DBCS File
system log message	SYSTEM LOGMSG	SYSTEM LOGDBCS
account log message	account LOGMSACC	account LOGDBACC
ACI group log message	acigroup LOGMSACI	acigroup LOGDBACI
user log message	userid LOGMSUSR	userid LOGDBUSR

Note: Account ID, ACI group, and user ID based log message files are only displayed if the corresponding support is enabled on the FEATURES statement of the system configuration file.

If the user's terminal is DBCS capable, CP attempts to find the relevant DBCS files mentioned in the table above. Should a DBCS file be missing, CP instead uses the contents of the corresponding SBCS file. If the user's terminal is not DBCS capable, CP attempts to find the SBCS file mentioned in the table above. In either case, if the SBCS file is not present on any CP-accessed minidisk, the corresponding part of the log message is omitted. If CP cannot find any of the files mentioned in the table, the user receives the following response:

```
There is no logmsg data
```

For example, let us assume that your system configuration file contains the following statement:

```
   Features  Enable  LOGMSG_From_File  ,
                     Show_Account  Yes ,
                     Show_ACIgroup No  ,
                     Show_Userid   Yes
```

Let us also assume that CP responds as follows to a QUERY CPDISKS command:

```
Label Userid   Vdev Mode Stat Vol-ID Rdev Type StartLoc EndLoc
CF1   MAINT    0CF1  A   R/O  M01RES 0A0F CKD      2400   2449
CF2   MAINT620 0CF2  B   R/O  620RL1 0A0F CKD      2450   2499
```

If a virtual machine called TESTER, with an account ID of 3GWTEST defined in its user directory entry, is logging on through a non-DBCS terminal, CP searches for the following files on its accessed minidisks:

- SYSTEM LOGMSG
- 3GWTEST LOGMSACC
- TESTER LOGMSUSR.

If all three files exists, their contents will be displayed on the user's terminal. Assume that there exists a file called SYSTEM LOGMSG on MAINT's CF1 disk containing the following records:

```
***************************************************************
*                                                             *
*    The system will be shutting down early this evening      *
*                                                             *
+    Contact Bob at extension 7665 if you need the system     *
+    later tonight.                                           *
+                                                             *
***************************************************************
```

Assume that there exists a file called 3GWTEST LOGMSACC on MAINT's CF1 disk containing the following records:

```
*                                                             *
*    Remember that all testing must be completed by June 1st  *
*                                                             *
***************************************************************
```

Finally, assume that there exists a file called TESTER LOGMSUSR on MAINT620's CF2 disk containing the following records:

```
*                                                             *
*    Access MAINT's 390 disk if you want to use the new       *
*    level of CMS.                                            *
*                                                             *
***************************************************************
```

When user TESTER first logs on to the system, CP displays all records in the files that begin with an asterisk (*):

```
LOGMSG - 15:58:39 EDT Thursday 06/11/92
***************************************************************
*                                                             *
*    The system will be shutting down early this evening      *
*                                                             *
***************************************************************
*                                                             *
*    Remember that all testing must be completed by June 1st  *
*                                                             *
***************************************************************
*                                                             *
*    Access MAINT's 390 disk if you want to use the new       *
*    level of CMS.                                            *
*                                                             *
***************************************************************
```

The LOGMSG header record reflects the date and time the first file displayed in the log message was modified.

Should user TESTER issue a QUERY LOGMSG command, CP displays all records in the log messages files regardless of whether they begin with an asterisk or not:

```
LOGMSG - 15:58:39 EDT Thursday 06/11/92
***************************************************************
*                                                             *
*    The system will be shutting down early this evening      *
*                                                             *
+    Contact Bob at extension 7665 if you need the system     *
+    later tonight.                                           *
+                                                             *
***************************************************************
*                                                             *
*    Remember that all testing must be completed by June 1st  *
*                                                             *
***************************************************************
*                                                             *
*    Access MAINT's 390 disk if you want to use the new        *
*    level of CMS.                                             *
*                                                              *
****************************************************************
```

You can use the CPLISTFILE and CPTYPE commands to display the contents of the various parts of the log message or you can issue commands like:

- QUERY LOGMSG SYSTEM
- QUERY LOGMSG ACCOUNT 3GWTEST
- QUERY LOGMSG USER TESTER

to display the various parts of the log message on your terminal.

Changing the Log Message

To change the contents of any of the log message files, follow these steps:

- Issue a CPRELEASE command instructing CP to release its access to the minidisk on which the file resides.
- Use the CP LINK and CMS ACCESS commands to gain access to the minidisk.
- Edit the file you wish to change.
- The CMS RELEASE and CP DETACH commands to release the minidisk and DETACH it from your virtual machine.
- Enter a CPACCESS command to re-establish CP's access to the minidisk.

Once you instruct CP to release its access to the minidisk, CP cannot read any files on the minidisk until it has been accessed again. We suggest, therefore, that you keep the log message files on two disks. When you wish to update the log message files, you can simply issue CPRELEASE for the first disk (CP will still have access to the second disk), make your changes, and enter CPACCESS to put the changes into effect. Next, enter CPRELEASE for the second disk, copy the changed log message files to the first disk, and issue CPACCESS to have CP access the second disk. Putting this information in the system log message brings it to the attention of all users as soon as they log on or reconnect.

To use the system log message to communicate with z/VM users, you must be able to:

- Create a new system log message
- Change an existing system log message.

Creating a New System Log Message

To create a new system log message, enter:

```
set logmsg nn * text
```

where *nn* is the line number of the message in which this text is to appear. For example, if *nn* is 2, the text appears in the second line of the message. In addition to a header line that indicates when the log message was last updated, a system log message can contain up to 50 lines of information. Each line may contain up to 76 characters, including blanks.

Example—Creating a New System Log Message

If you enter:

```
set logmsg 1 * system was cold started at 10:34 1/19/91
set logmsg 2 * all files were purged
```

users receive the following system log message the next time they log on:

```
LOGMSG - 10:37:28 EDT WEDNESDAY 01/23/98
* SYSTEM WAS COLD STARTED AT 10:34 1/22/98
* ALL FILES WERE PURGED
```

Note: If you do not include the asterisk when you enter the SET LOGMSG command, users must enter the QUERY LOGMSG command to receive the message; they do not automatically receive it when they log on. You can therefore create a log message with:

- Basic messages (on lines with asterisks in the first position)
- More detailed messages (on lines without asterisks).

If you create a log message with both basic and more detailed messages, the log message is short when users log on but complete when they enter the QUERY LOGMSG command.

If you shut down the z/VM system, CP saves the system log message. Then, if you bring up the system using a warm start, the log message contains the same information it had before you shut down. If you did not shut down the system with the SHUTDOWN command, or if you bring up the system with a force or cold start, you may have to create a new system log message.

Changing an Existing System Log Message

To change a specific line in an existing log message, enter the following command to find out its line number:

```
query logmsg
```

In response to this command, CP displays the log message. A log message contains:

- A header line that indicates the time the log message was last updated.

  A sample header line might look like this:

  ```
  LOGMSG - 10:37:28 EDT WEDNESDAY 01/23/98
  ```

- Up to 50 lines containing the line number and the log message text.

  CP displays one line for each nonnull log message line. If all of the lines are null, CP sends you a message saying that there is no LOGMSG data. In addition, CP displays line numbers for nonnull log message lines only if you have class B privileges.

Once you know the line number of a specific line, you can enter a SET LOGMSG command to change it to the new text.

To delete a specific line from an existing log message, enter:

```
set logmsg nn null
```

where *nn* is the line number of the message you want to delete.

Example 1—Changing Lines in an Existing Log Message

Assume you have a 12-line system log message and you want to change the line that gives the time you plan to shut down today. When you enter the QUERY LOGMSG command, you find that the line you want to change is the fifth line. To change line five of the log message, enter:

```
set logmsg 5 * system will shutdown today at 1700
```

In response to this command, CP replaces line 5 of the log message with this new line. It displays the log message header and all 12 lines when a user logs on or enters the QUERY LOGMSG command.

Example 2—Deleting Lines in an Existing Log Message

Assume you have a 12-line system log message and you want to delete the last three lines. To delete lines 10, 11, and 12 of the log message, enter:

```
set logmsg 10 null
set logmsg 11 null
set logmsg 12 null
```

In response to this command, CP deletes the lines. When a user logs on or enters the QUERY LOGMSG command, CP displays the log message header and the new nine-line log message.

When you change the system log message, send a message to all logged-on users. The message should say that the log message has changed and that they can enter the QUERY LOGMSG command to display the new one. When a user enters the QUERY LOGMSG command, CP displays the updated log message on the user's display screen. (For information on sending messages, see "Sending Messages to z/VM Users" on page 75.)

Note: If you have a CMS virtual machine, you can create a CMS EXEC file to make it easier to update the system log message. For information on how to create a CMS EXEC file, see the correct z/VM CMS book.

Deleting the Entire System Log Message

To delete the entire system log message, enter:

```
set logmsg null
```

In response to this command, CP deletes the log message. When you enter a QUERY LOGMSG command, CP displays only the header line.

Sending Messages to z/VM Users

In addition to including information in the system log message, you can communicate with z/VM users by sending a:

- General information message to all logged-on users
- General information message to a specific user
- Warning message to all logged-on users
- Warning message to a specific user.

Note:

1. The MESSAGE and WARNING commands are limited by the length of the input area. If the entire text of a message does not fit in this area, enter another command with the remaining text.
2. If an external security manager (ESM) is installed, you may not be authorized to use the MESSAGE or WARNING commands. However, messages sent to or from the system operator and messages sent with the ALL option are not subject to authorization checking by the ESM. For additional information, contact your security administrator.
3. If CP does not issue your message the way you entered it, it may be because you are including special line-editing symbols in the text of your message. For example, if logical line editing is in effect, the # symbol is your logical line-end symbol; if you include a # in your message text, CP cuts off your message.

   To prevent CP from interpreting these symbols as logical line-editing functions, enter:

   ```
   set linedit off
   ```

   CP then issues your messages as you enter them.

Sending a General Information Message to All Users

To send a general information message, such as "Query log for weekend schedule," to all logged-on users, enter:

```
message all query log for weekend schedule
```

If your system is a member of an SSI cluster, and you want to send the message to all logged-on users in the cluster, enter:

```
message all at all query log for weekend schedule
```

The following message appears on the display screen of all users able to receive the message:

```
hh:mm  * MSG FROM OPERATOR: QUERY LOG FOR WEEKEND SCHEDULE
```

where *hh:mm* indicates the time the message was sent.

Each user receives the message right away unless:

- An action is pending at the user's display

  In this case, the user receives the message when the action completes.
- The user is running in full-screen mode or has entered the SET MSG OFF command.

  In these cases, the user does not receive the message and you receive an error message.
- The user is not logged on.

Sending a General Information Message to a Specific User

Sometimes you need to send only a general information message to a specific user. For example, if you need to inform user VMUSER1 that the user is not authorized to use pack RC015, which the user just asked you to mount, enter the following command:

```
message vmuser1 you are not authorized to use pack rc015
```

As soon as there is no action pending at VMUSER1's display, the message appears on the display screen (provided the user is not running in full-screen mode or has not entered the SET MSG OFF command). Only VMUSER1 receives this message.

Sending a Warning Message to All Users

Occasionally, you might have a system problem that requires you to take some action, such as restarting an SFS server, that could disrupt a user's virtual machine operation. When this happens, enter the following command, instead of MESSAGE ALL, to inform all users of the situation:

```
warning all SFS server VMHOME: will be recycled in 5 minutes
```

If your system is a member of an SSI cluster, and you want all logged-on users in the cluster to receive the message, enter:

```
warning all at all SFS server VMHOME: will be recycled in 5 minutes
```

In response, CP displays the message on all users' display screens within the next 60 seconds (provided they are not running in full-screen mode and have not entered the SET WNG OFF command). This should give them time to save what they are currently working on.

⚠ **Attention:** If you use the MESSAGE command to send a warning, a user with an action pending does not receive it until that action completes, which might be too late.

Sending a Warning Message to a Specific User

Sometimes you might have to send a high-priority message to a specific user. For example, user MVSOPR1 might submit a high-priority file that needs to be processed on printer 00E, but the printer is dedicated to user VMUSER1. Before you reassign the printer to user MVSOPR1, you might want to send user VMUSER1 the following warning message:

```
warning vmuser1 system needs printer 00e - draining in 5 minutes
```

If user VMUSER1 is not running in full-screen mode and has not entered the SET WNG OFF command, the user receives this message within the next 60 seconds and should stop sending files to this printer for processing.

Updating the Production System's Log Message

To update your production system's log message (if any), enter the appropriate production system commands from your production system console.

Sending Messages to Production System Users

To send messages to your production system users, enter the appropriate production system commands from your production system console.

Summary of Running the System

To run a z/VM system, you need to:

1. Check the status of the system.

   This includes checking its performance, processors, cryptographic hardware, storage, devices, service virtual machines, and users.

   The items to check are:

   - System performance (see "Collecting Performance Information" on page 180)
   - The status of the system's real processors and their scheduling (see "Controlling Processors" on page 64)
   - The status of the system's Crypto Express adapters (see "Finding out the Status of the Crypto Resources" on page 66)
   - The status of the system's real storage (see "Controlling Storage" on page 66)
   - The status of the system's local devices (see "Checking the Status of Real Devices" on page 41)
   - The status of the accounting and error recording (EREP) virtual machines (enter the CP QUERY RECORDING command)
   - The status of the licensed program virtual machines (see the publications listed in the Bibliography).

   To check the status of users, use the following CP commands:

   - QUERY USERS—to find out how many users are logged on and dialed
   - QUERY NAMES—to find out which users are logged on, as well as the real or logical device number of the display to which each user is logged on
   - QUERY *userid*—to find out if a specific user is logged on
   - INDICATE USER *userid*—to find out a user's system resource use
   - QUERY SHARE *userid*—to find out a user's scheduling share and if a processor is dedicated to the user's virtual machine
   - QUERY MAXUSERS—to find out the maximum number of users allowed to log on to the system
   - QUERY RESPOOL ALL—to find out what resource pools are defined on the system and how many users are assigned to each pool
   - QUERY RESPOOL USER *userid*—to find out if a user is assigned to a resource pool
   - QUERY RESPOOL POOL *poolname* MEMBERS—to find out the CPU type and CPU limit setting for a resource pool and list the assigned users
2. Watch the system for changes and requests.

   This includes watching the z/VM system console and the production system's system console.
3. Control local devices.

   To control local devices, see Chapter 6, "Controlling Local Devices," on page 81.
4. Control remote devices.

   - To control remote devices, use VM/Pass-Through commands.

- To control SNA/CCS terminals, see Chapter 6, “Controlling Local Devices,” on page 81. Use VCNA, VSCS, or VTAM commands.

5. Control processors and their scheduling.

   To control processors or their scheduling, use the following CP commands:

   - INDICATE LOAD—to find out which processors are active
   - QUERY PROCESSORS—to find out how CP is using processors
   - VARY ONLINE PROCESSOR—to make a processor available to the system
   - VARY OFFLINE PROCESSOR—to make a processor unavailable to the system
   - SET SHARE—to change a user's scheduling share
   - SET SRM IABIAS—to change the system's interactive bias
   - SET SRM DSPSLICE—to change the size of the dispatching time slice that CP assigns each user
   - SET SRM STORBUF—to partition host storage
   - SET SRM LDUBUF—to partition the system's paging resources when moving users from the eligible to the dispatch lists
   - DEDICATE—to dedicate a processor to a virtual machine
   - UNDEDICATE—to remove sole use of a processor from a virtual machine
   - DEFINE RESPOOL—to create a resource pool that limits the amount of CPU resources that a group of users are allowed to consume in aggregate
   - SCHEDULE—to assign a user to a resource pool or remove a user from a resource pool
   - SET RESPOOL—to change the CPU limit setting for a resource pool

6. Query the status of Crypto Express adapters.

   To query the status of the crypto resources, use the following CP commands:

   - QUERY CRYPTO—to determine if Crypto Express adapters are installed on the system.
   - QUERY CRYPTO DOMAINS—to determine the device status, configuration state and device usage of each crypto resource available on the system.

7. Control storage.

   To control storage, use the following CP commands:

   - QUERY FRAMES—to find out how real storage is allocated
   - QUERY SXSSTORAGE—to find out the host logical storage addresses of the system execution space
   - QUERY SXSPAGES—to find out the status of system execution space pages
   - SET RESERVED—to reserve pages for a virtual machine in host storage
   - LOCK—to lock virtual machine pages in host storage
   - UNLOCK—to release pages in host storage
   - LOCATE (Storage)—to locate data in host storage
   - DISPLAY (Host Storage)—to display areas of host storage
   - DUMP (Host Storage)—to dump areas of host storage
   - STORE (Host Storage)—to change values in host storage

8. Control the system's service virtual machines.

   To control the accounting and error recording virtual machines, see “Collecting Accounting Information” on page 183 or “Collecting Hardware Error Information” on page 190, respectively.

   To control TCP/IP, RSCS, and other z/VM facilities and features, use commands appropriate to each feature or facility.

   To control VM/Pass-Through Facility and other licensed products, use commands appropriate to each product.

9. Communicate with users.

   To communicate with z/VM users, use the following CP commands:

   - QUERY LOGMSG—to display the system log message
   - CPACCESS — to give CP access to new logmsg files
   - CPRELEASE— to remove CP access to logmsg files
   - SET LOGMSG *nn * text*—to change the system log message so that every user sees the text of this line at logon
   - SET LOGMSG *nn text*—to change the system log message so that users must enter QUERY LOGMSG to read the text of this line
   - SET LOGMSG *nn* NULL—to delete a line from the system log message
   - SET LOGMSG NULL—to delete the entire system log message
   - MESSAGE ALL *text*—to send a message to all users
   - MESSAGE *userid text*—to send a message to a specific user
   - WARNING ALL *text*—to send a warning message to all users
   - WARNING *userid text*—to send a warning message to a specific user

   To communicate with production system users, use the appropriate production system commands and facilities.

Chapter 6. Controlling Local Devices

This chapter tells you how to use z/VM commands to control devices connected to your processor complex. After reading this chapter, you can control:

- Channel paths to devices
- Dynamic switching devices
- Displays, 3270 printers, and communication lines
- Systems network architecture (SNA/CCS) terminals
- Direct access storage devices (DASDs)
- Tape drives
- Channel-to-channel adapters (CTCAs)
- Printers and printer spool file queues

Note: z/VM also provides the ability to change the real I/O configuration. CHPIDS, CONTROL UNITS, and DEVICES can be added, deleted, and modified dynamically without an IPL or POR (Power® On Reset).

How This Chapter Is Organized

This chapter begins with an overview of how devices are used by z/VM. Following the overview are sections that tell you how to control channel paths and each type of local device:

- "Channel Paths and Programs" on page 82
- "Dynamic Switching Devices" on page 84
- "Displays, 3270 Printers, and Communication Lines" on page 84
- "SNA/CCS Terminals" on page 92
- "Direct Access Storage Devices (DASDs)" on page 93
- "Tape Drives" on page 116
- "Channel-to-Channel Adapters (CTCAs)" on page 125
- "Printers" on page 131

As part of the sections on printers, this chapter also describes how to manage spool files. When you need information about a spool file in a printer or reader queue, see the appropriate section:

- "Controlling Spool Files in the Print Queue" on page 153
- "Controlling Spool Files in the Reader Queue" on page 162

For a summary of the commands you use to control devices, see Appendix A, "Common Operator Commands for z/VM," on page 247.

Overview of Local Devices

As a z/VM operator, you need to understand how z/VM uses devices that are available to it. You will be controlling devices that:

- Support the system itself.

  These are the DASDs that contain the system configuration files, the user directory, paging and spooling space, and temporary disk space. These devices contain space or data that z/VM needs in order to run.

  This category also includes the dump device. The dump device contains space that z/VM needs to produce a CP abend or stand-alone dump.
- Provide services for virtual machines.

This category includes real spooling devices, which process the printer and reader spool files. Also, channel-to-channel adapters allow virtual machine users to access remote systems, send data to remote systems, and provide communications for loosely coupled multiprocessing systems. The IBM Enterprise Systems Connection Architecture (ESCON®) I/O interfaces allow communication of data by using dynamic switching devices that connect multiple control units and ESCON channels designed to operate on fiber optic cables. This support provides reduced cable bulk, removes distance limitations on attachment of devices, improves security, and optimizes performance of devices and channels.

- Support the production system's virtual devices.

  These are devices that are dedicated to the production system's virtual machine for its exclusive use. Any type of device may be dedicated to the production system: dynamic switching devices, displays, 3270 printers, communication lines, DASDs, tape drives, CTCAs, printers, and punched card devices.

- Support z/VM users' virtual devices.

  Most devices that z/VM users use are not dedicated to them for their exclusive use. These include displays, 3270 printers, and communication lines that z/VM users use, and the DASDs that contain their minidisks.

  However, you may dedicate devices to z/VM users, especially if they are running test machines.

Channel Paths and Programs

A *channel path* connects a device to a processor complex's channel subsystem through a control unit or by direct attachment. The I/O configuration data set (IOCDS) that you select during system initialization defines the channel paths on the processor complex, the control units attached to the channel paths, and the I/O devices assigned to the control units.

The IOCDS defines the physical availability of a channel path to a device. z/VM allows you to control its logical availability. If a channel path is offline to a device, it is logically unavailable to that device. This means that CP prevents I/O to the device from using the channel path, even though the channel path may be physically available.

To control the channel paths logically available to devices, you must be able to:

- Find out channel path status
- Change the logical availability of a channel path to a device.

A *channel program* is one or more channel command words that control a specific sequence of data channel operations. Execution of the specific sequence is initiated by a single Start I/O instruction.

You can stop execution of an active channel program on a specified real device by using the HALT command.

Attention: Use the HALT command only in extreme cases and only after careful consideration. Indiscriminate use of HALT can cause unwanted results.

To stop execution of an active channel program, enter:

```
halt rdev
```

where *rdev* is the device number of the real device you want to stop.

After entering HALT, you should enter the QUERY (real devices) command to determine the status of the device. If the device is in an intervention-required state, you should make the device ready.

The following response indicates that CP has accepted the command and initiated action to halt the device:

```
HALT INITIATED TO type rdev
```

The following response indicates that HALT was issued to an idle device and that CP has not accepted the command:

```
HALT NOT INITIATED BECAUSE type rdev IS NOT ACTIVE
```

Finding Out Channel Path Status

To find out channel path status by device, enter:

```
query paths to rdev
```

where *rdev* is the real device number of the device.

In response, CP lists all the channel paths that are physically available to the device and indicates whether the channel paths are logically online or offline.

To display the status of the machine's channel paths, enter:

```
query chpids
```

To find out channel path status by channel path, enter:

```
query chpid nn
```

where *nn* is the identifier of the channel path.

In response, CP lists all of the devices to which the channel path is physically available and, for each device, indicates whether the channel path is logically online or offline.

Changing Channel Path Availability to a Device

Usually, your installation sets up the I/O configuration of your system before you bring up z/VM, in such a way that you do not have to change the availability of channel paths to devices. However, under certain error conditions, it can be useful to remove channel paths to devices temporarily while errors are being corrected. For example, when an error occurs that prevents a channel path from being used, you can remove the channel path from any devices to which it is physically available.

In addition, if devices are connected to multiple control units, you can isolate a control unit that needs service without affecting operation. In this case, you can remove the channel paths involving the control unit without affecting other channel paths.

To remove a channel path from a device:

1. If the device is in use by a virtual machine, *make sure the channel path is not in use*.

   For example, if the device is in use by the production system, enter at the production system's system console any commands the production operating system requires to release the channel path from use.

   IBM strongly recommends that you perform this step first to bring to an orderly end any I/O operations using the channel path that have been initiated by the virtual machine.
2. Enter:

   ```
   vary offline path nn from rdev
   ```

   where

   nn

   is the identifier of the channel path.

   rdev

   is the real device number of the device.

In response, CP removes the channel path logically from the device. You can now remove the path physically without affecting CP operations.

To make a channel path available to a device, enter:

```
vary online path nn to rdev
```

where

nn
is the identifier of the channel path.

rdev
is the real device number of the device.

In response, CP makes the channel path logically available to the device.

Note: If the path to a device is varied offline by the configuration manager, you must use the FORCE option of the VARY PATH command to vary the path online. If the FORCE option is not used, you receive a "request rejected" message. For more information about the VARY PATH command, see the *z/VM: CP Commands and Utilities Reference*.

If the device status of the device you are attempting to logically vary the path on is SUBCHOFF, then you must first vary the subchannel online. An attempt to vary on a path to a device that is in the SUBCHOFF state will fail with a message. For more information about the VARY SUBCHANNEL command, see the *z/VM: CP Commands and Utilities Reference*.

You may then either restore use of the channel path to the appropriate virtual machine yourself, or notify the operator of the virtual machine that the channel path is again available for use.

Dynamic Switching Devices

Dynamic switching devices connect control units with ESCON channels designed to operate on fiber optic cables. To control dynamic switching devices you must be able to:

- Find out their status
- Make them available to the system
- Make them available for access by z/VM users
- Dedicate them to the production system
- Remove them from the production system
- Remove them from access by z/VM users
- To make them unavailable to the system

These tasks for dynamic switching devices, as well as for other devices, are described in the following section, "Displays, 3270 Printers, and Communication Lines" on page 84.

Displays, 3270 Printers, and Communication Lines

To control displays, 3270 printers, or communication lines, you must be able to:

- Find out their status
- Make them available to the system
- Make them available for access by z/VM users
- Dedicate them to the production system
- Remove them from the production system
- Remove them from access by z/VM users
- To make them unavailable to the system
- Set up a 3270 printer to print a copy of a display screen
- Determine if a security label is associated with the printer.

Note: You may operate the IBM 5080 Graphics System as a dedicated display.

Finding Out the Status of These Devices

To find out the status of a specific display, 3270 printer, or communication line enter:

```
query rdev
```

where *rdev* is the real device number of the device. To find out the status of all the displays and 3270 printers in your z/VM system, enter:

```
query graf all
```

To find out the status of all the communication lines in your z/VM system, enter:

```
query lines all
```

To find out the status of all the dynamic switching devices in your z/VM system, enter:

```
query swch all
```

Table 3 on page 85 lists the keywords that appear in responses to QUERY commands for displays, 3270 printers, communication lines, and dynamic switching devices. It also shows:

- The meaning of each keyword
- Whether the keyword indicates an active, free, or offline device
- Whether the keyword indicates a dedicated device or a device in use by CP.

Table 3. Display, 3270 Printer, Communication Line, and Dynamic Switching Device Status

Device Status Keyword	Keyword Meaning	Type of Status
ATTACHED TO	The device is dedicated to the indicated virtual machine.	Active (dedicated)
DIALED TO	The device is dedicated to the indicated virtual machine through the DIAL command.	Active (dedicated)
DISABLED	The device is unavailable to users.	Free
ENABLED	The device is available to users but is not in use.	Free
INT REQUIRED	The device requires operator intervention.	Active (CP or dedicated)
LOGON AS	A user has logged on to the device (display only).	Active (CP)
OFFLINE	The device is unavailable to CP.	Offline
SEC	Specifies the security label of files that can be printed on the printer. This is used only when an external security manager is installed and security label checking is enabled. If a security label is not assigned to the printer, the SEC field is followed by the word NONE. No files can be printed until the printer is STARTed with a valid security label.	Active (CP)
SUBCHOFF	The subchannel for this device is disabled.	Offline Disabled

Making a Display, a 3270 Printer, or Communication Lines Available to the System

If a display, 3270 printer, or communication line is offline, you must make it available to the system in order to use it. To make it available to the system, enter:

```
vary online rdev
```

where *rdev* is the real device number of the device.

Now you may either make it available for access by z/VM users, or you may dedicate it to the production system.

Making a Display, a 3270 Printer, or Communication Lines Available for User Access

To make a display, 3270 printer, or communication line available for access by z/VM users, do the following:

1. Make sure the device status says DISABLED. Enter:

   ```
   query rdev
   ```

   where *rdev* is the real device number of the device.

   If the device status says OFFLINE, first follow the procedure described under "Making a Display, a 3270 Printer, or Communication Lines Available to the System" on page 86.

   If the device status says DISABLED, go to the next step.
2. Enter:

   ```
   enable rdev
   ```

   where *rdev* is the real device number of the device.

If the device is a display or communication lines, a z/VM user may now use it to log on to the system. If it is a 3270 printer, a z/VM user may now use it to get a hard copy of the user's display screen. See "Setting Up a 3270 Printer to Print a Copy of a Display Screen" on page 91.

Example 1—Making a Display Available for User Access

z/VM user VMUSER1 is unable to log on display 4C3. When you enter:

```
query 4c3
```

CP tells you that display 4C3 is offline. To make it available for access by VMUSER1:

1. Enter:

   ```
   vary online 4c3
   ```
2. Enter:

   ```
   enable 4c3
   ```

VMUSER1 should now be able to log on z/VM or enter the DIAL command to gain access to an operating system running in a virtual machine.

Example 2—Making a 3270 Printer Available for User Access

z/VM user VMUSER1 has logged on display 4C3 and now needs access to 3270 printer 4C4. When you enter:

```
query 4c4
```

CP tells you that 3270 printer 4C4 is offline. To make it available for access by VMUSER1:

1. Enter:

   ```
   vary online 4c4
   ```

2. Enter:

   ```
   enable 4c4
   ```

VMUSER1 should now be able to use 3270 printer 4C4.

Example 3—Making Communication Lines Available for User Access

z/VM user VMUSER1 is unable to log on communication line 4C5. When you enter:

```
query 4c5
```

CP tells you that communication line 4C5 is offline. To make it available for access by VMUSER1:

1. Enter:

   ```
   vary online 4c5
   ```

2. Enter:

   ```
   enable 4c5
   ```

VMUSER1 should now be able to log on z/VM.

Dedicating These Devices to the Production System

To dedicate a display, 3270 printer, or communication line to the production system for its exclusive use:

1. Make sure the device status says DISABLED. Enter:

   ```
   query rdev
   ```

   where *rdev* is the real device number of the device.

   If the device status says OFFLINE, first follow the procedure described under "Making a Display, a 3270 Printer, or Communication Lines Available to the System" on page 86.

   If the device status says GRAF *rdev* LOGON AS or GRAF *rdev* DIALED TO, you must wait until it is no longer in use.

   If the device status says ENABLED, first follow the procedure described under "Removing These Devices from the Production System" on page 89.

   If the device status says DISABLED, go to the next step.

2. Enter:

   ```
   attach rdev userid vdev
   ```

   where:

   rdev
   is the real device number of the device.

userid
is the user ID of the production system.

vdev
is the virtual device number you want to assign to the device.

Note: If *rdev* and *vdev* are the same, you may enter:

```
attach rdev userid
```

3. At the production system's system console, enter any commands that the production operating system requires in order to use the device.

Example 1—Dedicating a Display to the Production System

The display 4C1 should be dedicated to the production system, MVSESA, as virtual device BC1. When you enter:

```
query 4c1
```

CP tells you that display 4C1 is offline. To dedicate it to the production system, MVSESA:

1. Enter:

   ```
   vary online 4c1
   ```

2. Enter:

   ```
   attach 4c1 mvsesa bc1
   ```

3. At the MVSESA system's system console, enter appropriate z/OS commands to make sure the display is usable.

Example 2—Dedicating a 3270 Printer to the Production System

The 3270 printer 5E1 should be dedicated to the production system, VSEAF1, as virtual device CE1. When you enter:

```
query 5e1
```

CP tells you that 3270 printer 5E1 is offline. To dedicate it to the production system, VSEAF1:

1. Enter:

   ```
   vary online 5e1
   ```

2. Enter:

   ```
   attach 5e1 vseaf1 ce1
   ```

3. At the VSEAF1 system's system console, enter appropriate VSE commands to make sure the 3270 printer is usable.

Example 3—Dedicating Communication Lines to the Production System

The communication line 6E1 should be dedicated to the production system, VMESA, as virtual device DEL. When you enter:

```
query 6e1
```

CP tells you that communication line 6E1 is offline. To dedicate it to the production system, VMESA:

1. Enter:

   ```
   vary online 6e1
   ```

2. Enter:

   ```
   attach 6e1 VMESA del
   ```

3. At the VMESA system's system console, enter appropriate z/VM commands to make sure the display is usable.

Removing These Devices from the Production System

Removing a display, 3270 printer, or communication line from the production system is the opposite of dedicating it:

1. At the production system's system console, enter any commands that the production operating system requires in order to release the device from use.
2. Enter:

   ```
   detach rdev userid
   ```

   where:

   rdev
   is the real device number of the device.

   userid
   is the user ID of the production system.

A z/VM user may now use the display, communication line, or 3270 printer; or you may dedicate it to another virtual machine.

Example 1—Removing a Display from the Production System

To remove display 4C1 from the production system, MVSESA:

1. At the MVSESA system's system console, enter appropriate z/OS command s to make sure the display is not in use.
2. Enter:

   ```
   detach 4c1 mvsesa
   ```

Example 2—Removing a 3270 Printer from the Production System

To remove 3270 printer 5E1 from the production system, VSEAF1:

1. At the VSEAF1 system's system console, enter appropriate VSE commands to make sure the 3270 printer is not in use.
2. Enter:

   ```
   detach 5e1 vseaf1
   ```

Example 3—Removing Communication Lines from the Production System

To remove communication line 6F1 from the production system, VMESA0:

1. At the VMESA system's system console, enter appropriate z/VM commands to make sure the display is not in use.
2. Enter:

   ```
   detach 6f1 vmesa
   ```

Removing a Display, a 3270 Printer, or Communication Lines from User Access

Removing a display, 3270 printer, or communication line from user access prevents a z/VM user from using it to enter the LOGON or DIAL commands. To remove a display, 3270 printer, or communication line from user access:

1. Make sure it is not in use. Enter a QUERY command for the device to find out if it is.

   If it is in use, ask the user to log off. Make sure the status says ENABLED before you go to the next step. Otherwise, the DISABLE command does not remove the device from user access.

2. Enter:

   ```
   disable rdev
   ```

   where *rdev* is the real device number of the device.

The display, 3270 printer, or communication line is now unavailable for user access to z/VM. You may now dedicate the device to the production system or make it unavailable to the system.

Attention: If you enter a DISABLE ALL command, immediately enter:

```
enable rdev
```

where *rdev* is the real device number of your primary system console. Doing this prevents CP from locking you out of the system if you accidentally get logged off or disconnected.

If you enter a DISABLE ALL command, CP disables all inactive displays, 3270 printers, communication lines, 328x printers, and SNA/CCS terminals. After each user logs off or disconnects, CP also disables that user's display. If you, as the primary system operator, log off or get disconnected, CP disables your display as well. In this circumstance, you can only log on again if one of the following is true:

- A still logged-on user with the right privilege class enters the ENABLE command to reenable your display
- You reload CP on the processor complex's system console.

However, if you immediately reenable your display, you avoid this situation.

Making a Display, a 3270 Printer, or Communication Lines unavailable to the system

Making a display, 3270 printer, or communication line unavailable to the system is the opposite of making it available. p: To make it unavailable to the system:

1. Make sure it is not in use. Enter a QUERY command for the device to find out if it is.

   If the status indicates that a user is logged on the display or communication line or is using the 3270 printer, follow the procedure at "Removing a Display, a 3270 Printer, or Communication Lines from User Access" on page 90. The status should read DISABLED before you go to the next step.

   If the status indicates that it is dedicated to the production system, follow the procedure at "Removing These Devices from the Production System" on page 89. The status should read DISABLED before you go to the next step.

2. Enter:

   ```
   vary offline rdev
   ```

   where *rdev* is the real device number of the device. The display, 3270 printer, or communication line is now unavailable to z/VM.

Setting Up a 3270 Printer to Print a Copy of a Display Screen

You can set up a PF key on your display to copy its screen to a 3270 printer. To do this:

1. Make sure the 3270 printer is enabled.

   If it is not, use the commands described under "Making a Display, a 3270 Printer, or Communication Lines Available to the System" on page 86. When you enter the QUERY command for the device, the status should read ENABLED before you go to the next step.

2. Enter:

   ```
   set pfnn copy rdev
   ```

   where:

   nn
   is the number of one of the PF keys on your display.

   rdev
   is the real device number of the 3270 printer.

When you press the PF*nn* key, a copy of your display screen goes to the 3270 printer you specified.

Note:

1. Some displays have a PRINT key that allows you to copy the contents of your display screen to a 3270 printer attached to the same control unit without intervention by CP. Although using the SET PF COPY command is similar, it is not the same as this hardware function.
2. Communication lines do not support the PF COPY command.

Transferring Files from a 3270 Printer

The TRANSFER command transfers files from the printer queue of another user. If an ESM is installed, you may not be authorized to use the TRANSFER command. For additional information, contact your security administrator.

To transfer spool files from a printer, enter:

```
transfer prt class c|userid1|all to|from all|*|userid2 rdr|prt|pun
```

where:

CLASS *c*
is the class of the spool files you want to transfer.

userid1
is the user whose files are to be transferred or reclaimed.

ALL
specifies the transfer of all files in the queue.

TO|FROM *
indicates to or from your own userid.

userid2
identifies the user to whom the files are transferred.

Example 1—Transferring a Spool File from the Printer to a User

A spool file with spoolid 0123 needs to be transferred from the print queue to the reader queue of USERX. To transfer the spool file, enter:

```
transfer prt 0123 to userx
```

Example 2—Transferring Spool Files from the Printer to Your Reader

Spool files on your printer queue need to be transferred to your reader queue. To transfer the spool files, enter:

```
transfer prt all to * rdr
```

Example 3—Transferring Class B Files from the Printer Queue to Your Reader

Class B files on your printer queue need to be transferred to your reader queue. To transfer the class B files, enter:

```
transfer prt class B to * rdr
```

Example 4—Reclaiming Spool Files Sent to Other Reader Queues

You need to reclaim spool files sent to the reader queues of other users and place them on your printer queue.

To do this, enter:

```
transfer prt all from all rdr
```

SNA/CCS Terminals

An SNA/CCS terminal is any terminal accessing z/VM that is managed by a VTAM service machine.

Systems Network Architecture (SNA) is a logical structure containing formats, protocols, and operational sequences through which you can transfer information units and control the configuration and operation of networks. It allows the VTAM service machine to control the appearance of a z/VM terminal screen attached to the SNA network.

Terminals managed by the VTAM service machine may be:

- Local or remote
- SNA or non-SNA.

They may use communication lines or EBCDIC character sets and be graphic or TTY-type terminals. To find out the status of an SNA/CCS terminal, you may need to use CP commands, VCNA commands, VSCS commands, or VTAM commands. See VM/VTAM publications for the specific terminals supported.

In order to control an SNA/CCS terminal, you must be able to:

- Make it available for access by z/VM users
- Remove it from access by z/VM users.

Making a SNA/CCS Terminal Available for User Access

To make a SNA/CCS terminal available for access by z/VM users, enter:

```
enable sna userid
```

where *userid* is the user ID of the VTAM service machine that owns the SNA/CCS terminal.

A z/VM user may now use it to communicate through SNA.

Example: Making a SNA/CCS Terminal Available for User Access

A SNA/CCS terminal user wants to access z/VM through VTAM service machine VSERV1. To make it available for access, enter:

```
enable sna vserv1
```

All SNA/CCS terminals owned by VTAM service machine VSERV1 can now access z/VM.

Removing an SNA/CCS Terminal from User Access

Removing a SNA/CCS terminal from user access prevents a user from using it to access z/VM. To remove a SNA/CCS terminal from user access, enter :

```
disable sna userid
```

where *userid* is the user ID of the VTAM service machine that owns the SNA/CCS terminal.

All SNA/CCS terminals controlled by the VTAM service machine are now unable to access z/VM.

⚠ **Attention:** If you enter a DISABLE ALL command, immediately enter:

```
enable rdev
```

where *rdev* is the real device number of your primary system console. Doing this prevents CP from locking you out of the system if you accidentally get logged off or disconnected.

If you enter a DISABLE ALL command, CP disables all inactive displays, 3270 printers, 328x printers, and all SNA/CCS terminals. After each user logs off or disconnects, CP also disables that user's display. If you, as the primary system operator, log off or get disconnected, CP disables your display as well. In this circumstance, you can only log on again in one of the following circumstances:

- A still logged-on user with the right privilege class enters the ENABLE command to reenable your display
- You reload CP on the processor complex's system console.

However, if you immediately reenable your display, you avoid this situation.

Direct Access Storage Devices (DASDs)

DASD volumes contain much of your installation's data. Accordingly, you must spend a lot of your time controlling DASDs.

In this book, *DASD* refers to any direct access storage device. *Cached DASD* refers to a DASD that is attached to a storage control unit that contains a high-speed storage buffer called a *cache*. Examples of such control units include the IBM DASD subsystems.

Caching (writing to or reading from cache storage) can improve system performance. However, paging I/O is typically a poor candidate for cached DASD, therefore disable the cache for paging volumes or place the paging volumes elsewhere. You may need to perform more tasks to control cached DASDs than to control noncached DASDs.

Many of the DASD volumes in your z/VM system may be shared by different operating systems. These operating systems may run in production and test virtual machines, on other processor complexes, or both. Sharing data in this way affects the way you control the DASD volumes. Accordingly, the next sections focus on shared DASD operation. "Controlling DASDs" on page 97 then describes the main DASD operation tasks.

Operating with Shared DASD Volumes

Your z/VM system probably includes a number of shared DASD volumes. Therefore, you should:

- Be aware of the effect of sharing DASDs on your operation of the system
- Understand how z/VM permits sharing of DASD volumes
- Be familiar with the facilities available for controlling the sharing of DASD volumes, when necessary.

How Sharing DASDs Affects System Operation

Many installations run more than one system on more than one processor complex. In these installations, the systems generally share DASDs with each other. If this is true at your installation, you should make sure you know when DASDs are in use by more than one system.

The production system you are running in a virtual machine under z/VM may share one or more of its DASDs with one or more operating systems. These operating systems may run in virtual machines under z/VM, on other processor complexes, or both. When this is true, you must be careful when you switch a DASD volume from one virtual machine to another. You must also be careful when you remove the volume from z/VM 's control (when you vary it offline) or when you shut down the system.

Normally, being careful means you must make sure that the DASD volume is unavailable for use by the other system. Before you detach or vary offline a DASD volume, or enter the SHUTDOWN command, contact the operators of the other systems and follow your installation's normal procedures.

Understanding Shared DASD Support

z/VM allows your installation to share DASD volumes safely through reserve/release support. Understanding this support may help you to control shared DASDs.

Note: DASD volumes can also be shared among multiple systems in a z/VM single system image (SSI) cluster or among multiple non-SSI systems using cross-system link (XLINK). For further information, see the *z/VM: CP Planning and Administration*.

The purpose of reserve/release support, in general, is to preserve data integrity while still allowing several operating systems to use the same DASD volume. Preserving data integrity guarantees that the results of operations on data are correct. Reserve/release guarantees this by allowing only one operating system at a time to access data on a DASD volume. Serializing access to data in this way prevents one system from reading data that another system is changing at the same time. Without this serialization, the results of operations on such data could not be guaranteed to be correct.

Serializing access to data allows several operating systems to use the same DASD volume. If the operating systems are running on different processor complexes, the access is serialized through real reserve/release support. If the operating systems are running in different z/VM virtual machines, the access is serialized through virtual reserve/release support. If a DASD volume is shared by *both* kinds of operating systems, its data integrity is preserved by concurrent virtual and real reserve/release support.

Real Reserve/Release Support

Real reserve/release support makes use of a hardware function available on all the DASDs z/VM supports. To communicate with DASDs and other devices, operating systems use channel command words (CCWs). To serialize access to DASDs, operating systems use particular CCWs called reserve/release CCWs. A reserve CCW exclusively reserves a DASD volume for the issuer. A release CCW releases the volume, allowing another operating system to reserve the device. To share a DASD volume among a z/VM virtual machine and operating systems running on other processor complexes, your installation must set up the volume so that its sharing invokes CP's real reserve/release support.

To invoke real reserve/release support, your installation must dedicate the device to a virtual machine. Your installation dedicates the device by coding a DEDICATE statement in the user directory. Alternatively, you may invoke the support by entering the ATTACH command.

Virtual Reserve/Release Support

Virtual reserve/release support is a software function of CP. When a DASD volume is shared only among operating systems running in z/VM virtual machines, CP can itself perform the serialization of access to data on the volume. In this case, CP virtually reserves and releases the device and does not invoke the hardware reserve/release function. To share one DASD volume strictly among virtual machines, your installation must set up the volume so that its sharing invokes CP's virtual reserve/release support.

To invoke virtual reserve/release support, your installation must set up the volume as a virtually shareable minidisk. A *minidisk* is a virtual disk that contains either a portion or all of a real DASD volume. Your installation defines a virtually shareable minidisk by coding in the user directory an MDISK statement with the V option on the access mode.

You can use the same method to define a virtual disk in storage as shareable using virtual reserve/release support. A virtual disk in storage is a minidisk that is allocated in host storage instead of being mapped to a real DASD volume.

Concurrent Virtual and Real Reserve/Release Support

So far we have looked at real reserve/release and virtual reserve/release. Real reserve/release support allows your installation to share a DASD volume that is dedicated to one virtual machine with operating systems that run on other processor complexes. By contrast, virtual reserve/release support allows your installation to share a DASD volume only among operating systems running in z/VM virtual machines. To accomplish both kinds of sharing for the same volume requires concurrent virtual and real reserve/release support.

Concurrent virtual and real reserve/release support involves first the hardware and then the software reserve/release function. To share a DASD volume among operating systems in virtual machines *and* operating systems on other processor complexes, your installation must define the volume so that it invokes CP's virtual and real reserve/release support at the same time.

To invoke this concurrent support, your installation must set up the volume as a virtually shareable full-pack minidisk that is also shareable to CP. A *full-pack minidisk* is a virtual disk that contains all of a real DASD volume. Your installation defines a virtually shareable full-pack minidisk by coding in the user directory an MDISK statement with the V option on the access mode. Your installation defines the volume as shareable to CP by specifying:

- The SHARED operand on the SET RDEVICE (DASD) statement in the system configuration file
- The SET SHARED ON command.

Controlling the Sharing of DASD Volumes

As mentioned before, your installation can set up shared DASD volumes during system generation. You usually do not have to change that setup. But if your installation requires it (for example, if the number of test and production operating systems at your installation changes regularly), you may change the sharing of DASD volumes between systems.

To control the sharing of DASD volumes, you must be able to:

- Find out which DASDs are shared
- Change the way DASDs are shared
- Handle reserve/release errors.

Finding Out Which DASDs Are Shared

To find out which DASDs in your system are shared among multiple real and virtual systems:

1. Find out which DASDs are defined as virtually shareable full-pack minidisks.

   To do this, contact the person who maintains the z/VM user directory at your installation.
2. Find out which DASDs are defined as shareable to CP. Enter:

   ```
   query dasd system
   ```

   CP responds with the status of all active DASD in use by the system (the command excludes dedicated DASDs). If the status of a device is SHARED, the device is shareable to CP.

   If cross-system link (XLINK) is enabled, enter:

   ```
   xlink check volidn
   ```

where *volidn* identifies the volume or volumes to be checked in order to determine if they are under cross-system link control.

3. Compare the first list to the second list.

   Devices that appear on both lists meet the requirements for concurrent virtual and real reserve/release. These devices are eligible to be shared between operating systems running within z/VM virtual machines running on other processor complexes.

Note: CP issues real reserve/release CCWs only to a shared device that is also defined and in use as a full-pack minidisk.

To find out whether a particular DASD volume is shared among multiple real and virtual systems:

1. Find out if the DASD is defined as a virtually shareable full-pack minidisk.

   To do this, contact the person who maintains the z/VM user directory at your installation.

2. Find out if the DASD is defined as shareable to CP. Enter:

   ```
   query rdev
   ```

   Where *rdev is the real device number of the DASD. Again, if the status for the device is SHARED, the device is eligible to be shared among multiple real and virtual systems. But if the device is not also defined and in use as a full-pack minidisk, CP does not issue real reserve/release CCWs when accessing it.*

Changing a DASD's Shared Status

Your installation may share DASD volumes with operating systems on other processor complexes during certain times of day but not others. For performance reasons, your installation may wish to make DASD volumes shareable to CP only when needed; that is, only when the DASD must be shared with other processor complexes. Accordingly, you may use the SET SHARED command to change the shared status of a DASD volume.

Note: For a DASD volume to be shareable between real and virtual systems:

- Your installation must have defined it during system generation as a full-pack minidisk that is virtually shareable.
- The device must now be in use as a full-pack minidisk.

Also, note that you may enter the SET SHARED ON command for any DASD volume, but if the volume is not in use as a full-pack minidisk, CP does not issue real reserve/release CCWs to the device.

To make a DASD volume shareable by multiple real and virtual systems, enter:

```
set shared on for rdev
```

Where *rdev* is the real device number of the DASD. You may also specify a range of up to 256 devices. In response, if the device is a full-pack minidisk, CP issues real reserve/release CCWs when accessing the device.

You may also use the SET SHARED OFF command to remove real reserve/release support of the device, but you should do so with extreme caution and only after careful consideration. Before you enter the SET SHARED OFF command, contact the operator of any other system that shares the device and make sure the system is not using it. Follow your installation's normal procedures.

ATTENTION: Using the SET SHARED OFF command for a device could cause a data integrity exposure if an operating system on another processor complex is still using the device.

For more information about the SET SHARED command, see the *z/VM: CP Commands and Utilities Reference*.

Handling Reserve/Release Errors

Under certain error conditions, an operating system may hold a real reserve to a DASD volume for an abnormally long time. To find out if this is occurring, you may use the QUERY DASD RESERVE command.

To display the status of all reserved DASD, enter:

```
query dasd reserve
```

In response, CP displays the status of reserved DASD, if any. One of the following responses may indicate a problem with the device:

```
RESERVED BY userid or SYSTEM
```

This response tells you that either CP or an operating system running in the indicated virtual machine holds the reserve to the device.

```
RESERVE PENDING FOR userid or SYSTEM
```

This response indicates that an operating system running on another processor complex holds the reserve.

If you receive one of these responses and the condition persists, follow your installation's normal procedures to release the device.

If you must release a device that CP has reserved for a virtual machine, you may use the RESET RESERVE command.

ATTENTION: Using the RESET RESERVE command causes the device to become immediately available to another sharing system. Use the RESET RESERVE command with extreme caution and only after careful consideration.

For more information about the RESET RESERVE command, see the *z/VM: CP Commands and Utilities Reference*.

Controlling DASDs

To control DASDs, you need to be able to:

- Find out a DASD's status
- Locate a specific DASD volume
- Dedicate a DASD to a z/VM user or to the production system
- Remove a DASD from a z/VM user or from the production system
- Mount a disk pack for a z/VM user or for the production system.

To control system volumes, you must be able to:

- Attach a system volume to the system
- Detach a system volume from the system
- Check the usage of system volumes
- Check the format of a system volume and if necessary format and allocate it
- Change the volume identifier or label on a DASD volume.

You also need to be able to:

- Define a DASD as a dump device
- Inspect the physical condition of a DASD volume
- Copy one DASD volume to another

Some storage subsystems (such as the IBM DASD subsystem) provide nonvolatile storage (NVS), a DASD fast write function, and a dual copy function as well as regular caching. If you have a storage subsystem that provides those functions, you need to know how to activate, display the status of, and deactivate them.

Before you start, you should be aware of the effect of sharing z/VM DASD volumes with other operating systems. If you are not familiar with operating z/VM with shared DASD, see "Operating with Shared DASD Volumes" on page 93.

Note: The following information includes a discussion about the cache subsystem, nonvolatile storage (NVS), and DASD fast write. To manipulate data within the IBM DASD subsystem, you must activate the DASDFW, CACHE, and NVS features. To do this, enter the following three commands:

```
set cache subsys on
set nvs on
set dasdfw on
```

Managing Nonvolatile Storage

Nonvolatile storage (NVS) is random access storage with an electronic backup. This maintains system data integrity for DASD fast write data not yet written to DASD in the event of a storage subsystem power failure. NVS must be activated to exploit the DASD fast write function.

To make nonvolatile storage available or unavailable for a particular IBM DASD subsystem, enter:

```
set nvs subsystem on rdev
```

where:

subsystem
is optional, is the default, and specifies the entire storage subsystem, regardless of which device within the subsystem is affected.

on
specifies that the nonvolatile storage is made available to the subsystem.

off
specifies that the nonvolatile storage is made unavailable to the subsystem.

rdev
is the real device number, a list of real device numbers, or a range of real device numbers for devices in the subsystem.

Note:

1. SET NVS is only valid for cache storage subsystems containing nonvolatile storage. If the specified subsystem does not have nonvolatile storage, an error message is generated.
2. Making nonvolatile storage unavailable to the subsystem causes all modified DASD fast write data in the subsystem to be destaged.

To display the current state of the nonvolatile storage for a particular IBM DASD subsystem, use the QUERY NVS command. To query the current state of NVS, enter:

```
query nvs rdev
```

where *rdev* is the real device number, a list of real device numbers, or a range of real device numbers of devices in the subsystem being queried.

Note: QUERY NVS is only valid for cache storage subsystems containing nonvolatile storage. If the specified subsystem does not have nonvolatile storage, an error message is generated.

Finding Out a DASD's Status

To find out the status of a specific DASD, enter:

```
query rdev
```

where *rdev* is the real device number of the DASD. To find out the status of all the DASDs in your z/VM system, enter:

```
query dasd all
```

DASD fast write provides the appearance of speedier access to data residing on DASD in IBM DASD subsystems. It causes I/O operations to appear to complete more quickly by presenting device-end status with channel-end status as soon as the data is received by the control unit. If nonvolatile storage is turned on, the data is saved in nonvolatile storage; this eliminates the possibility of data loss because of power failure. The data is transferred from NVS to the backing-storage device by the control unit.

To display the current state of the DASD fast write function for a device attached to cache storage subsystems, enter:

```
query dasdfw rdev
```

where *rdev* is the real device number, a list of real device numbers, or a range of real device numbers of the devices being queried.

Note: QUERY DASDFW is only valid for cache storage subsystems that support the DASD fast write function. If the specified subsystem does not support DASD fast write, CP generates and error message.

Table 4 on page 99 lists the keywords that appear in responses to QUERY commands for DASD. It also shows:

- The meaning of each keyword
- Whether the keyword indicates an active, free, or offline DASD
- Whether the keyword indicates a dedicated DASD or a DASD in use by CP.

Table 4. DASD Status

Device Status Keyword	Keyword Meaning	Type of Status
ATTACHED TO	The DASD is dedicated to the indicated virtual machine.	Active (dedicated)
CP OWNED	The DASD is used by the system for paging and spooling activity and for temporary disk space and directory space.	Active (CP)
CP SYSTEM	The DASD contains user minidisks.	Active (CP)
FREE	The DASD is available but is not in use.	Free
INT REQUIRED	The DASD requires operator intervention.	Active (CP or dedicated)
OFFLINE	The DASD is unavailable to CP.	Offline
SUBCHOFF	The subchannel for this device is disabled.	Offline Disabled
Volume identifier	The DASD is available but is not in use.	Free

Finding Out a DASD's Hardware Status

Most DASDs have lights that indicate their hardware status, but some do not. To find out the hardware status of a DASD that has no indicator light, enter:

```
query rdev status
```

where *rdev* is the real device number of the DASD. CP's response tells you that the device is operational, not operational, or offline, or that an equipment check or a channel error has occurred for the device.

Changing a DASD's Status

The DASD fast write function can be activated or deactivated for devices in an IBM DASD subsystem. To turn the DASD fast write function on or off, enter:

```
set dasdfw device on rdev
```

where:

device
is optional, is the default, and specifies that the DASD fast write function is to be activated or deactivated for the specified devices.

on
indicates the DASD fast write function is to be activated for the specified device; OFF indicates the DASD fast write function is to be deactivated for the specified device.

forceoff
indicates the DASD fast write function is to be unconditionally deactivated for the specified device, regardless of any internal subsystem errors.

rdev
is the real device number, a list of real device numbers, or a range of real device numbers of devices to be activated or deactivated for the DASD fast write function.

Note: SET DASDFW is only valid for cache storage subsystems that contain nonvolatile storage and support the DASD fast write function. If the specified subsystem does not support this function, CP generates an error message.

ATTENTION: The FORCEOFF option is intended only for recovery purposes. It should be used only when absolutely necessary, because data may be lost.

The FORCEOFF option should be used only after the SET DASDFW OFF command has been used, but the subsystem could not deactivate the DASD fast write function due to internal errors.

Finding Out a DASD's Caching Status

Some DASDs may be attached to storage control units that contain a high-speed storage buffer called a *cache*. Normally, to ensure good performance, you should make sure that these cached DASDs can use the cache. For caching to be available for use, the cache must be available to the subsystem and, if applicable, activated for the device.

To check the caching status of a cached DASD, enter:

```
query cache rdev
```

where *rdev* is the real device number of the DASD. In response, CP displays information about the caching status of the device. The information displayed depends on the storage control model to which the DASD is attached. Table 5 on page 101 shows the keywords that appear when the cache is available to the device.

The cache fast write function improves the performance of access to data residing on DASD in IBM DASD subsystems. It causes I/O operations to appear to complete more quickly by presenting device-end status with channel-end status as soon as the data is received by the control unit. However, this data is *not* saved in nonvolatile storage. Cache fast write should be used only for data that does not need to be saved across a system abend, system re-IPL, or storage subsystem failure; that is, data that does not need to be written to the backing storage device.

The QUERY CACHEFW command allows you to display the current state of the cache fast write function for the subsystem. To display the current state of the cache fast write function, enter:

```
query cachefw rdev
```

where *rdev* is the real device number, a list of real device numbers, or a range of real device numbers of the devices in the subsystems being queried.

Note: QUERY CACHEFW is only valid for cache storage subsystems supporting the cache fast write function. If the specified subsystem does not support cache fast write, CP generates an error message.

Table 5. DASD Caching Status

Storage Control Model	Keywords Displayed when the Cache is Available to the Device
IBM DASD subsystem	CACHE AVAILABLE FOR SUBSYSTEM CACHE ACTIVATED FOR DEVICE CACHE FAST WRITE AVAILABLE FOR SUBSYSTEM CACHE FAST WRITE ACTIVATED FOR DEVICE

Changing a DASD's Caching Status

For performance reasons, a cached DASD should normally have use of the subsystem's cache. For the DASD to have use of the cache, the cache must be both available to the subsystem and, if applicable, activated for the device. The commands you use to change a DASD's caching status depend on the storage control model to which the DASD is attached.

For a particular IBM DASD subsystem, the SET CACHEFW command allows you to activate or deactivate the cache fast write function. To activate cache fast write, enter:

```
set cachefw subsystem on rdev
```

where:

rdev
is the real device number, a list of real device numbers, or a range of real device numbers of the devices in the subsystem.

on
indicates that the cache fast write function is to be activated for the subsystem.

To deactivate cache fast write, enter:

```
set cachefw subsystem off rdev
```

where:

rdev
is the real device number, a list of real device numbers, or a range of real device numbers of the devices in the subsystem.

off
indicates that the cache fast write function is to be deactivated for the subsystem.

Note:

1. SET CACHEFW is only valid for cache storage subsystems supporting the cache fast write function.
2. If the subsystem cache is unavailable when you enter SET CACHEFW ON, cache fast write data access begins when the cache is made available.
3. When you enter SET CACHEFW OFF, the cache fast write data for all devices in the subsystem is destaged to the appropriate DASD.
4. If the subsystem is not in the correct mode to accept the SET CACHEFW ON command, certain error messages may result. For information about these messages, see the *z/VM: CP Messages and Codes*.

To transfer cache fast write data to backing storage (on an addressed device), use the COMMIT command. To do this, enter:

```
commit device rdev
```

where:

device
: is optional, is the default, and emphasizes that only data for the specified devices is to be written to DASD.

rdev
: is the real device number, a list of real device numbers, or a range of real device numbers affected by COMMIT.

Note: COMMIT is only valid for cache storage subsystems supporting the cache fast write function. If the subsystem containing the specified device does not support this function, CP generates an error message.

Table 6 on page 102 shows the commands to use to make the cache available to a DASD.

Table 6. DASD Caching Commands

Storage Control Model	Commands to Make the Cache Available to a DASD

IBM DASD subsystem	`set cache subsystem on rdev`
	`set cache device on rdev`
	`set cachefw subsystem on rdev`

In response, CP makes the cache available to the subsystem and, if applicable, activates the cache for the device. For IBM DASD subsystems, both settings are required for the DASD to have use of the cache.

Your installation may at times wish to discontinue use of the cache by the device and by the subsystem. For example, your installation may wish to measure differences in performance with and without the cache. In these situations, you may use the SET CACHE command to discontinue use of the cache by a device.

For IBM DASD subsystems to discontinue use of the cache by a device, enter:

```
set cache device off rdev
```

where *rdev* is the real device number of the DASD. In response, CP deactivates caching by the device. The availability of the cache to the subsystem is unchanged. The settings for other devices in the subsystem are also unchanged. They take effect when you again make the cache available to the subsystem.

Managing Pinned Data

Data that the IBM DASD subsystem cannot destage from cache or NVS because of hardware failures is pinned data. To identify the data tracks of one or more DASD that are pinned in the subsystem, use the QUERY PINNED command. To do this, enter:

```
query pinned device rdev
```

where:

device
: is the default and displays the pinned tracks for the specified device.

subsystem
: displays the total amount of subsystem cache and NVS containing pinned data.

rdev
: is the real device number, a list of real device numbers, or a range of real device numbers of the devices in the subsystems being queried.

Once the pinned data is identified, you can discard that pinned data and free up the cache or NVS for other data.

ATTENTION: The data is discarded, not transferred to backing storage.

To discard the pinned data, enter:

```
discard pinned all|cachefw rdev
```

where:

all
discards all pinned data for the specified device.

cachefw
discards only pinned cache fast write data for the specified device.

rdev
is the real device number, a list of real device numbers, or a range of real device numbers for the devices that contain data to be discarded.

Note: DISCARD is only valid for cache storage subsystems that support cache fast write, DASD fast write, and dual copy functions. If the specified subsystem does not support these functions, CP generates an error message.

Querying Fenced Components

Fencing is the isolation (by partitioning) of failing components of a DASD subsystem by hardware, licensed internal code, or system software on an IBM DASD subsystem. Once the component is fenced it is no longer available for any I/O activity until it is unfenced.

To display the fenced components, if any exist, enter:

```
query fences rdev
```

This tells you if the subsystem is running in a degraded mode that requires service. To unfence the fenced components use ICKDSF.

Transferring Data to Backing Storage

To terminate the use of the facilities of the IBM DASD subsystems, you should transfer data residing exclusively in cache or NVS to backing storage on a subsystem basis. To transfer all data, enter:

```
destage subsystem rdev
```

where:

subsystem
is optional, is the default, and specifies that the data for the entire storage subsystem be destaged, regardless of which device within the storage subsystem received the command.

rdev
is the real device number, a list of real device numbers, or a range of real device numbers of the devices receiving the destage order.

Note: DESTAGE is valid only for cache storage subsystems supporting cache fast write, DASD fast write, and dual copy functions. If the specified subsystem does not support these functions, CP generates an error message.

Using the Dual Copy Function

Dual copy lets the 3990 Models 3 and 6 Storage Control create a duplicate copy of the data on a volume and store it on a different device in the same subsystem. The two physical devices, the primary device and the secondary device, make up a duplex pair. Both devices must have the same track and cylinder format and the capacity of the secondary device must be equal to or greater than the primary device. Dual copy and DASD fast write can be used at the same time.

I/O operations are directed to the primary device. For write operations, after the subsystem writes the data to the primary device, channel-end and device-end are presented and the storage control later completes the write operation from the cache to the secondary device. The write operation to the secondary device is transparent to the host. For read operations, if the data is not in the cache, the 3990 Models 3 and 6 Storage Control attempts to read it from the primary device. If the read fails on the primary device, then the storage control attempts to read the data from the secondary device of the duplex pair.

Dual copy provides these important advantages:

1. Protects data on a critical volume from a single device failure.
2. Is transparent to the application program.
3. Switches operations automatically to the secondary device if a failure occurs on the primary device. The switching is transparent to the application.
4. Provides for nearly continuous operations by providing a backup copy if loss of access occurs to one of the devices in the duplex pair.
5. Ensures that the second copy is logically identical with the primary copy.

Dual copy has four device states:

Simplex
A "single copy" volume. Dual copy is not active.

Duplex Pending
A temporary state. This state occurs when the subsystem is doing a copy to synchronize the duplex pair. When the copy is complete, the device enters the duplex state. If the copy does not complete successfully, the device enters the suspended duplex state.

When in the duplex pending state only the primary device is functional to the host systems.

Duplex
This state has two devices, the primary and the secondary which contain identical data. Only the primary device is functional to the host systems.

Suspended Duplex
A state caused by one of two events: (1) a host program (either CP or a guest system) requests the state of the duplex pair to be changed to suspended, or (2) the 3990 Models 3 or 6 are unable to synchronize the contents of the two devices, and has stopped duplicate writes.

The host program can put the pair into suspended duplex state when the host error recovery procedure detects a permanent error. The host issues a channel program that causes a state change to suspended duplex.

If one of the two devices fails, the 3990 Models 3 or 6 itself can put the pair into suspended duplex state.

When in the suspended duplex state only the primary device is functional to the host systems.

Any event or command that suspends or reinstates the dual copy operation can cause the physical primary and secondary devices to swap. When using the term *primary device*, keep in mind that the devices might have swapped—the device originally identified as primary may no longer be the same physical device.

Ungrouping Storage Paths

Duplex pairs cannot be created if the paths for the devices that are online to sharing host systems are grouped.

If you attempt to create a duplex pair with devices that have grouped paths, the duplex pair *will not* be created and these messages will be generated:

The person issuing the CP command will receive the following message:

HCPERP296E Status is not as required - *rdev*; duplex pair
could not be established, code = 17.

The system operator will receive the following message:

HCPERP500I DASD *rdev* AN OPERATION WAS TERMINATED
BECAUSE A COMMAND REJECT ERROR OCCURRED

There are two ways to ungroup paths for devices that you wish to use as a duplex pair. You can ungroup the paths by:

- Varying the devices offline from other host systems.

  Before a duplex pair can be created, the devices that are to be duplexed should be varied offline from other systems to ungroup the paths.
- Doing a System Reset from the system console.

  Even though a sharing host may be idle (that is, no operating system currently IPLed), the paths may still be grouped. By doing a System Reset from the system console, *all* paths to that system will become ungrouped.

The CP DUPLEX command can be used to establish, re-establish, and terminate duplex pairs for CP volumes, including minidisk (full-pack) volumes, as well as DASD dedicated to guest operating systems and CMS users.

For example, to establish a duplex pair in which real device 0200 is the primary volume and 0201 is the secondary volume, you would enter:

```
duplex 0200 on 0201
```

By default, DUPLEX copies the entire contents of the primary volume to the secondary volume so that their contents are synchronized. Subsequent changes to device 0200 would automatically be made to device 0201. Use the QUERY DUPLEX command to display the status of duplex pairs:

```
query duplex 0200
```

The QUERY DUPLEX response indicates the duplex status. When the volumes are synchronized and regular disk activity is in progress, the QUERY DUPLEX response is:

```
Duplex pair (0200,0201) - active
```

The QUERY VIRTUAL DUPLEX command can be used to display the duplex status of virtual devices. Suppose, for example, a user's virtual 0301 device was defined as a few cylinders on real device 0200, which we just duplexed. The user could determine the duplex status by entering:

```
query virtual duplex 0301
```

QUERY VIRTUAL DUPLEX responses are similar to those displayed for QUERY DUPLEX.

To end the duplex relationship, enter:

```
duplex 0200 off
```

Support for guest use of dual copy is provided for DASD that are dedicated to a guest and for DASD that are shared among guests as full-pack minidisks. If you want to let the guest control the duplexing of full-pack minidisks, the minidisks must be defined to CP by using the DEVNO operand of the MDISK directory control statement. Minidisks defined in this way are known to CP by their device numbers (DEVNO) rather than by volume serial number (VOLSER). CP cannot use the volume serial number because guests that form and dissolve duplex pairs can create duplicate volume serial numbers. If CP were to use duplicate volume serial numbers, data integrity problems could result. For more about using dual copy with guests, see *z/VM: Virtual Machine Operation*.

See the *z/VM: CP Commands and Utilities Reference* for more about the DUPLEX, QUERY DUPLEX, and QUERY VIRTUAL DUPLEX commands.

Delayed Responses to CP Commands

Delayed responses result when a CP command associated with DASD subsystems causes a channel program to be issued to a device or subsystem. In most instances, these operations complete in a short time and the command response is immediately presented. However, for the IBM DASD subsystems, some of these operations do not complete immediately.

The CP commands that can result in delayed responses are:

- COMMIT
- DESTAGE
- DUPLEX
- SET CACHE ON|OFF SUBSYSTEM
- SET CACHE OFF DEVICE
- SET CACHEFW OFF
- SET DASDFW OFF
- SET NVS ON|OFF.

The responses to these commands are delayed as follows:

- If the command completes immediately, the normal command response is displayed.
- If the command does not complete immediately, the following response is displayed:

  ```
  Command started:  command rdev.
  ```

- If the command completes successfully, the normal command response is displayed.
- If the command completes unsuccessfully because of errors during asynchronous processing, the following response is displayed:

  ```
  Command failed:  command rdev.
  ```

- If the cache storage subsystem does not notify CP when it has completed a command, the normal completion responses are not generated. Command processing stops after CP displays the following response:

  ```
  Command results lost:  command rdev.
  ```

- If the originator of the command is not the system operator, the following response is issued to the system operator when the command completes:

  ```
  Normal command response by userid
  ```

To determine the status of commands entered but not yet completed, enter:

```
query pending commands rdev.
```

The following response is displayed for each pending CP command:

```
Command pending:  command rdev.
```

To determine the status of all commands that are pending for a device, enter:

```
query pending commands allusers rdev.
```

where

```
allusers
```

causes the following response to be displayed for each pending command:

```
Command pending for userid  command rdev.
```

Collecting Information about Subsystem Resources

The remote service access word (RSAW) is used by remote service personnel to gain access to subsystem resources for problem determination and maintenance. As the system operator, you are responsible for obtaining the RSAW for remote service personnel. To generate and display the RSAW for storage clusters in the DASD subsystem, enter:

```
query rsaw cluster 0|1|all rdev
```

where:

cluster 0|1|all
indicates the storage clusters containing the RSAW to be obtained.

rdev
is the real device number of one of the devices in the subsystem.

If CLUSTER ALL is specified, the following response is issued for each storage cluster:

```
rdev rsaw is nnnn for cluster n, serial number sssss.
```

If the support facility is unavailable and CLUSTER ALL is specified, the following response is issued for each storage cluster:

```
rdev subsystem support facility is unavailable.
```

If the remote service is not enabled and CLUSTER ALL is specified, the following response is issued for each storage cluster:

```
rdev remote service is not enabled.
```

The Storage Subsystem Support Facility supports subsystem error recovery, maintenance, diskette drive operations, and the operator panel. The Support Facility generates the RSAW recognizes it to allow remote maintenance attachment.

Commands to Offline Devices

The IBM DASD subsystems report subsystem level changes and asynchronous message availability on any device in the subsystem. A device can be online or offline to CP. You can enter the following commands to both online and offline devices:

- QUERY CACHE
- SET CACHE
- QUERY DASDFW
- SET DASDFW
- DISCARD
- QUERY DUPLEX
- QUERY PINNED
- QUERY RSAW.

Note: For the QUERY CACHE and SET CACHE commands, offline device support only applies to devices attached to the IBM DASD subsystems.

Status Not as Required Messages

The subsystem must be in the appropriate state to process a particular command. If it is not in the appropriate state, the command fails with a command rejection in format 0, message F. If a CP subsystem management command fails, CP displays the appropriate “status not as required” message (0296E). With the addition of IBM DASD subsystems, there are eight new messages and a general message that reports the reason code with the “status not as required” message. This message has several formats:

- `Nonvolatile storage is not available`
- `FORCEOFF is invalid in the current state`
- `Recovery action for pinned data is required`
- `Unlike channel types cannot be grouped`
- `A Diagnostic Control command failed`
- `The control unit had insufficient message buffer space`
- `Pinned tracks are not identifiable because their location and data are in failed nonvolatile storage`
- `Device is undergoing media maintenance`
- `Duplex pair could not be established, code =` *code*.
- `A duplex state error occurred, code =` *code*.
- code=*nn*

For more information about these messages see the *z/VM: CP Messages and Codes*.

Managing State-Change-Pending

If an IBM DASD subsystem receives an I/O request it cannot process for an extended amount of time, it fails the request with a state-change-pending unit check. The IBM DASD subsystems present a state change interrupt to notify the operating system of a change from the not-ready to the ready state or to signal a change to subsystem storage status, duplex DASD status, nonvolatile storage status, DASD fast write status, or cache fast write status.

If CP detects a state-change-pending unit check for a DIAGNOSE I/O request or for a non-high-priority CP I/O request, CP enqueues the request until the pending state change is completed. In all other cases, CP sends the error to the requesting guest and to the system operator.

Virtual state-change interrupts are sent to those guests who received the associated state-change-pending unit check.

Locating a Specific DASD Volume

To locate a particular DASD volume and find out the status of the DASD on which it is mounted, enter:

```
query dasd volid
```

where *volid* is the volume identifier of the DASD you want to locate.

Dedicating a DASD to a z/VM User

If a DASD is offline and you need to dedicate it to a z/VM user, do the following:

1. Make sure the user is authorized to access data on the device. Follow your installation's normal procedures.
2. Enter:

   ```
   vary online rdev
   attach rdev userid vdev
   ```

   where:

   rdev
   is the real device number of the DASD.

   userid
   is the user's user ID

   vdev
   is the virtual device number the user asked you to assign to the DASD.

Note: If *rdev* and *vdev* are the same, you may enter:

```
attach rdev userid
```

For a cached DASD, CP by default assigns the DEVCTL level of subsystem control to the user. This allows the user's virtual machine to issue CCWs that control device resources but not system resources. However, the user may request a different level of subsystem control than the default DEVCTL.

If the user requests the SYSCTL level of subsystem control, follow the procedure above but add the SYSCTL operand to the ATTACH command:

```
attach rdev userid vdev with sysctl
```

In response, CP dedicates the device to the user and lets the user's virtual machine successfully issue CCWs that control both subsystem and device resources.

If the user requests the NOCTL level of subsystem control, follow the procedure above but add the NOCTL operand to the ATTACH command:

```
attach rdev userid vdev with noctl
```

In response, CP dedicates the device to the user and prevents the user's virtual machine from successfully issuing CCWs that control either subsystem or device resources.

For more information on the subsystem control operands of the ATTACH command, see the *z/VM: CP Commands and Utilities Reference*.

Dedicating a DASD to the Production System

If a DASD is offline and is supposed to be dedicated to the production system for its exclusive use, do the following:

1. Enter:

   ```
   vary online rdev
   attach rdev userid vdev
   ```

   where:

   rdev
   is the real device number of the DASD.

   userid
   is the user ID of the production system

   vdev
   is the virtual device number you want to assign to the DASD.

   Note: If *rdev* and *vdev* are the same, you may enter:

   ```
   attach rdev userid
   ```

2. At the production system's system console, enter any commands that the production operating system requires in order to use the device.

For a cached DASD, CP by default assigns the DEVCTL level of subsystem control to a production system's virtual machine. This allows the virtual machine to issue CCWs that control device resources but not subsystem resources. However, you may assign a different level of subsystem control than the default (DEVCTL) to a production system.

To assign the SYSCTL level of subsystem control to a production system, follow the procedure above, but add the SYSCTL operand to the ATTACH command:

```
attach rdev userid vdev with sysctl
```

In response, CP dedicates the device to the production system and lets the virtual machine successfully issue CCWs that control both subsystem and device resources.

To assign the NOCTL level of subsystem control to a production system, follow the procedure above, but add the NOCTL operand to the ATTACH command:

```
attach rdev userid vdev with noctl
```

In response, CP dedicates the device to the production system and prevents the virtual machine from successfully issuing CCWs that control either subsystem or device resources.

For more information on the subsystem control operands of the ATTACH command, see the *z/VM: CP Commands and Utilities Reference*.

Example—Dedicating a DASD to the Production System

DASD 8B0 should be dedicated to the production system, MVS01, as virtual device 8B0. When you enter:

```
query 8b0
```

CP tells you that DASD 8B0 is offline. To dedicate it to the production system, enter:

```
vary online 8b0
attach 8b0 mvs01
```

At the MVS01 system's system console, enter appropriate commands to make sure the volume is usable.

Removing a DASD from a z/VM User

Removing a DASD from a z/VM user is the opposite of dedicating it:

Enter:

```
detach rdev userid
```

where:

rdev
is the real device number of the DASD.

userid
is the user ID of the user.

Now you may vary the DASD offline or dedicate it to another virtual machine. To vary the volume offline, enter:

```
vary offline rdev
```

where *rdev* is the real device number of the DASD. The DASD is now unavailable to z/VM.

Making a DASD unavailable to the Production System

Removing a DASD from the production system is the opposite of dedicating it:

1. At the production system's system console, enter any commands that the production operating system requires to release the device from use.
2. Contact the operator of any other system that shares the volume and make sure the system is not using it. Follow your installation's normal procedures.
3. Enter:

   ```
   detach rdev userid
   ```

   where:

rdev
is the real device number of the DASD.

userid
is the user ID of the production system.

Now you may vary the DASD offline or dedicate it to another virtual machine. To vary the volume offline, enter:

```
vary offline rdev
```

where *rdev* is the real device number of the DASD. The DASD is now unavailable to z/VM.

Example—Removing a DASD from the Production System

To remove DASD 8B0 from the production system, MVS01:

1. At the MVS01 system's system console, enter appropriate commands to make sure the volume is not in use.
2. Make sure the volume is not in use by any other system.
3. Enter:

   ```
   detach 8b0 mvs01
   ```

Attaching a System Volume to the System

CP needs DASD space to perform a number of system functions. Your system administrator or system programmer defines *CP-owned* system volumes to contain space for such functions as:

- Checkpoint/warm start data
- User directory
- Paging
- Spooling
- Temporary disks
- User minidisks.

Your system administrator may also define other system volumes whose status says CP SYSTEM. This second set of system volumes may be used for dedicated DASD or for user minidisk space.

If either type of system volume is offline, you must attach it to the system for CP to use it. To attach it to the system, enter:

```
vary online rdev
attach rdev system volid
```

where:

rdev
is the real device number of the DASD.

volid
is the volume identifier.

The volume is now available to CP to use for system functions.

Note: If the status of the volume says FREE, you need to enter only the ATTACH SYSTEM command.

Detaching a System Volume from the System

Removing a system volume from z/VM is the opposite of attaching it:

1. Enter:

```
detach rdev system
vary offline rdev
```

where *rdev* is the real device number of the DASD.

The DASD is now unavailable to CP for user minidisk space or CP functions.

Displaying a List of CP-Owned Volumes

To display a list of CP-owned DASD volumes, enter:

```
query cpowned
```

In response, CP displays information on all CP-owned volumes and gaps in the volume list. For more information about the QUERY CPOWNED command, see the *z/VM: CP Commands and Utilities Reference*.

To display a list of CP-owned system volumes and the type of areas allocated on each volume, enter:

```
query alloc all
```

In response, CP lists all of the CP-owned volumes that contain temporary disk, directory, paging, or spooling space. For each volume, CP displays the number of cylinders or pages allocated, in use, and available for temporary disk, directory, paging, and spooling space. CP also indicates whether a volume contains the active CP directory.

To display the list of all CP-owned system volumes containing paging space, enter:

```
query alloc page all
```

In response, CP lists all of the CP-owned volumes that contain paging space. For each paging extent on each volume, CP displays the starting cylinder or page and ending cylinder or page, the total number of pages, the number of pages in use, the highest page in use, and the percentage of pages in use.

To display the list of all CP-owned system volumes containing spooling space, enter:

```
query alloc spool all
```

In response, CP lists all of the CP-owned volumes that contain spooling space. For each spooling extent on each volume, CP displays the starting cylinder or page and ending cylinder and page, the total number of pages, the number of pages in use, the highest page in use, and the percentage of pages in use.

To display the list of all CP-owned system volumes containing temporary disk space, enter:

```
query alloc tdisk all
```

In response, CP lists all of the CP-owned volumes that contain temporary disk space. For each temporary disk extent on each volume, CP displays the starting cylinder or page and ending cylinder or page, the total number of cylinders or pages, the number of cylinders or pages in use, the highest cylinder or page in use, and the percentage of cylinders or pages in use.

To display the list of all CP-owned system volumes containing directory space, enter:

```
query alloc drct all
```

In response, CP lists all of the CP-owned volumes that contain directory space. For each extent on each volume, CP displays the starting cylinder or page and ending cylinder or page, the total number of cylinders and pages, the number of cylinders and pages in use, the highest cylinder or page in use, and the percentage of cylinders or pages in use. The response from this command also indicates which volume contains the active directory.

To display the allocation mapping of all CP-owned system volumes, enter:

```
query alloc map all
```

In response, CP lists all of the CP-owned volumes that contain temporary disk, directory space, paging, or spooling space. For each extent on each volume, CP displays the starting cylinder or page and ending cylinder or page, the total number of cylinders or pages, the number of cylinders or pages in use, the highest cylinder or page in use, the percentage of cylinders or pages in use, and the allocation type of the extent.

Checking System Volume Use

To find out how many users are using system volumes, enter:

```
query system all
```

CP's response shows the number of users either linked to minidisks or using temporary disk space on each CP-owned and CP SYSTEM system volume. CP also details the kind of use for each user.

Checking the Allocation and Format of a System Volume

Before you attach a DASD volume to z/VM for system use, make sure that it is allocated and formatted correctly. To find out whether it is allocated correctly, you may use the CPFMTXA utility or the Device Support Facilities program (ICKDSF). However, using ICKDSF is the recommended method to perform all maintenance on DASD volumes for CP use. For more information on using ICKDSF to format, label, or allocate DASD volumes, see the *Device Support Facilities User's Guide and Reference* book. For details on the CPFMTXA utility, see the *z/VM: CP Commands and Utilities Reference*.

To find out whether a system volume is allocated correctly using the CPFMTXA utility, do the following:

1. Make the device available to the system and dedicate it to your virtual machine. Enter:

   ```
   vary online rdev
   attach rdev to * as vdev
   ```

   Where:

   rdev
   is the real device number of the DASD.

   vdev
   is the virtual device number you want to assign to the device.

   **** (asterisk)***
   indicates that you want CP to dedicate the device to your own virtual machine.

2. To find out whether the volume is already allocated for use with z/VM, enter:

   ```
   cpfmtxa vdev volid allocate
   ```

   where:

   vdev
   is the virtual device number you assigned to the DASD.

   volid
   is the volume identifier of the volume that is mounted on this device.

   When using CPFMTXA, ICKDSF is invoked and you will receive ICKDSF messages. Normally, ICKDSF first displays the current allocation of the volume. Under the headings of TYPE, ICKDSF displays the type of space and its cylinder or page range, respectively. ICKDSF may display under TYPE any of the following types of space:

   - Directory (DRCT)
   - Paging (PAGE)
   - User minidisk, checkpoint, or warm start (PERM)
   - PARM space
   - Spooling, dumps, system data files, or paging overflow (SPOL)

- Temporary disk (TDSK)
- Invalid allocation map (BAD).

CPFMTXA then prompts you to enter new allocation data.

3. Enter:

```
end
```

In response, CPFMTXA again displays the current allocation (unchanged), it tells you that the allocate operation and the CPFMTXA utility are complete.

If no errors occur and the volume contains no invalid allocation map (BAD) cylinders, the volume is in z/VM format.

If I/O errors are reported, error messages are issued, or the volume contains invalid allocation map (BAD), see "Formatting and Allocating a System Volume" on page 114 before contacting support personnel for help in formatting and allocating the volume.

Formatting and Allocating a System Volume

If a z/VM system volume has the wrong format or has not been formatted, your installation can format it for system use. Your support personnel must provide you with the appropriate input.

Before you format and allocate a system volume, you may wish to use the Device Support Facilities program (ICKDSF) to inspect the DASD volume for defective tracks and to assign alternate tracks if required. For more information on using this program, see "Running the Device Support Facilities Program" on page 245, or refer to the *Device Support Facilities User's Guide and Reference* book.

Note: Although in most instances the CPFMTXA utility can be used for such maintenance procedures, it is recommended that instead, the Device Support Facilities program (ICKDSF) be used in all CP volume maintenance. This is because some DASD devices require the use of the ICKDSF program. In fact, when CPFMTXA is invoked, z/VM will call the ICKDSF program. Only if ICKDSF is available will the system continue with the original command.

Use the CPFMTXA utility or invoke ICKDSF directly to format DASD. For convenience, this section illustrates the use of the CPFMTXA utility. The responses and messages may vary slightly depending on the given scenario.

For more details on the CPFMTXA utility, see the *z/VM: CP Commands and Utilities Reference*.

To format and allocate a system volume using the CPFMTXA utility:

1. Vary the device online and attach it to your virtual machine. Enter:

```
vary online rdev
attach rdev to * as vdev
```

where:

rdev
is the real device number of the DASD.

vdev
is the virtual device number you want to assign to the device.

**** (asterisk)***
indicates that you want CP to dedicate the device to your own virtual machine.

2. Run CPFMTXA. Enter:

```
cpfmtxa vdev volid
```

where:

vdev
is the virtual device number you assigned to the DASD.

volid
is the volume identifier you want to assign to the volume that is mounted on this device.

CPFMTXA tells you that it will erase the cylinders or pages on the disk and asks you if you want to continue. If you answer NO, it does not format the disk. If you answer YES to the continuation prompt, ICKDSF formats the volume.

Note: If ICKDSF is not on an accessed disk, formatting does not take place.

3. Enter the input your support personnel supplies.

   For example, if cylinders 0 through 200 are to be used as spooling space, cylinders 201 through 400 as paging space, and the remaining cylinders as temporary disk space, enter:

   ```
   spol 0-200
   page 201-400
   tdsk 401-end
   end
   ```

   For FBA type DASD page expressions rather than cylinders would be used in the above example.

4. After the volume is allocated, enter:

   ```
   detach rdev from userid
   attach rdev to system as volid
   ```

   where:

   rdev
   is the real device number of this DASD.

   userid
   is your user ID.

   volid
   is the volume identifier of the volume that is mounted on this device.

   These commands remove the volume from your virtual machine and attach it to the system.

Note: If you are adding the volume for checkpoint and warm-start data, paging, spooling, dump, directory, or temporary disk space, your installation must have defined the volume identifier of the device by specifying the CP_OWNED statement in the system configuration file. If your installation did not define the volume identifier, CP has no reference to the volume and does not use it for checkpoint and warm-start data, paging, spooling, dump, directory, or temporary disk space.

To find out whether CP is using the volume for these CP functions, enter:

```
query system rdev
```

where *rdev* is the real device number of the DASD you just attached to the system. If CP's response includes the words “ATTACHED CPVOL,” CP is using the volume. If the response includes the words “ATTACHED SYSTEM,” however, CP is not using the volume for the CP functions listed earlier.

Changing the Volume Identifier of a System Volume

The name of a system volume resides in a 6-character field called the *volume identifier*. The volume identifier resides in a longer record called the volume label. If you need to change the volume identifier of a system volume, you may use the CPFMTXA utility or the Device Support Facilities program (ICKDSF). However, using ICKDSF is the recommended method for all maintenance on DASD volumes for CP use. For more information on using ICKDSF for CP maintenance procedures, see the *Device Support Facilities User's Guide and Reference* book. For more details on the CPFMTXA utility, see the *z/VM: CP Commands and Utilities Reference*.

For convenience, this section shows the use of the CPFMTXA utility. The responses and messages may vary slightly depending on the given scenario.

To change the volume identifier of a system volume using the CPFMTXA utility, do the following:

1. Make the device available to the system and dedicate it to your virtual machine. Enter:

   ```
   vary online rdev
   attach rdev to * as vdev
   ```

   where:

   rdev
   is the real device number of the DASD.

   vdev
   is the virtual device number you want to assign to the device.

   **** (asterisk)***
   indicates that you want CP to dedicate the device to your own virtual machine.

2. To change the volume identifier, enter:

   ```
   cpfmtxa vdev volid label
   ```

   where:

   vdev
   is the virtual device number you assigned to the DASD.

   volid
   is the new volume identifier you want to assign to the volume.

   In response to this command, ICKDSF changes the volume identifier of the system volume to the new one you specified.

Changing the Label of a DASD Volume for the Production System

If you need to change the label of a production system's DASD volume, use the Device Support Facilities (ICKDSF) program. For more information, see "Running the Device Support Facilities Program" on page 245 or refer to the *Device Support Facilities User's Guide and Reference* book.

Defining a DASD as a Dump Device

To prepare the system for a possible failure, you may need to define a DASD as a dump device. To find out how to do this, see "Preparing the System for a Possible Failure" on page 45.

Inspecting the Physical Condition of a DASD Volume

If you need to inspect a DASD volume for defective tracks, use the device support facilities program. For more information, see "Running the Device Support Facilities Program" on page 245.

Copying One DASD Volume to Another

If you need to copy one DASD volume to another, use the DASD dump restore utility that IBM provides with z/VM. For more information, see "Running the DASD Dump Restore Utility (DDRXA)" on page 238.

Tape Drives

Tapes contain much of your installation's data. Accordingly, you must spend a lot of your time controlling tape drives.

Some of the tape drives in your z/VM system may be shared by operating systems running on other processor complexes. Sharing tape drives in this way affects the way you control them. Accordingly, the next section focuses on shared tape-drive operation. "Controlling Tape Drives" on page 117 follows with the main tape-drive operation tasks.

Operating with Shared Tape Drives

The only tape drives z/VM allows you to share with other systems are IBM 3480 and follow on Magnetic Tape Subsystems. z/VM does not support the sharing of other tape drives.

z/VM allows your installation to share 3480s or follow on tape drives safely through assign/unassign support. Assign/unassign support for tape drives is similar in purpose to reserve/release support for DASDs; it guarantees data integrity while still allowing several operating systems to use the same device. Assign/unassign allows only one operating system at a time to access a tape drive. This prevents one system from reading data on a tape that another system is changing at the same time.

Assign/unassign support makes use of a hardware function available only on 3480 and follow on tape drives. To serialize access to 3480 and follow on tape drives, operating systems use particular channel command words (CCWs) called assign/unassign CCWs. An assign CCW exclusively reserves a tape drive for the issuer. An unassign CCW releases the drive, allowing another operating system to reserve it. Assign/unassign CCWs replace the physical partitioning switches that are present on IBM tape subsystems of earlier design.

When you dedicate a 3480 or follow on tape drive to a z/VM virtual machine, you assign the tape drive to the virtual machine by default. You may also specify the NOASSIGN option to dedicate the tape drive without trying to assign it to the virtual machine.

The assignment of the tape drive can also be established when the drive is made available to the system (vary online). By specifying the ASSIGN option on the VARY ONLINE command, CP will assign the drive to the system when the device is brought online. When the drive is eventually dedicated (attached) to a virtual machine, no attempt will be made to assign the drive. When the drive is detached from the virtual machine it will not be unassigned. This method of tape management reduces the chance of the drive becoming "assigned elsewhere" when being reattached to different virtual machines. When the ASSIGN option is specified on the VARY ONLINE command and the drive is found to be "assigned elsewhere", the drive will not be brought online. Simply issuing the VARY ONLINE command without the ASSIGN option (the default), will bring the device online. You can still specify the NOASSIGN option to the dedicate, which will cause an UNASSIGN to be issued to the drive. Once a dedicate with the NOASSIGN option has been specified, the assign that was acquired at vary online is no longer in effect for this drive. Future dedicates will now assign the drive upon attachment to the virtual machine (the default).

If you must release a drive that CP has assigned for a virtual machine or the virtual machine has left unassigned, you may use the RESET ASSIGN command.

ATTENTION: The RESET ASSIGN command is intended for use in the removal of spurious assignments from a real device. You should use this command **only** when you know that the virtual machine user of the device, if any, is hung or has otherwise failed, and is thus unable to complete the work being done on the device. Use of this command on a device being actively used can allow the device to be accessed by another real system with shared access to the device, potentially destroying the active work.

Controlling Tape Drives

To control tape drives, you need to be able to:

- Find out a tape drive's status
- Make a tape drive available to the system
- Dedicate a tape drive to a z/VM user or to the production system
- Remove a tape drive from a z/VM user or from the production system
- Remove a tape drive from the system
- Mount a tape for a z/VM user or for the production system
- Dump data on DASD to tape, restore data to DASD from tape, and copy one tape to another
- Dump or load spool files or system data files

Finding Out a Tape Drive's Status

To find out the status of a specific tape drive, enter:

```
query rdev
```

where *rdev* is the real device number of the tape drive.

To find out the status of all the tape drives in your system, enter:

```
query tapes all
```

Table 7 on page 118 lists the keywords that appear in responses to QUERY commands for tape drives. It also shows:

- The meaning of each keyword
- Whether the keyword indicates an active, free, or offline tape drive
- Whether the keyword indicates a dedicated tape drive or a tape drive in use by CP.

Table 7. Tape Drive Status

Device Status Keyword	Keyword Meaning	Type of Status
ASSIGNED SYSTEM	The tape drive is processing a dump, an SPXTAPE function, or a TRSAVE function.	Active (CP)
ATTACHED TO	The tape drive is dedicated to the indicated virtual machine.	Active (dedicated)
FREE	The tape drive is available but is not in use.	Free
INT REQUIRED	The tape drive requires operator intervention.	Active (CP or dedicated)
OFFLINE	The tape drive is unavailable to CP.	Offline
SUBCHOFF	The subchannel for this device is disabled.	Offline Disabled

Making a Tape Drive Available to the System

If the status of a tape drive is offline, you must make it available to the system in order to use it. To make it available to the system, enter:

```
vary online rdev
```

or, to make it available and assigned to the system, enter:

```
vary online rdev assign
```

where *rdev* is the real device number of the tape drive.

You may now dedicate it to a z/VM user or to the production system.

You may get an error message that the tape drive was not varied online because it was assigned elsewhere. You can still bring the device online by reissuing the command without the ASSIGN option (the default).

Dedicating a Tape Drive to a z/VM User

If a tape drive is offline and you need to dedicate it to a z/VM user, enter:

```
vary online rdev
attach rdev userid vdev
```

where:

rdev
is the real device number of the tape drive.

userid
is the user's user ID.

vdev
is the virtual device number that the user asked you to assign to the tape drive.

If you want the drive assigned to the system during VARY ONLINE, enter:

```
vary online rdev assign
```

You may get an error message that the tape drive was not varied online because it was assigned elsewhere. You can still bring the device online by reissuing the command without the ASSIGN option (the default).

If the tape drive is a 3480 or follow on, you may get an error message that the tape drive was not attached because it was assigned elsewhere. If the virtual machine is running an operating system that handles assign/unassign support, you can dedicate the tape drive by entering:

```
attach rdev userid vdev noassign
```

In response, CP dedicates the tape drive without trying to assign it to the virtual machine.

Note: CMS does not support use of a shared virtual tape device. Refer to the *z/VM: CMS User's Guide*. If the tape drive you are trying to dedicate to a CMS virtual machine is assigned elsewhere, find another free tape drive and dedicate it without the NOASSIGN option.

Dedicating a Tape Drive to the Production System

If a tape drive is offline and you want to dedicate it to the production system for its exclusive use, do the following:

1. Enter:

   ```
   vary online rdev
   attach rdev userid vdev
   ```

   where:

   rdev
   is the real device number of the tape drive.

   userid
   is the user ID of the production system.

   vdev
   is the virtual device number you want to assign to the tape drive.

   Note: If *rdev* and *vdev* are the same, you can enter:

   ```
   attach rdev userid
   ```

   If you want the drive assigned to the system during VARY ONLINE, enter:

   ```
   vary online rdev assign
   ```

You may get an error message that the tape drive was not varied online because it was assigned elsewhere. You can still bring the device online by reissuing the command without the ASSIGN option (the default). If the tape drive is a 3480 or follow on, you may get an error message that the tape drive was not attached because it was assigned elsewhere. In this case, dedicate the tape drive by entering:

```
attach rdev userid vdev noassign
```

If the tape drive you are trying to dedicate is assigned to a system that does not support the assign/unassign CCWs, find another free tape drive and dedicate it without the NOASSIGN option.

2. At the production system's system console, enter any commands that the production operating system requires to use the device.

Example—Dedicating a Tape Drive to the Production System

Tape drive 5C0 should be dedicated to the production system, z/VM, as virtual device 5C0. When you enter:

```
query 5c0
```

CP tells you that tape drive 5C0 is offline. For example, to dedicate it to the production system, enter:

```
vary online 5c0
attach 5c0 vmsp
```

At the z/VM system console, enter appropriate z/VM commands to make sure the tape drive is usable.

Transferring Control of a Tape Drive

You can transfer ownership of dedicated tape drives among users running on the same system by using the GIVE command. You are most likely to use the GIVE command when the service virtual machine is considered the owner of the tape drive and other virtual machines want to borrow that tape drive.

Note: Through the GIVE command, a privileged user can also transfer control of a tape device from a second user to a third.

The system operator, as a privileged user, obtains dedicated use of a tape drive by entering the ATTACH command. Once you have obtained the tape drive, you can transfer ownership of that tape drive by entering the GIVE command. (You can also transfer the tape drive from a program by using the CP DIAGNOSE X'08' interface.) To reclaim ownership of the tape drive once the receiver is finished with it, you specify the RETURN option with the GIVE command. If you give the tape drive with the RETURN option, you should also specify whether the tape should be rewound and unloaded or left in its current position. You do this with the UNLOAD and LEAVE options, respectively. You can also specify whether the receiver is to have read/write or read/only access to the given tape drive.

The receiver logically owns the tape drive until it is detached with the DETACH command or until the receiver logs off the system. If the virtual machine that has received the tape drive has class B authorization, and the tape drive was given without the RETURN option, the receiver can give the tape drive to a third party. If the RETURN option was specified, the second GIVE is rejected, and CP issues a message.

Note: In an installation with more than one tape server virtual machine, one of the servers may omit the RETURN option. In this situation, one server machine can give a tape drive to another without the RETURN option; the second server machine can then become the owner of the tape drive and can give it to a virtual machine that requests a tape drive.

To cancel a return request for the tape drive, you (as the giver) can use the DETACH command, specifying the reserved virtual device number, or you can log off your virtual machine. Even if you log off and log on again before the receiver detaches the tape drive, the device is not returned to you. When the receiver enters the DETACH command, the tape drive is returned either to the giver or to the system, depending on the current RETURN status of the tape drive.

When a class B user enters the DETACH command to remove a tape drive, the LEAVE or UNLOAD option specified on DETACH overrides any option specified on the GIVE command.

Example 1—Transferring a Tape Drive with Read/Write Access

The receiver wants to receive control of a tape drive with read/write access. You (the giver) want the tape returned in its current position. To transfer the tape you can take the following steps:

1. You log on the service virtual machine and verify that all tape drives are dedicated to it.
2. The receiver logs on and requests a tape.
3. You mount the tape and read and verify the label.
4. You attach the tape to the receiver by entering:

   ```
   give vaddr1 to receiver as vaddr2 return leave
   ```

   where:

   vaddr1
   is the virtual address of the tape to be transferred.

   vaddr2
   is the virtual address of to the target virtual machine.

   The tape is returned to you and left in the current position. By specifying RETURN and LEAVE, you verify the integrity of the label.
5. The receiver accesses the tape (and reads and writes).
6. When finished with the tape, the receiver detaches the tape by entering:

   ```
   detach virtual vdev
   ```

   where:

   virtual
   indicates a virtual device is to be detached.

   vdev
   is a virtual device number, a list of virtual device numbers, or a range of virtual device numbers to be detached from the virtual machine.
7. Control of the tape drive returns to you with the tape left in its current position.
8. You rewind the tape and read the label to make certain the receiver did not write over the label.
9. Unload the tape.

Example 2—Transferring a Tape Drive with Read Only Access

The receiver wants control of a tape drive. For the particular application, read only access is necessary. You want the tape returned and rewound. To transfer the tape you may do the following:

1. You log on the service virtual machine and verify that all tape drives are dedicated to it.
2. The receiver logs on and requests a tape.
3. You mount the tape and read and verify the label.
4. You attach the tape to the receiver by entering:

   ```
   give vaddr1 to receiver as vaddr2 return unload r/o
   ```

   where:

   vaddr1
   is the virtual address of the tape to be transferred.

   vaddr2
   is the virtual address of the target virtual machine.

   R/O attaches the tape to the receiver in read only mode; any attempts to write to the tape drive result in an error.

5. The receiver accesses the tape (and reads from it).
6. The receiver, when finished with the tape, detaches the tape by entering:

   ```
   detach virtual vdev
   ```

   where:

   virtual
   indicates a virtual device is to be detached.

   vdev
   is a virtual device number, a list of virtual device numbers, or a range of virtual device numbers to be detached from the virtual machine.
7. Control of the tape drive returns to you with the tape unloaded and rewound.

Removing a Tape Drive from a z/VM User

Usually z/VM users remove tape drives from their virtual machines by entering the DETACH (Virtual Device) command. However, if you need to remove a tape drive from a z/VM user, enter:

```
detach rdev userid
```

where:

rdev
is the real device number of the tape drive.

userid
is the user ID of the user.

Note: If the 3480 or follow on drive was varied online with the ASSIGN option, CP will not unassign the drive during the detach process.

If a 3480 or follow on tape drive was attached with the NOASSIGN option, CP does not rewind and unload the tape; it leaves the tape positioned on the tape drive. All other tape drives rewind and unload the tape.

If you want to be sure the tape remains positioned on the tape drive, enter:

```
detach rdev userid leave
```

where:

rdev
is the real device number of the tape drive.

userid
is the user ID of the user.

If you want to be sure the tape rewinds and unloads from the tape drive, enter:

```
detach rdev userid unload
```

where:

rdev
is the real device number of the tape drive.

userid
is the user ID of the user.

Now you may remove the tape drive from the system or dedicate it to another virtual machine.

Removing a Tape Drive from the Production System

Removing a tape drive from the production system is the opposite of dedicating it:

1. At the production system's system console, enter any commands that the production operating system requires in order to release the device from use.
2. Enter:

   ```
   detach rdev userid
   ```

   where:

   rdev
   is the real device number of the tape drive.

   userid
   is the user ID of the production system.

   Note: If the 3480 or follow on drive was varied online with the ASSIGN option, CP will not unassign the drive during the detach process.

   If a 3480 or follow on tape drive was attached with the NOASSIGN option, CP does not rewind and unload the tape; it leaves the tape positioned on the tape drive. All other tape drives rewind and unload the tape.

   If you want to be sure the tape remains positioned on the tape drive, enter:

   ```
   detach rdev userid leave
   ```

   where:

   rdev
   is the real device number of the tape drive.

   userid
   is the user ID of the production system.

   If you want to be sure the tape rewinds and unloads from the tape drive, enter:

   ```
   detach rdev userid unload
   ```

   where:

   rdev
   is the real device number of the tape drive.

   userid
   is the user ID of the production system.

Now you may remove the tape drive from the system or dedicate it to another virtual machine.

Removing a Tape Drive from the System

Removing a tape drive from the system is the opposite of making it available. To remove it from the system:

1. Make sure the tape drive is not in use. Enter a QUERY command for the device to find out if it is.

   If it is in use, take appropriate steps to remove it from the virtual machine. The status should say FREE before you go to the next step.
2. Enter:

   ```
   vary offline rdev
   ```

   where *rdev* is the real device number of the tape drive.

   The tape drive is now unavailable to z/VM.

Note: As part of vary offline processing, CP will ensure that the drive is unassigned at command completion.

Mounting a Tape

If you need to mount a tape for a z/VM user, a production system user, or a job, do the following:

1. Locate the tape.
2. Make sure the user or job is authorized to access data on the tape. Follow your installation's normal procedures.
3. Follow the procedures in “Finding Out a Tape Drive's Status” on page 118 and “Dedicating a Tape Drive to a z/VM User” on page 119 or “Dedicating a Tape Drive to the Production System” on page 119 to check the status of the tape drive and dedicate it to the user or production system.
4. Mount the tape (enabled for writing, or not, as requested).

   The tape is now available to the user.
5. At the production system's system console, enter any commands that the production operating system requires in order to use the device.

   The tape drive is now available to the production system user or job.

Dumping Data on DASD to Tape

If you need to dump all or part of the data on DASD to a tape, use the DASD Dump Restore program that IBM provides with z/VM. For more information, see “Running the DASD Dump Restore Utility (DDRXA)” on page 238.

If you want to dump one or more CMS files to tape, use the CMS TAPE DUMP command.

For example, if you want to dump to tape all of the files on your A disk with a file type of ACCOUNT and the tape drive that is dedicated to your virtual machine is assigned virtual device number 184, enter:

```
tape dump * account a (tap4
```

The asterisk indicates that you want to dump all of the files with the specified file type. If you want to dump only a specific file, specify the file's name instead of the asterisk.

Note: The records in these CMS files can be either fixed or of variable length. However, none of the files can be larger than one tape volume.

If you want to dump a file that contains a loadable (through IPL) version of a program to tape, use the CMS FILEDEF and MOVEFILE commands.

Example—Dumping Data on DASD to Tape

Suppose you want to dump to tape an IPLable version of the DDR program. The program resides on the K disk with a file name and file type of IPL DDR. To put this file on tape, do the following:

1. Ask the operator to:
   - Mount on a tape drive a scratch tape enabled for writing
   - Attach the tape drive to your virtual machine as virtual device number 181.
2. Define for the MOVEFILE command the file you want to move to tape. Enter:

   ```
   filedef in disk ipl ddr k
   ```
3. Define for the MOVEFILE command the tape drive to which you want to move the file. Enter:

   ```
   filedef out tap1 (recfm f lrecl 80 block 80
   ```
4. Wait for CP to send you a message that the tape drive has been attached.
5. Move the DDR file from disk to tape. Enter:

   ```
   movefile in out
   ```

Restoring Data to DASD from Tape

If you need to restore data to DASD from a tape, and you used the DASD dump restore utility to create the tape, use the same program to restore it. For more information, see "Running the DASD Dump Restore Utility (DDRXA)" on page 238.

If you want to load one or more CMS files from a tape onto a CMS minidisk, and you use the CMS TAPE DUMP command to dump them, use the CMS TAPE LOAD command to load them.

For example, if you want to load onto your K disk some CMS files that were previously dumped to tape, and the tape drive on which the tape is mounted is assigned the virtual device number 183, enter:

```
tape load * * k (tap3
```

The asterisks indicate that you want CMS to load all the files. If you want CMS to load only a specific file, specify the file's name and type instead of the asterisks.

Copying One Tape to Another

If you need to copy data from one tape to another, use the DASD dump restore utility that IBM provides with z/VM. For more information, see "Running the DASD Dump Restore Utility (DDRXA)" on page 238.

Dumping Spool Files or System Data Files to Tape

If you need to dump spool files or system data files to tape, use the SPXTAPE command. For more information, see "Backing Up Standard Spool Files and System Data Files" on page 214.

Loading Spool Files or System Data Files from Tape

If you need to load spool files or system data files from tape back on to the system's spooling space, use the SPXTAPE command. For more information, see "Backing Up Standard Spool Files and System Data Files" on page 214.

Channel-to-Channel Adapters (CTCAs)

To control channel-to-channel adapters (CTCAs), you need to be able to:

- Find out a CTCA's status
- Make a CTCA available to the system
- Dedicate a CTCA to a z/VM user or to the production system
- Remove a CTCA from a z/VM user or from the production system
- Make a CTCA unavailable to the system.

But before you start, you should know how your installation uses CTCAs to transmit data from one system to another.

When You Share Data with Another System through CTCAs

Many installations run more than one system on more than one processor complex. In these installations, the systems can be loosely coupled through channel-to-channel adapters (CTCAs). If this is true at your installation, you should make sure you know how to shut down data transmission safely before you remove CTCAs from z/VM.

If your installation uses CTCAs to connect z/VM or the production system to a network, you must be careful when you switch a CTCA from one virtual machine to another. You must also be careful when you remove the CTCA from z/VM's control (when you vary it offline) or when you shut down the system.

Normally, being careful means you must make sure that other systems have stopped sending data over the CTCA. Before you detach a CTCA or vary it offline, or enter the SHUTDOWN command, contact the operators of the other systems and follow your installation's normal procedures.

Finding Out a CTCA's Status

To find out the status of a specific CTCA, enter:

```
query rdev
```

where *rdev* is the real device number of the CTCA.

To find out the status of all the CTCAs in your z/VM system, enter:

```
query ctca all
```

Table 8 on page 126 lists the keywords that appear in responses to QUERY commands for CTCAs. It also shows:

- The meaning of each keyword
- Whether the keyword indicates an active, free, or offline CTCA
- Whether the keyword indicates a dedicated CTCA or a CTCA in use by CP.

Table 8. CTCA Status

Device Status Keyword	Keyword Meaning	Type of Status
ATTACHED TO	The CTCA is dedicated to the indicated virtual machine.	Active (dedicated)
FREE	The CTCA is available but is not in use.	Free
INT REQUIRED	The CTCA requires operator intervention.	Active (CP or dedicated)
OFFLINE	The CTCA is unavailable to CP.	Offline
SUBCHOFF	The subchannel for this device is disabled.	Offline Disabled

Making a CTCA Available to the System

If the status of a CTCA is offline, you must make it available to the system in order to use it. To make it available to the system, enter:

```
vary online rdev
```

where *rdev* is the real device number of the CTCA.

Now you may dedicate it to a z/VM user or to the production system.

Dedicating a CTCA to a z/VM User

If a CTCA is offline and you need to dedicate it to a z/VM user, enter:

```
vary online rdev
attach rdev userid vdev
```

where:

rdev
is the real device number of the CTCA.

userid
is the user's user ID.

vdev
is the virtual device number that the user asked you to assign to the CTCA.

Dedicating a CTCA to the Production System

If a CTCA is offline and you want to dedicate it to the production system for its exclusive use, do the following:

1. Enter:

   ```
   vary online rdev
   attach rdev userid vdev
   ```

   where:

   rdev
   is the real device number of the CTCA.

   userid
   is the user ID of the production system.

   vdev
   is the virtual device number that you want to assign to the CTCA.

   Note: If *rdev* and *vdev* are the same, you may enter:

   ```
   attach rdev userid
   ```

2. At the production system's system console, enter any commands that the production operating system requires to use the device.

Example: Dedicating a CTCA to the Production System

CTCA 620 should be dedicated to the production system, MVSESA, as virtual device 620. When you enter:

```
query 620
```

CP tells you that CTCA 620 is offline. To dedicate it to the production system, enter:

```
vary online 620
attach 620 mvsesa
```

At the MVSESA system's system console, enter appropriate operating system commands to make sure the CTCA is usable.

Removing a CTCA from a z/VM User

A z/VM user usually removes a dedicated CTCA from the user's virtual machine by entering a DETACH VIRTUAL command. However, if you need to remove a CTCA from a z/VM user, enter:

```
detach rdev userid
```

where:

rdev
is the real device number of the CTCA.

userid
is the user's user ID.

Now you may make the CTCA unavailable to the system or dedicate it to another virtual machine.

Removing a CTCA from the Production System

Removing a CTCA from the production system is the opposite of dedicating it:

1. Make sure that any systems that are connected through the CTCA have stopped sending data. Follow your installation's normal procedures.
2. At the production system's system console, enter any commands that the production operating system requires to release the device from use.
3. Enter:

   ```
   detach rdev userid
   ```

   where:

   rdev
   is the real device number of the CTCA.

   userid
   is the user ID of the production system.

Now you may make the CTCA unavailable to the system or dedicate it to another virtual machine.

Removing a CTCA from the System

Making the CTCA unavailable to the system is the opposite of making it available. To make the CTCA unavailable to the system:

1. Make sure the CTCA is not in use. Enter a QUERY command for the device to find out if it is.

   If it is in use, take appropriate steps to remove it from the virtual machine. The status should say SYSTEM before you go to the next step.
2. Enter:

   ```
   vary offline rdev
   ```

   where *rdev* is the real device number of the CTCA.

   The CTCA is now unavailable to z/VM.

Open Systems Adapters (OSAs)

To control open systems adapters (OSAs), you need to be able to:

- Find out an OSA's status
- Make an OSA available to the system
- Dedicate an OSA to a z/VM user or to the production system
- Remove an OSA from a z/VM user or from the production system
- Make an OSA unavailable to the system.

Finding Out an OSA's Status

To find out the status of a specific OSA, enter:

```
query rdev
```

where *rdev* is the real device number of the OSA.

To find out the status of all the OSAs in your z/VM system, enter:

```
query osa all
```

Table 9 on page 129 lists the keywords that appear in responses to QUERY commands for OSAs. It also shows:

- The meaning of each keyword
- Whether the keyword indicates an active (dedicated), free, or offline OSA.

Table 9. OSA Status

Device Status Keyword	Keyword Meaning	Type of Status
ATTACHED TO	The OSA is dedicated to the indicated virtual machine.	Active (dedicated)
FREE	The OSA is available but is not in use.	Free
OFFLINE	The OSA is unavailable to CP.	Offline
SUBCHOFF	The subchannel for this device is disabled.	Offline Disabled

Making an OSA Available to the System

If the status of an OSA is offline, you must make it available to the system in order to use it. To make it available to the system, enter:

```
vary online rdev
```

where *rdev* is the real device number of the OSA.

Now you may dedicate it to a z/VM user or to the production system.

Dedicating an OSA to a z/VM User

If an OSA is offline and you need to dedicate it to a z/VM user, enter:

```
vary online rdev
attach rdev userid vdev
```

where:

rdev
is the real device number of the OSA.

userid
is the user's user ID.

vdev
is the virtual device number that the user asked you to assign to the OSA.

Dedicating an OSA to the Production System

If an OSA is offline and you want to dedicate it to the production system for its exclusive use, do the following:

1. Enter:

   ```
   vary online rdev
   attach rdev userid vdev
   ```

   where:

   rdev
   is the real device number of the OSA.

   userid
   is the user ID of the production system.

   vdev
   is the virtual device number that you want to assign to the OSA.

Note: If *rdev* and *vdev* are the same, you may enter:

```
attach rdev userid
```

2. At the production system's system console, enter any commands that the production operating system requires to use the device.

Example—Dedicating an OSA to the Production System

OSA 620 should be dedicated to the production system, MVSESA, as virtual device 620. When you enter:

```
query 620
```

CP tells you that OSA 620 is offline. To dedicate it to the production system, enter:

```
vary online 620
attach 620 mvsesa
```

At the MVSESA system's system console, enter appropriate z/OS commands to make sure the OSA is usable.

Removing an OSA from a z/VM User

A z/VM user usually removes a dedicated OSA from the user's virtual machine by entering a DETACH VIRTUAL command. However, if you need to remove an OSA from a z/VM user, enter:

```
detach rdev userid
```

where:

rdev
is the real device number of the OSA.

userid
is the user's user ID.

Now you may remove the OSA from the system or dedicate it to another virtual machine.

Removing an OSA from the Production System

Removing an OSA from the production system is the opposite of dedicating it:

1. At the production system's system console, enter any commands that the production operating system requires to release the device from use.
2. Enter:

   ```
   detach rdev userid
   ```

   where:

   rdev
   is the real device number of the OSA.

   userid
   is the user ID of the production system.

Now you may remove the OSA from the system or dedicate it to another virtual machine.

Removing an OSA from the System

Removing an OSA from the system is the opposite of making it available. To remove it from the system:

1. Make sure the OSA is not in use. Enter a QUERY command for the device to find out if it is.

If it is in use, take appropriate steps to remove it from the virtual machine. The status should say FREE before you go to the next step.

2. Enter:

```
vary offline rdev
```

where *rdev* is the real device number of the OSA.

The OSA is now unavailable to z/VM.

Printers

A great deal of the work users do results in printed output. Accordingly, you need to spend a lot of your time controlling printers.

To control a printer, you must be able to:

- Find out a printer's status
- Set the security label for a printer
- Dedicate a printer to a z/VM user or to the production system
- Remove a printer from a z/VM user or from the production system
- Print files submitted by z/VM users on 3800, impact, and advanced function printers
- Stop a printer and remove it from the system
- Control the print spool files in the print spool file queue
- Control the active file on a printer.

Note: If you run CMS in your operator's virtual machine and you enter the START command without abbreviating it, CMS assumes you mean the CMS START command, not the CP START command. If you want to use the CP START command under CMS to start a printer, abbreviate the command to STA or enter CP START.

If an external security manager (ESM) is installed and security label checking is enabled, creation of printed output may be restricted to only authorized users. For additional information, contact your security administrator.

Finding Out a Printer's Status

To find out the status of a specific printer, enter:

```
query rdev
```

where *rdev* is the real device number of the printer.

To find out the status of a specific print server virtual machine, enter:

```
query prtid
```

where *prtid* is the identification of the advanced function printing (AFP) printer controlled by the print server virtual machine.

To find out the status of all the printers (and other unit record devices) in your z/VM system, enter:

```
query ur all
```

Table 10 on page 132 lists the keywords that appear in responses to QUERY commands for printers. It also shows:

- The meaning of each keyword
- Whether the keyword indicates an active, free, or offline printer
- Whether the keyword indicates a dedicated printer or a printer in use by CP.

Table 10. Printer Status

Device Status Keyword	Keyword Meaning	Type of Status
ATTACHED TO	The printer is dedicated to the indicated virtual machine.	Active (dedicated)
DRAINED	The printer is unavailable for processing spool files.	Free
DRAINING	The printer will become unavailable for processing spool files after finishing the current file.	Active (CP)
INT REQ	The printer requires operator intervention.	Active (CP)
INT REQUIRED	The printer requires operator intervention.	Active (CP or dedicated)
MOUNT REQ	The printer is waiting for you to change the paper (mount a new form).	Active (CP)
OFFLINE	The printer is unavailable to CP.	Offline
PRINTING	The printer is printing a spool file.	Active (CP)
SEC	Specifies the security label of files that can be printed on the printer. This is used only when an ESM is installed and security label checking is enabled. If a security label is not assigned to the printer, the SEC field is followed by the word NONE. No files can be printed until the printer is STARTed with a valid security label.	Active (CP)
SETUP REQ	The impact printer is waiting for you to check the alignment of the paper on the printer.	Active (CP)
STARTED	The printer is available for processing spool files.	Active (CP)
SUBCHOFF	The subchannel for this device is disabled.	Offline Disabled

Dedicating a Printer to a z/VM User

If a printer is offline and you need to dedicate it to a z/VM user, enter:

```
vary online rdev
attach rdev userid vdev
```

where:

rdev
is the real device number of the printer.

userid
is the user's user ID.

vdev
is the virtual device number that the user asked you to assign to the printer.

Note that you can use the method outlined above to dedicate an AFP printer to a print server virtual machine and then enter:

```
start prtid
```

where *prtid* is the identification of the AFP printer controlled by the print server virtual machine.

Note: The *prtid* option is *not* valid for the VARY command. Therefore, this command is meaningless for advanced function printers and their servers.

Dedicating a Printer to the Production System

If a printer is offline, and you want to dedicate it to the production system for its exclusive use, do the following:

1. Enter:

   ```
   vary online rdev
   attach rdev userid vdev
   ```

   where:

   rdev
   is the real device number of the printer.

   userid
   is the user ID of the production system.

   vdev
   is the virtual device number that you want to assign to the printer.

   Note: If *rdev* and *vdev* are the same, you may enter:

   ```
   attach rdev userid
   ```

2. At the production system's system console, enter any commands that the production operating system requires to use the device.

Example: Dedicating a Printer to the Production System

Printer 00E should be dedicated to the production system, VMSPHPO, as virtual device 00E. When you enter:

```
query 00e
```

CP tells you that printer 00E is offline. To dedicate it to the production system, enter:

```
vary online 00e
attach 00e vmsphpo
```

At the VMSPHPO system's system console, enter appropriate VM/SP HPO commands to make sure the printer is usable.

Removing a Printer from a z/VM User

A z/VM user usually removes a printer from the user's virtual machine by using a DETACH VIRTUAL command. If you need to remove a printer from a z/VM user, however, enter:

```
detach rdev userid
```

where:

rdev
is the real device number of the printer.

userid
is the user's user ID.

Now you may remove the printer from the system, use it to print files submitted by z/VM users, or dedicate it to another virtual machine.

To remove a printer from the system, enter:

```
vary offline rdev
```

where *rdev* is the real device number of the printer.

The printer is now unavailable to z/VM.

Removing a Printer from the Production System

Removing a printer from the production system is the opposite of dedicating it:

1. At the production system's system console, enter any commands that the production operating system requires to release the device from use.
2. Enter:

   ```
   detach rdev userid
   ```

   where:

   rdev
   is the real device number of the printer.

   userid
   is the user ID of the production system.

Now you may remove the printer from the system, use it to print files submitted by z/VM users, or dedicate it to another virtual machine.

To remove the printer from the system, enter:

```
vary offline rdev
```

where *rdev* is the real device number of the printer.

The printer is now unavailable to z/VM.

Printing Spool Files on a 3800

This section describes how to print files on an IBM 3800 Printing Subsystem Model 1 (referred to in this section as simply “3800”). If you want to find out how to print files on a 3203, 3262, 4245, or 4248 impact printer, see “Printing Spool Files on an Impact Printer” on page 144.

For convenience, this section refers only to 3800 Model 1 printers. You may also print files on a 3800 Model 3 printer in Model 1 compatibility mode. The procedure is the same for both models, with the following exceptions:

- You may use an FCB of 10 lines per inch for the 3800-3 or 3800-6, but not for the 3800-1.
- You must use a different image library for a 3800-3 or 3800-6 than for a 3800-1.
- Files created on a virtual 3800-3 may not print exactly as intended on a real 3800-1 printer. Therefore, IBM recommends that your installation set up different print file classes for the two kinds of files. Then you can use the appropriate real printer for each kind of file.

Note: The 3800-6 printer is supported but can only be run as a 3800 Model 3.

Overview of Printing Spool Files on 3800s

To print spool files, whether you are using a 3800 or impact printers, you must have:

- A printer that is set up to print a certain group of files
- A group of files in the print queue whose characteristics match the printer's setup.

Spool files come into the print queue as users create them. As the files arrive, a printer prints any files that match its setup. Files that do not match remain in the print queue. To print these nonmatching files, you change the printer's setup.

If you have more than one printer, and each is set up to print a different group of spool files, you can print more than one group at a time. Alternatively, if you want to print one particular group of files as quickly as possible, you can set up all your printers to print the same group.

Thus, printing spool files is a matter of responding to the changing print queue. Files come into the print queue and either match a printer's setup (get printed) or do not match (wait). When too many files are waiting, you change the setup on one or more printers to match the characteristics of the waiting files.

That said, the main difference between printing on a 3800, as opposed to an impact printer, is in how you set up the printer. Because a 3800 has more features, it requires more settings, and this in turn may require more options on the START command. But your installation can make 3800 operation simpler by:

- Specifying default values for many of the 3800 settings.

  During system generation, your installation can set up default 3800 settings to reflect your normal print operation. Doing so reduces the number of options that you need to enter on the START command.
- Setting up PF keys for commands you use most often in 3800 operation.

  During system operation, you can set up PF keys to enter the commands you use to operate a 3800. This reduces the amount of typing you need to do to enter, for example, a START command with many options. If your installation sets up your virtual machine to run CMS, you can further simplify 3800 operation by creating a PROFILE EXEC to set up the PF keys.

Though your installation may use a combination of these methods to make your job easier, the procedure that follows does not assume this has been done.

The procedure that follows shows you how to start with a 3800 printer in any state and print a sample group of files. The procedure assumes that some settings are normal for 3800 operation. For example, it assumes that most files you print do not need a forms overlay and that you want to print files without regard to their 3800 load CCW content. “Using Other START Command Options for 3800 Printers” on page 139 describes other 3800 settings you can specify. If necessary, substitute these other settings as appropriate.

The procedure that follows also assumes you are familiar with the way z/VM presents both printer and spool file status. If you are not, see Appendix B, “Printer and Print File Status,” on page 263. The appendix also describes ways your installation is likely to group spool files.

To print a new group of spool files on the 3800, perform the following steps:

1. If necessary, make the 3800 available to the system.
2. Drain the 3800.
3. Find out the status of the 3800.
4. Find out what files are waiting to be printed.
5. Choose a new group of files to print.
6. Choose the correct START command options for the new group of files.
7. Choose START command options to correctly control the 3800.
8. Enter the START command with all the correct options.
9. If necessary, change the paper on the 3800.
10. Start the 3800.

The next sections explain these steps in detail, using settings assumed to reflect typical 3800 printer operation. “Using Other START Command Options for 3800 Printers” on page 139 explains alternate settings you may specify.

Making the 3800 Printer Available to the System

Before you can use a 3800 to print files, it must be available to the system. To find out if it is available to the system, enter:

```
query rdev
```

where *rdev* is the real device number of the 3800.

If the response does not indicate that the device is offline or attached to a virtual machine, then it is already available to the system. Go to the next step.

If the status says the 3800 is offline, enter:

```
vary online rdev
```

where *rdev* is the real device number of the 3800. Go to the next step.

If the 3800 is dedicated to a virtual machine, remove it from the virtual machine, and go to the next step. See "Removing a Printer from a z/VM User" on page 133 or "Removing a Printer from the Production System" on page 134, as appropriate.

Draining the 3800 Printer

The next thing to do when you are getting ready to print a group of files is to make sure the 3800 does not print them before you are ready. An easy way to do this is to enter the DRAIN command. In z/VM, this is called draining a printer.

The DRAIN command does two things: it tells the 3800 to finish any file it is now printing, and it puts the 3800 in a state that prevents it from printing any other file, that is, the drained state.

Normally, a 3800 that has already been set up to print is in the started state. In this state, the 3800 may print a file before you are ready. Whether it does depends on how the 3800 is set up and what characteristics the files in the print queue have. If there are no files in queue with characteristics that match the setup of the 3800, no files print. In this case, it is not necessary to drain the 3800. But by draining the 3800 anyway, you can make sure no new files print.

To drain the 3800 printer, enter:

```
drain rdev
```

where *rdev* is the real device number of the 3800.

If the 3800 is active, CP sends you two messages, one when the 3800 starts draining and another when it is drained. Otherwise, CP sends you just the drained message. When CP tells you the 3800 is drained, continue with the next step.

Finding Out the Status of a 3800 Printer

Printing takes place only when the characteristics of a file in the print queue match the way a printer is set up. You change the way a 3800 printer is set up through options on the START command, which you enter in a later step. To find out what options you need to specify on the START command, you must find out the status of the 3800. To do this, enter:

```
query rdev
```

where *rdev* is the real device number of the 3800.

In response, CP displays the status of the 3800. If you are not familiar with the status of 3800 printers, see the detailed explanation in "Understanding Printer Status" on page 263.

Finding Out What Files Are Waiting to Be Printed

After you know how the 3800 is set up, you need to find out what files are waiting to be printed. To do this, enter:

```
query printer
query printer exp
```

In response, CP displays information about all the files that are waiting to be printed. If you are not familiar with the status of files in the print queue, see the detailed explanation in “Understanding Print File Status” on page 268.

Choosing a New Group of Files to Print on the 3800

From the files that are waiting, choose a group of files to print.

Each installation classifies its printed output differently, but it is common to use a combination of print class, form, and destination to group output. If your installation has a lot of different output groups, you probably have a table that tells you, for each kind of print file, what options to use on the START command. For 3800 printers, the START command options most likely to vary by group are:

- **CLASS** for the print class (A through Z or 0 through 9)
- **FORM** for the type of paper to use (a name your installation uses by convention)
- **DEST** for the destination value of a particular printer or group of printers.

Finally, you may need to select files that require a forms overlay. “Printing Files with a Forms Overlay on a 3800” on page 142 shows you how to do this. For now, choose files that do not require a forms overlay.

Choosing START Command Options for New Files (3800 Printers)

At this point, you have decided which files to print on the 3800. Now you need to choose the correct options to specify with the START command for those files. The options are:

- **USER** *userid*|SYSTEM

  where *userid* is the user ID whose files will print. SYSTEM specifies any users files will print.

- **CLASS** *abcdefgh*

  where *abcdefgh* is a string of one to eight classes. If you specify CLASS *, any class file will print.

- **FORM** *operform*

  where *operform* is the name of the type of paper the output requires.

- **DEST** *dest*

  where *dest* is a 1 - 8 character value assigned to the printer. Up to four destination values may be specified. Specify a DEST of * to indicate that files are to be selected for processing regardless of the file's DEST setting.

- **CHARS** *cccc*

  where *cccc* is the name of the character set the 3800 should use to print the separator pages. The 3800 also uses this character set to print a file if the file has no other character set associated with it.

- **FCB** *ffff*

  where *ffff* is the default spacing. You may specify the spacing as either a number (6, 8, 10, or 12), indicating lines per inch, or as a name. If you specify a name, the 3800 uses the FCB defined with that name in the image library it is using. You may specify an FCB of 10 lines per inch only for a 3800 Model 3 or 3800 Model 6 printer.

- **SECLABEL** *ssssssss*

  where *ssssssss* is a 1 - 8 character alphanumeric value that identifies the security label of files to be printed on the printer. The SECLABEL option is valid only if an ESM is installed and security label checking is enabled.

Choosing the START Command Options for 3800 Printer Control

In addition to the settings that relate to specific kinds of files, you must also decide how you want to operate the 3800 printer. "Understanding Printer Status" on page 263 describes the various settings in detail. For now, look at the printer status you got when you entered the QUERY command for the device. It should say:

- On the second line, MANUAL, SEP, LIMIT, and ANY3800
- On the third line, FLASH *name*, IMAGE *imagelib*, CHARS *name*, FCB *name*, and HOLD
- On the fourth line, DEST *dest*
- On the fifth line, NOAFP.

If necessary, select the following START command 3800 printer control options:

- MANUAL, if the status says AUTO
- SEP, if the status says NOSEP
- LIMIT for the number of records of a file to be printed
- ANY3800, if the status says NO3800 or BEG3800
- FLASH OFF, if the status displays a name after FLASH
- IMAGE *imagelib*, if the status displays after IMAGE an image library name your installation does not normally use
- CHARS for the default character set to use (a name your installation defines as a member of a 3800 image library)
- FCB for the default spacing to use (6, 8, 10, or 12 lines per inch, or a name your installation defines as a member of a 3800 image library)
- HOLD, if the status says PURGE
- NOAFP, to prevent the printer from choosing files meant for advanced function printers
- SECLABEL, to specify the security label of files to print on the printer.
- MARK/NOMARK, to indicate whether 3800 separator trailer pages are being marked with separator bars.

Entering the START Command for the 3800

Based on the options you chose in the previous steps, you can now enter the START command, as follows:

```
start rdev user system file-options printer-options
```

where:

rdev
is the real device number of the 3800.

file-options
are the options that relate to the files you have chosen to print (CLASS, FORM, and DEST).

printer-options
are any required options that relate to the control of the 3800 printer (MANUAL, SEP, ANY3800, FLASH OFF, IMAGE, CHARS, FCB, HOLD, NOAFP, and SECLABEL, MARK or NOMARK).

You may enter the options in any order.

Example: Starting a 3800 Printer

If you want to start the third floor 3800 printer to select classified listing files for printing, enter:

```
start rdev class c form listing dest floor3
```

where *rdev* is the real device number of the 3800.

In response to the START command, CP sends you several lines indicating that the 3800 has started. Then CP displays a message that asks you to mount the new form.

Note: If you run CMS in your operator's virtual machine and you enter the START command without abbreviating it, CMS assumes you mean the CMS START command, not the CP START command. If you want to enter the CP START command under CMS to start a printer, abbreviate the command to STA or enter CP START.

Changing the Paper and Starting the 3800 Printer

When you receive CP's prompt to mount the new form, go to the 3800 and change the paper, if necessary. (If your installation defines more than one operator form name for the same kind of paper, you may not have to change the actual paper on the printer.) Then, tell CP that you have finished mounting the paper by doing either of the following, whichever is more convenient:

- If you are near the 3800, push the STOP and START buttons on the 3800.
- If you are near your console, enter:

  ```
  start rdev
  ```

  where *rdev* is the real device number of the 3800.

Either of these actions causes CP to recognize that the 3800 is ready to print. After the 3800 warms up, it starts printing the first file in the print queue that matches the characteristics you just finished setting up for the printer. As the first file starts to print, you get a message that tells you that the file is printing.

As each matching file starts to print, you get a similar message. When all the matching files are printed, you can repeat this procedure and print another group of files.

Using Other START Command Options for 3800 Printers

"Printing Spool Files on a 3800" on page 134 describes a procedure for printing a typical group of spool files on a 3800 printer. But you can control your printing operation more flexibly by changing the options on the START command. The following sections describe how to do this in more detail.

Assuring That Files with AFP Characteristics Do Not Print on 3800 Printers

A CP printer can select spool files having AFP characteristics, but because these attributes have no meaning for a CP printer, they may cause erroneous output. To avoid this, enter the START command with the NOAFP option. Even if the CLASS, FORM, and DEST options match, you should still use the NOAFP option to make sure the CP printer does not print AFP-characteristic files. AFP printers are controlled by a print server rather than CP.

Assigning a 3800 Printer to a User (Printing Only One User's Files)

Ordinarily, you print output by class and form, without regard to file owner. You accomplish this by specifying the keywords USER SYSTEM on the START command. But sometimes you may need to print the output of a high-priority user. This is called *assigning* the printer to a user. The user may or may not be the production system.

To assign a 3800 to a user, follow the normal procedure for printing files until you are about to enter the START command. Then, enter:

```
start rdev user userid other-options
```

where:

rdev
is the real device number of the 3800.

userid
is the user ID of the user to whom you want to assign the 3800.

other-options
are the other options you have chosen for the START command.

When CP responds to the START command, continue with the rest of the procedure as usual.

When you are ready to print all system files again, follow the normal procedure for printing files until you are about to enter the START command. Then, enter:

```
start rdev user system other-options
```

After you finish the rest of the procedure, the 3800 again starts printing available files without regard to their owners.

Example: Assigning a 3800 Printer to a User

Suppose you are printing files and user VIP asks you to print all his files now. The printer you are using is a 3800 printer that has real device number B32. User VIP's files are all of class V and form STANDARD. The character set associated with this group of files has the name GF10, and the files should be printed at 8 lines per inch. You must change the paper for the new form. The status of the 3800 indicates that the form mode is MANUAL, the separator page mode is SEP, and the other settings are correct. To assign 3800 printer B32 to user VIP:

1. Enter:

   ```
   start b32 user vip class v form standard chars gf10 fcb 8
   ```

2. When CP prompts you, change the paper on the 3800 for the operator form, STANDARD.
3. Enter:

   ```
   start b32
   ```

   The 3800 then starts to print user VIP's spool files.

Allowing CP to Choose the Next Form Automatically (3800 Printers)

On a 3800, you may choose between two *form mode* settings, MANUAL or AUTO. The form mode setting governs how the printer acts when no more files of the current form are left to print.

Normally, you set up a 3800 printer to run with the form mode set to MANUAL. This setting allows you to choose the next form to mount. The 3800 idles until another file of the current form joins the queue, or until you start the printer with a new form.

If you prefer, you can allow CP to choose the next form automatically. When the form mode is set to AUTO and there are no more files of the current form, CP chooses a new form. The new form is a form associated with one of the files waiting to be printed, and if possible, a form that is not on another printer. CP then starts the printer and prompts you to mount the new form.

To set up the 3800 so that CP automatically chooses the next form, follow the normal procedure for printing files until you are about to enter the START command. Then, enter:

```
start rdev auto other-options
```

where:

rdev
is the real device number of the 3800.

other-options
are the other options you chose for the START command.

When CP responds to the START command, continue with the rest of the procedure as usual.

Printing without Separator Pages on a 3800

Most of the time you will want separator pages between the files that you print. These header and trailer pages help you separate the output, which would otherwise run together. But if you are printing on a special form (checks, for example), you may want to print files without separator pages.

To print without separator pages, follow the normal procedure for printing files on a 3800 until you are about to enter the START command. Then, enter:

```
start rdev nosep other-options
```

where:

rdev
is the real device number of the 3800.

other-options
are the other options you have chosen for the START command.

When CP responds to the START command, continue with the rest of the procedure as usual.

Note: If an ESM is installed and security label checking is enabled, the NOSEP option is not valid.

Controlling the 3800 Printer's 3800 Load CCW Mode

A printer's *3800 load CCW mode* determines whether the printer prints files with 3800 load CCWs.

A *3800 load CCW* is a special instruction imbedded in a spool file. It tells a 3800 to load data that defines the spacing, characters, or copy modification data that the printer is to use to print the file. CP keeps track of whether a spool file has any 3800 load CCWs, and if so, whether they appear only at the beginning or throughout the file. Files that do not have 3800 load CCWs are NO3800 files. BEG3800 files refer to files that have 3800 load CCWs only at the beginning of the file. ANY3800 files refer to files that have 3800 load CCWs throughout the file.

ANY3800 is the normal setting for a 3800. This setting allows a 3800 to print a print file without regard to its 3800 load CCW content. But because it takes longer to print files with 3800 load CCWs, you may want to print such files only at nonpeak times. Depending on the nature of your printing workload, you may want to change the 3800 load CCW setting on the 3800.

To change the 3800 printer to print only NO3800 files, follow the normal procedure for printing files until you are about to enter the START command. Then, enter:

```
start rdev no3800 other-options
```

where:

rdev
is the real device number of the 3800.

other-options
are the other options you have chosen for the START command.

When CP responds to the START command, continue with the rest of the procedure as usual.

To change the 3800 to print NO3800 and BEG3800 files, enter:

```
start rdev beg3800 other-options
```

To change the 3800 to print NO3800, BEG3800, and ANY3800 files, enter:

```
start rdev any3800 other-options
```

Printing Files with a Forms Overlay on a 3800

Some of the files you print may require a forms overlay. This feature, when active, causes the 3800 to flash an image onto selected pages of a print file through a film negative mounted in a frame. To print files with a forms overlay on a 3800:

1. Choose a group of files that need a forms overlay.

   To do this, enter:

   ```
   query printer
   query printer exp
   ```

   Under the FLASH heading, find a group of one or more files that have a forms overlay name. For that group of files, also note the CLASS, FORM, DEST, CHARS, and FCB settings you need to enter with the START command.
2. Drain the 3800 by entering:

   ```
   drain rdev
   ```

   where *rdev* is the real device number of the printer.
3. Locate the correct film negative and mount it in the frame in the 3800 printer.

   For each forms overlay name, your installation should have a corresponding film negative. If you do not have a film negative for a particular forms overlay name, contact the person at your installation who makes the negatives available for use.
4. Start the 3800 with the correct forms overlay name by entering:

   ```
   start rdev flash name other-options
   ```

   where:

   rdev
   is the real device number of the 3800.

   name
   is the forms overlay name you want to start.

   other-options
   are the other options you have chosen for the START command.
5. When you receive CP's request to mount the new form, change the paper, if necessary.
6. Enter:

   ```
   start rdev
   ```

   where *rdev* is the real device number of the 3800.

The 3800 then starts printing both files that require the specified forms overlay and files that do not require any overlay. The 3800 flashes an image onto print files that require the overlay. Files that do not require an overlay are printed without it. When printing ends, you may repeat the procedure for a new forms overlay name.

When you wish to print only files that do not require a forms overlay, follow the normal procedure for printing files until you are about to enter the START command. Then, enter:

```
start rdev flash off other-options
```

and continue as usual.

Example: Printing Files with a Forms Overlay on a 3800

Suppose you are in the process of printing files that do not require a forms overlay, and a user needs to have some billing output right away. The output requires a forms overlay, but because it is a high-priority request, you decide to print it. To print the file on 3800 printer B32:

1. Enter:

   ```
   query printer
   query printer exp
   ```

   In response, CP displays the status of all of the files in the print queue. Only the status of the file of interest is shown here.

   ```
   OWNERID  FILE CLASS RECORDS  CPY HOLD USERFORM OPERFORM DEST     KEEP MSG
   FINANCE1 1612 S PRT 00004833 001 NONE BILLING1 BILLFORM BLDG1    ON   OFF

   OWNERID  FILE CLASS RECORDS  FLASH FCB  MDFY   FLSHC LOAD CHARS               SIZE
   FINANCE1 1612 S PRT 00004833 BILL  BFCB BMOD 1 001   ANY  BT10 BT10 BB10 BB10 004
   ```

   From this display, you note the forms overlay name (BILL), the class (S), the form (BILLFORM), and the dest (BLDG1). A table your installation provides also shows that the character set for the separator pages should be GF10, while the FCB should be 8 lines to the inch. (The file specifies its own character sets and FCB, which the 3800 uses to print the file itself.)
2. Enter:

   ```
   drain b32
   ```

3. Locate the film negative labeled BILL and mount it in the frame in the 3800.
4. Enter:

   ```
   start b32 flash bill class s form billform dest bldg1 chars gf10 fcb 8
   ```

5. When CP requests you to mount the new form, load the BILLFORM paper in the 3800.
6. Enter:

   ```
   start b32
   ```

   The 3800 then prints file 1612 with the appropriate billing forms overlay.

Changing a 3800 Printer's Image Library

To operate, a 3800 must have access to an image library. An *image library* is a set of modules that define the spacing, characters, and copy modification data the printer may select to print a spool file. An installation typically stores all of these modules in one image library and specifies that image library's name as the one the 3800 uses by default. So you usually do not have to change the image library of a 3800.

But sometimes you may have to set up a 3800 to use a new image library. For example, the person who maintains the image libraries at your installation may want to test a new one. Or your installation may use different image libraries for different kinds of output. In either case, you must change the 3800 printer' image library.

To find out what image libraries are available, enter:

```
query img
```

CP responds with the status of all of the image libraries in queue. Available image libraries are marked with an A under the CL heading on the response. The name of each image library appears under the FILENAME heading.

To change an image library, follow the normal procedure for printing files until you are about to enter the START command. Then enter:

```
start rdev image name other-options
```

where:

rdev
is the real device number of the 3800.

name
is the new image library's name.

other-options
are the other options you have chosen for the START command.

When CP responds to the START command, continue with the rest of the procedure as usual.

Changing a 3800 Printer's Load Check Action

The load-check action setting controls what a 3800 printer does when it encounters a load-check error while printing a file. A load-check error indicates a problem with a spool file's 3800 load CCW. It usually results from an error in the way the user created the file.

HOLD is the normal load-check action setting; it causes CP to put the file in system hold. If there are an excessive number of files with these problems, though, you may wish to have CP delete them instead (the PURGE setting). To change the load check action to PURGE, enter:

```
start rdev purge
```

where *rdev* is the real device number of the 3800.

After the active file has finished printing, the PURGE setting takes effect. CP then deletes any files with load-check errors.

Printing Spool Files on an Impact Printer

This section describes how to print files on a 3203, 3262, 4245, or 4248 impact printer. If you want to find out how to print files on a 3800, see "Printing Spool Files on a 3800" on page 134.

Overview of Printing Spool Files on Impact Printers

Whether you are using impact printers or 3800 printers, you must have the following to print spool files:

- A printer that is set up to print a certain group of files
- A group of files in the print queue whose characteristics match the printer's setup.

Spool files come into the print queue as users create them. As the files arrive, a printer prints any files that match its setup. Files that do not match remain in the print queue. To print these nonmatching files, you change the printer's setup.

If you have more than one printer, and each is set up to print a different group of spool files, you can print more than one group at a time. Alternatively, if you want to print one particular group of files as quickly as possible, you can set up all your printers to print the same group.

Printing spool files is, therefore, a matter of responding to the changing print queue. Files come into the print queue and either match a printer's setup (get printed) or do not match (wait). When too many files are waiting, you change the setup on one or more printers to match the characteristics of the waiting files.

That said, the main difference between printing on an impact printer and printing on a 3800 is in the way you set up the printer. An impact printer requires different settings, and this in turn requires different options on the START command. Your installation can also make operation of the impact printer simpler by:

- Specifying default values for many of the impact printer settings. During system generation, your installation can set up default impact printer settings to reflect your normal print operation. Doing so reduces the number of options you need to enter on the START command.
- Setting up PF keys for commands you use most often in impact printer operation. During system operation, you can set up PF keys to enter the commands you use to operate an impact printer. This reduces the amount of typing you need to do to enter, for example, a START command with many options. If your installation sets up your virtual machine to run CMS, you can further simplify impact printer operation by creating a PROFILE EXEC to set up the PF keys.

Though your installation may use a combination of these methods to make your job easier, *the procedure that follows does not assume this has been done*.

The procedure that follows shows you how to start with an impact printer in any state and print a sample group of files. The procedure assumes that some settings are normal for impact printer operation. For example, it assumes that you want to print only those files that do not have 3800 load CCWs. "Using Other START Command Options for Impact Printers" on page 149 describes other impact printer settings you can specify. If necessary, substitute these other settings as appropriate.

The procedure that follows also assumes you are familiar with the way z/VM presents both printer and spool file status. If you are not, see Appendix B, "Printer and Print File Status," on page 263. The appendix also describes ways with which your installation is likely to group spool files.

To print a new group of spool files on the impact printer, perform the following steps:

1. If necessary, make the impact printer available to the system.
2. Drain the impact printer.
3. Find out the impact printer's status.
4. Find out what files are waiting to be printed.
5. Choose a new group of files to print.
6. Choose the correct START command options for the new group of files.
7. Set up the correct character set for the new group of files.
8. Set up the correct spacing for the new group of files.
9. Choose START command options to correctly control the impact printer.
10. Enter the START command with all the correct options.
11. Change and align the paper on the impact printer.
12. Start the impact printer.

The next sections explain these steps in detail, using settings assumed to reflect typical impact printer operation. "Using Other START Command Options for Impact Printers" on page 149 explains alternate settings you may specify.

Making the Impact Printer Available to the System

Before you can use an impact printer to print files, it must be available to the system. To find out if it is available to the system, enter:

```
query rdev
```

where *rdev* is the impact printer's real device number.

If the status does not say OFFLINE or ATTACHED to a virtual machine, it is already available to the system. Go to the next step.

If the status says the impact printer is offline, enter:

```
vary online rdev
```

where *rdev* is the impact printer's real device number, and go to the next step.

If the impact printer is dedicated to a virtual machine, remove it from the virtual machine.

Go to the next step. See "Removing a Printer from a z/VM User" on page 133 or "Removing a Printer from the Production System" on page 134, as appropriate.

Draining the Impact Printer

The next thing to do when you are getting ready to print a group of files is to make sure the impact printer does not print them before you are ready. An easy way to do this is to use the DRAIN command. In z/VM, this is called draining a printer.

The DRAIN command does two things: it tells the impact printer to finish any file it is now printing, and it puts the impact printer in a state that prevents it from printing any other file, that is, the drained state.

Normally, an impact printer that has already been set up to print is in the started state. In this state, the impact printer may print a file before you are ready. Whether it does or not depends on how the printer is set up and what characteristics the files in the print queue have. If there are no files in queue with characteristics that match the impact printer's setup, no files print. In this case, it is not necessary to drain the printer. But by draining the impact printer anyway, you can make sure no new files print.

To drain the impact printer, enter:

```
drain rdev
```

where *rdev* is the real device number of the impact printer.

If the impact printer is active, CP sends you two messages, one when the impact printer starts draining and another when it is drained. Otherwise, CP sends you just the drained message. When CP tells you the impact printer is drained, continue with the next step.

Finding Out an Impact Printer's Status

Printing takes place only when the characteristics of a file in the print queue match the way a printer is set up. You change the way an impact printer is set up through options on the START command, which you will enter in a later step. To find out what options you need to specify with the START command, you must find out the impact printer's status. To do this, enter:

```
query rdev
```

where *rdev* is the real device number of the impact printer.

In response, CP displays the status of the impact printer. If you are not familiar with the status of impact printers, see the detailed explanation in "Understanding Printer Status" on page 263.

Finding Out What Files Are Waiting to Be Printed

After you know how the impact printer is set up, you need to find out what files are waiting to be printed. To do this, enter:

```
query printer
query printer exp
```

In response, CP displays information about all the files waiting to be printed. If you are not familiar with the status of files in the print queue, see the detailed explanation in "Understanding Print File Status" on page 268.

Choosing a New Group of Files to Print on the Impact Printer

From the files waiting to be printed, choose a group of files to print.

Each installation classifies its printed output differently, but it is common to use a combination of print CLASS, FORM, and DEST to group output. If your installation has a lot of different output groups, you probably have a table that tells you, for each kind of print file, what options to use on the START command. For impact printers, the START command options most likely to vary by group are:

- CLASS for the print class (A through Z or 0 through 9)
- FORM for the type of paper to use (a name your installation uses by convention)
- DEST for the destination value of a particular printer or group of printers.

Choosing START Command Options for New Files (Impact Printer)

At this point, you have decided which files to print on the impact printer. Now you need to choose the correct options specify with the START command for those files. The options are:

- **CLASS** *abcdefgh*

  where *abcdefgh* is a string of one to eight classes. If you specify *class **, any class file will print.

- **FORM** *operform*

  where *operform* is the name of the type of paper the output requires.

- **DEST** *dest*

  where *dest* is a 1 - 8 character value assigned to the printer. Up to four destination values may be specified. Specify a DEST of * to indicate that files are to be selected for processing regardless of the file's DEST setting.

- **CHARS** *cccc*

  where *cccc* is the name of the character set the impact printer should use to print the separator pages. The printer uses this character set to print a file if the file has no other character set associated with it.

- **FCB** *ffff*

  where *ffff* is the default spacing. You have to specify an FCB name. The printer uses the FCB defined with that name in the image library it is using.

- **SECLABEL** *ssssssss*

  where *ssssssss* is a 1 - 8 character alphanumeric value that identifies the security label of files to be printed on the printer. The SECLABEL option is valid only if an ESM is installed and security label checking is enabled.

Choosing START Command Options for Impact Printer Control

In addition to the settings that relate to specific kinds of files, you must also decide how you want to operate the impact printer. "Understanding Printer Status" on page 263 describes the various settings in detail. For now, look at the printer status you got when you entered the QUERY command for the device. It should say:

- On the second line, MANUAL, SEP, LIMIT, DEFFCB, and ANY3800
- On the third line, NOFOLD, IMAGE *imagelib*, CHARS *name*, and FCB *name*
- On the fourth line, DEST *dest*
- On the fifth line, AFP or NOAFP.

If necessary, select the following options for impact printer control with the START command:

- SEP, if the status says NOSEP
- `NO3800`, if the status says BEG3800 or ANY3800
- `LIMIT`, to specify the maximum number of records of a file to be printed
- DEFFCB uses the default FCB that was loaded in the printer
- `IMAGE` newname, if the status displays after IMAGE
- CHARS for the default character set to use (a name your installation defines as a member of a 3800 image library)
- FCB for the default spacing to use (6, 8, 10, or 12 lines per inch, or a name your installation defines as a member of a 3800 image library)
- `DEST *`, to choose files without regard to destination
- `NOAFP`, to prevent the printer from choosing files meant for advanced function printers.
- `SECLABEL`, to specify the security label of files to print on the printer.

In addition, select:

- SETUP, whether the status says MANUAL, AUTO, or SETUP.

Entering the START Command for the Impact Printer

Based on the options you have chosen in the previous steps, you can now enter the START command, as follows:

```
start rdev user system file-options setup printer-options
```

where:

rdev
is the real device number of the impact printer.

file-options
are the options that relate to the files you have chosen to print (CLASS, FORM, and DEST).

printer-options
are any required options that relate to the control of the impact printer (SEP, NO3800, NOFOLD, IMAGE, CHARS, FCB, INDEX, NOAFP, LIMIT, DEFFCB and SECLABEL).

Always enter the SETUP printer option. This option puts the impact printer in SETUP mode, which allows you to make sure the paper is aligned properly.

You may enter the options in any order.

Example: Starting an Impact Printer

If you want to start the third-floor impact printer to select classified listing files for printing, enter:

```
start rdev class c form listing dest floor3 setup
```

where *rdev* is the real device number of the impact printer.

In response to the START command, CP sends you a couple of lines that indicate that the impact printer has started. Then CP displays a message that asks you to set up the new form for a forms alignment check.

Note: If you run CMS in your operator's virtual machine and you enter the START command without abbreviating it, CMS assumes you mean the CMS START command, not the CP START command. If you want to use the CP START command under CMS to start a printer, abbreviate the command to STA or enter CP START.

Changing the Paper and Starting the Impact Printer

When you receive CP's prompt to set up the new form, go to the impact printer and change the paper, if necessary. (If your installation defines more than one operator form name for the same kind of paper, you may not have to change the actual paper on the printer.) At this point, the impact printer is in SETUP mode, which allows you to make sure the paper is aligned properly.

To check the forms alignment:

1. Push the STOP and START buttons on the printer.

   Doing this causes the impact printer to print an *alignment page*. An alignment page is the first non-blank page of the first file that prints, except that the letters on the page appear as Xs and the numbers appear as 9s. By looking at the alignment page, you can tell whether the file will print properly.
2. Look at the alignment page, and if necessary, adjust the paper on the printer.

   If the paper is not lined up correctly, adjust its alignment on the printer. Then push the STOP and START buttons on the printer again to print out another alignment page. Continue this procedure until the paper is lined up correctly.

When you are satisfied with the alignment of the paper, go back to the console and enter:

```
start rdev manual
```

where *rdev* is the real device number of the impact printer.

This command causes CP to recognize that the printer is ready to print. The impact printer starts printing the first file that matches the characteristics you just finished setting up for the printer. As the first file starts to print, you get a message telling you that the file is printing.

As each matching file starts to print, you get a similar message. When all the matching files have printed, you can repeat this procedure and print another group of files.

Using Other START Command Options for Impact Printers

"Printing Spool Files on an Impact Printer" on page 144 describes a procedure for printing a typical group of spool files on an impact printer. But you can control your printing operation more flexibly by changing the options specified with the START command. The following sections explain in more detail how to do this.

Assuring that Files with AFP Characteristics Do Not Print on Impact Printers

A CP printer can select spool files having AFP characteristics, but since these attributes have no meaning for a CP printer, they may cause erroneous output. To avoid this, enter the START command with the NOAFP option. Even if the CLASS, FORM, and DEST options match, you should still use the NOAFP option to make sure the CP printer does not print AFP-characteristic files.

Note: AFP printers are controlled by a print server rather than CP.

Assigning an Impact Printer to a User (Printing Only One User's Files)

Ordinarily, you print output by class and form, without regard to file owner. You accomplish this by specifying the keywords USER SYSTEM on the START command. But sometimes you may need to print the output of a high-priority user. This is called *assigning* the printer to a user. The user may or may not be the production system.

To assign an impact printer to a user, follow the normal procedure for printing files until you are about to enter the START command. Then, enter:

```
start rdev user userid other-options
```

where:

rdev
is the real device number of the impact printer

userid
is the user ID of the user to whom you want to assign the impact printer

other-options
are the other options you have chosen for the START command.

When CP responds to the START command, continue with the rest of the procedure as usual.

When you are ready to print all system files again, follow the normal procedure for printing files until you are about to enter the START command. Then enter:

```
start rdev user system other-options
```

After you finish the rest of the procedure, the impact printer again starts printing available files without regard to their owners.

Example: Assigning an Impact Printer to a User

Suppose you are printing files and user VIP asks you to print all of his files now. The printer you are using is a 3262 printer that has real device number 003. User VIP's files are all of class V and form STANDARD. The print train image associated with class V has the name T11 and the forms control buffer has the name FCB8. You must change the print train cartridge and the paper for the new output. To assign impact printer 003 to user VIP:

1. Enter:

   ```
   drain 003
   ```

2. Mount the print train cartridge for class V.
3. Enter:

   ```
   loadbuf 003 ucs t11 ver
   ```

4. Look at the print train image that printed out when you entered the LOADBUF command to make sure it is the right one.
5. Enter:

   ```
   loadbuf 003 fcb fcb8
   ```

6. Enter:

   ```
   start 003 user vip class v form standard setup
   ```

7. When CP prompts you, change and align the paper on the impact printer for the operator form, STANDARD.

   To print out an alignment page, press the STOP and START buttons on the printer. If necessary, adjust the alignment.

8. Enter:

   ```
   start 003 manual
   ```

   The impact printer then starts to print user VIP's spool files.

Allowing CP to Choose the Next Form Automatically (Impact Printers)

On an impact printer, you may choose between three *form mode* settings: SETUP, MANUAL, and AUTO.

SETUP mode allows you to align the paper on an impact printer before you start printing files. When you want the impact printer to leave SETUP mode, you may start the printer with the MANUAL setting, the AUTO setting, or with neither setting. If you specify neither MANUAL nor AUTO, the printer reverts to the form mode it had before you put it in SETUP mode. The MANUAL and AUTO form mode settings govern how the impact printer acts when no more files of the current form are left to print.

Normally, you use SETUP mode to start an impact printer with a new form, and then you put the printer into MANUAL mode when you start printing files. The MANUAL setting allows you to choose the next form to mount. The impact printer idles until another file of the current form joins the queue or until you start the printer with a new form.

If you prefer, you can allow CP to choose the next form automatically. When the form mode is set to AUTO and there are no more files of the current form, CP chooses a new form. The new form is a form associated with one of the files waiting to be printed, and if possible, a form that is not on another printer. CP then starts the printer and prompts you to mount the new form.

To set up the impact printer so that CP automatically chooses the next form, follow the normal procedure for printing files until you are about to enter the START command that causes the printer to leave SETUP mode. Then enter:

```
start rdev auto
```

where *rdev* is the real device number of the impact printer.

The impact printer starts printing files. When there are no more files of the current form, CP chooses a new form and waits for you to mount the new form.

If you prefer, when you want the impact printer to leave SETUP mode, you may start the printer without specifying either the MANUAL or AUTO setting. In this case, the printer reverts to the form mode setting that was in effect before you put the printer in SETUP mode.

Printing without Separator Pages on an Impact Printer

Most of the time you want separator pages between the files that you print. These header and trailer pages help you to separate the output, which would otherwise run together. But if you are printing on a special form (checks or labels, for example), you may want to print files without separator pages.

To print without separator pages, follow the normal procedure for printing files on an impact printer until you are about to enter the START command. Then, enter:

```
start rdev nosep other-options
```

where:

rdev
is the real device number of the impact printer.

other-options
are the other options you have chosen for the START command.

When CP responds to the START command, continue with the rest of the procedure as usual.

Note: If an ESM is installed and security label checking is enabled, the NOSEP option is not valid.

Controlling the Impact Printer's 3800 Load CCW Mode

A printer's *3800 load CCW mode* determines whether the printer prints files with 3800 load CCWs. Usually, you print files that contain 3800 load CCWs on a 3800 printer, whereas the only kind of files you print on an impact printer are NO3800 files. The normal setting for an impact printer, therefore, is NO3800.

You may print BEG3800 and ANY3800 files on an impact printer, but doing so does not make use of any required 3800 features. An impact printer ignores 3800 load CCWs.

If you need to change an impact printer to print BEG3800 and NO3800 files, follow the normal procedure for printing files until you are about to enter the START command. Then enter:

```
start rdev beg3800 other-options
```

where:

rdev
is the real device number of the impact printer.

other-options
are the other options you have chosen for the START command.

When CP responds to the START command, continue with the rest of the procedure as usual.

If you need to change an impact printer to print NO3800, BEG3800, and ANY3800 files, enter:

```
start rdev any3800 other-options
```

Changing an Impact Printer's Image Library

To operate, printers need access to an image library. An *image library* is a set of modules that define the spacing, characters, and copy modification data the printer may select to print a spool file. An installation typically stores all these modules in one image library, and specifies that image library's name as the one the printer uses by default. You usually do not have to change a printer's image library.

But sometimes you may have to set up a printer to use a new image library. For example, the person who maintains the image libraries at your installation may want to test a new one. Or your installation may use different image libraries for different kinds of output. In either case, you must change the printer's image library.

To find out what image libraries are available, enter:

```
query img
```

CP responds with the status of all of the image libraries in queue. Available image libraries are marked with an A under the CL heading in the response. The name of each image library appears under the FILENAME heading.

To change an image library, follow the normal procedure for printing files until you are about to enter the START command. Then enter:

```
start rdev image name other-options
```

where:

rdev
is the real device number of the impact printer.

name
is the new image library's name.

other-options
are the other options you have chosen for the START command.

When CP responds to the START command, continue with the rest of the procedure as usual.

Overview of Printing Spool Files on Advanced Function Printers

Advanced function printing (AFP) printers are controlled through print servers, such as the IBM Print Services Facility™ (PSF) licensed program. You can control an AFP printer by specifying the *prtid* (identification of the AFP printer that is controlled by a print server virtual machine) instead of the real device number. The *prtid* operand is valid for the START, DRAIN, FLUSH, QUERY (real device), and SEND commands. For further information on this operand, refer to the specific command as described in the *z/VM: CP Commands and Utilities Reference*.

Note: AFP printers are controlled by a print server machine rather than CP. Therefore, it is necessary that you have some print server program product to support AFP printers and AFP characteristic type files. For more information about advanced function printing, see *Advanced Function Printing Software: General Information*.

The following list outlines the steps that are necessary to print a new group of spool files on an AFP printer:

1. If necessary, make the AFP printer available to the system.
2. Dedicate an AFP printer to a print server virtual machine using the CP ATTACH command.
3. Find out what files are waiting to be printed.
4. Choose a new group of files to print.
5. Choose the correct START command options for the new group of files.
6. Enter the START command with all the correct options, specifying the appropriate *prtid*.

For a more detailed explanation of the steps outlined above, see "Overview of Printing Spool Files on 3800s" on page 134 or "Overview of Printing Spool Files on Impact Printers" on page 144.

Example: Starting an AFP Printer

If you want to start the AFP printer on the first floor to select unclassified memo files for printing, enter:

```
start prtid class u form memo dest floor1
```

where *prtid* is the identification of the AFP printer that is controlled by the print server virtual machine that issues the START command to the AFP printer.

Stopping a Printer

To stop a printer after it finishes printing the current file, enter:

```
drain rdev
```

where *rdev* is the real device number of the printer. You may now set up to print a new group of files or remove the printer from the system.

Sometimes you may need to stop the printer without waiting for the current file to finish printing. For example, the printer may have the wrong print train cartridge for the file, or the file may take too long to print during first shift. If you want to stop a printer immediately and save the file that is interrupted, enter:

```
drain rdev
flush rdev hold
```

where *rdev* is the device number of the printer you want to stop. In response to this command, CP stops printing the active file and puts it back in the print queue in system hold. To print the interrupted file later, release it from system hold. See “Releasing Files in the Print Queue from System Hold” on page 157.

If you want to stop a printer immediately and delete the file that is interrupted, enter:

```
drain rdev
flush rdev all
```

where *rdev* is the device number of the printer you want to stop. In response to this command, CP stops printing the file and deletes it.

Removing a Printer from the System

To remove a printer from the system, do the following:

1. Make sure the printer's status is drained. Enter:

   ```
   query rdev
   ```

   where *rdev* is the real device number of the printer. If CP's response does not indicate that the printer is drained, enter:

   ```
   drain rdev
   ```

   where *rdev* is the real device number of the printer.
2. Enter:

   ```
   vary offline rdev
   ```

   where *rdev* is the real device number of the printer. The printer is now unavailable to z/VM.

Controlling Spool Files in the Print Queue

z/VM gives you options on the commands you use to manipulate spool files in the print queue. Spool files are collections of data to be processed. Spool files in the print queue are files waiting to be printed.

To control files in the print queue, you need to be able to:

- Find out what files are waiting to be printed
- Put one or more files at the top of the print queue
- Change one or more attributes associated with a file in the print queue
- Transfer files from the print queue

- Purge one or more files from the print queue.

Finding Out What Files Are Waiting to Be Printed

To find out how many files are waiting to be printed, enter:

```
query files system
```

CP's response to this command indicates how many spool files are in the reader, printer, and punch spool file queues.

In an SSI cluster you must specify the AVAIL operand in order to display information about all systems in the cluster. However, files that are locked by another system will *not* be displayed. If you do not specify the AVAIL operand, you will obtain information only about the files on the system from which the command was entered.

To display information about all of the spool files that are waiting to be printed, enter:

```
query printer
query printer all
query printer exp
query printer psf
```

In response, CP displays the information. In an SSI cluster, the response for the QUERY PRINTER ALL command differs from the response on a non-SSI system. If you are not familiar with the status of files in the print queue, see "Understanding Print File Status" on page 268.

For information about other options you can specify on the QUERY PRINTER command, see the QUERY READER|PRINTER|PUNCH command in the *z/VM: CP Commands and Utilities Reference*.

Putting Files at the Top of the Print Queue

If you want to print a particular file before any others and the file matches the setup of the printer, you can put that file at the top of the print queue. To do this, enter:

```
order userid printer nnnn
```

where:

userid
is the ID of the user who owns the spool file.

nnnn
is the spool file identification number of the file you want to print next.

To put several files at the top of the print queue, enter:

```
order userid printer nnn1 nnn2 nnn3 ....
```

where:

userid
is the ID of the user who owns the spool files.

nnn1 nnn2 nnn3
are spool file identification numbers of the files you want to print next.

You may enter as many spool file identification numbers for one user ID as fit on the input line.

To put a whole class of files at the top of the print queue, enter:

```
order userid printer class c
```

where:

userid
is the ID of the user who owns the spool files.

c
is the class of files you want to print next.

To put all of the files of a given operator form at the top of the print queue, enter:

```
order userid printer form operform
```

where:

userid
is the ID of the user who owns the spool files.

operform
is the operator form associated with the files you want to print next.

To put all of the files having a particular destination at the top of the print queue, enter:

```
order userid printer dest dest
```

where:

userid
is the ID of the user who owns the spool files.

dest
is the destination value of the files that you want to print next.

For information about other options you can specify with the ORDER command, see the *z/VM: CP Commands and Utilities Reference*.

Changing the Attributes of Files in the Print Queue

The responses to the QUERY PRINTER, QUERY PRINTER EXP, QUERY PRINTER ALL, and QUERY PRINTER PSF commands show all the attributes a file in the print queue can have. See Appendix B, “Printer and Print File Status,” on page 263. Sometimes you need to change some of these attributes. The attributes you usually change are a file's class, form, or destination, and its system hold status. You may also need to change the number of copies or the security label associated with files in the print queue.

The sections that follow tell you how to use the CHANGE command to:

- Change the class, form, or destination of a file in the print queue
- Put files in the print queue in system hold
- Release files in the print queue out of system hold
- Change the conversion status of a file in the print queue
- Change the number of copies of a file in the print queue
- Change the security label value associated with files in the print queue.

Note: If an ESM is installed you may not be authorized to change attributes of files in the print queue. For additional information, contact your security administrator.

For information about other options you can specify with the CHANGE command, see the *z/VM: CP Commands and Utilities Reference*.

Changing the Class, Form, or Destination of a File in the Print Queue

To change the class of a file in the print queue, enter:

```
change userid printer nnnn class c
```

where:

userid
is the ID of the user who owns the spool file you want to change.

nnnn
is the spool file identification number of the file.

c
is the new class.

To change the form of a file in the print queue, enter:

```
change userid printer nnnn form operform
```

where:

userid
is the ID of the user who owns the spool file you want to change.

nnnn
is the spool file identification number of the file.

operform
is the new operator form.

To change the destination of a file in the print queue, enter:

```
change userid printer nnnn dest dest
```

where:

userid
is the ID of the user who owns the spool file you want to change.

nnnn
is the spool file identification number of the file.

dest
is the new destination value.

Putting Files in the Print Queue in System Hold

To prevent a file from being printed, you may put it in system hold. To do this, enter:

```
change userid printer nnnn sys
```

where:

userid
is the ID of the user who owns the spool file you want to put in system hold.

nnnn
is the spool file identification number of the file.

You cannot print the file until you take it out of system hold.

To prevent a class of files from being printed, put them in system hold by entering:

```
change userid printer class c sys
```

where:

userid
is the ID of the user who owns the spool files.

c
is the class of files you want to hold.

You cannot print these files until you take them out of system hold.

To put all of the files in the print queue in system hold, enter:

```
change userid printer all sys
```

Releasing Files in the Print Queue from System Hold

A file in the print queue in system hold cannot be printed. To take a file in the print queue out of system hold, enter:

```
change userid printer nnnn nosys
```

where:

userid
is the ID of the user who owns the spool file you want to release from system hold.

nnnn
is the spool file identification number of the file.

To take a class of files in the print queue out of system hold, enter:

```
change userid printer class c nosys
```

where:

userid
is the ID of the user who owns the spool files you want to release from system hold.

c
is the class of files.

To take all of the files in the print queue out of system hold, enter:

```
change userid printer all nosys
```

Changing the Conversion Status of a File in the Print Queue

Even though a file has been converted, it can still be changed and printed with different options. To cancel any preprocessing that the print server may have done to a file, enter:

```
change userid printer nnnn unconv
```

where:

userid
is the ID of the user who owns the spool file that you want to print.

nnnn
is the spool file identification number of the file that you want to change to an unconverted status.

Changing the Number of Copies of a File in the Print Queue

At a user's request, you may change the number of copies of a file in the print queue. To do this, enter:

```
change userid printer nnnn copy cpy
```

where:

userid
is the ID of the user who owns the spool file you want to print.

nnnn
is the spool file identification number of the file.

cpy
is the total number of copies you want to print.

Changing the Security Label of a File in the Print Queue

Class D users can change the value of the security label associated with any file in the print queue if the following conditions are met:

- An external security manager (ESM) is installed
- Security label checking is enabled, and:
  - The user is exempt from ESM authorization checking, or
  - The system has been placed in a tranquil state by the security administrator.

For additional information, contact your security administrator.

To change the security label of a file in the printer queue, enter:

```
change userid printer nnnn seclabel ssssssss
```

where:

userid
is the ID of the owner of the spool file.

nnnn
is the spool file identification number of the file you want to change the security label value of.

ssssssss
is a 1 - 8 character alphanumeric value to be assigned to the file for the security label.

Transferring Files from the Print Queue

To transfer spool files from a printer, enter:

```
transfer prt class c|userid1|all to|from all|*|userid2 rdr|prt|pun
```

where:

CLASS *c*
is the class of the spool files you want to transfer.

userid1
is the user whose files are to be transferred or reclaimed.

ALL
specifies the transfer of all files in the queue.

TO|FROM *
indicates to or from your own user ID.

userid2
identifies the user to whom the files are transferred.

Example 1: Transferring a Spool File from the Printer to a User

A spool file with spoolid 9234 needs to be transferred from the print queue to the reader queue of USERB. To transfer the spool file, enter:

```
transfer prt 9234 to userb
```

Example 2: Transferring Spool Files from the Printer to Your Punch

Spool files on your printer queue need to be transferred to your punch queue. To transfer the spool files, enter:

```
transfer prt all to * pun
```

Example 3: Reclaiming Spool Files Sent to Other Printer Queues

You need to reclaim spool files sent to the printer queues of other users and place them on your printer queue. To do this, enter:

```
transfer prt all from all prt
```

Note: If an ESM is installed, you may not be authorized to use the TRANSFER command. For additional information, contact your security administrator.

Purging Files from the Print Queue

At a user's request, you may remove one or more files from the print queue. To purge particular files, enter:

```
purge userid printer nnn1 nnn2 ...
```

where:

userid
is the identification of the user whose files are to be purged.

nnn1, nnn2
are spool file identification numbers of the files you want to purge from the print queue.

CP then erases the files.

To purge a class of files in the print queue (for example, a group of files with a class not defined by your installation), enter:

```
purge system printer class c
```

where *c* is the class of files you want to purge from the print queue.

To purge all files in the print queue that have a particular operator form (for example, a group of files with a form not defined by your installation), enter:

```
purge userid printer form operform
```

where:

userid
is the ID of the user who owns the spool files.

operform
is the name of the operator form you want to purge.

To purge all files in the print queue that have a particular destination value, enter:

```
purge userid printer dest dest
```

where:

userid
is the ID of the user who owns the spool files.

dest
is the destination value of the files that you want to purge.

Dumping Files in the Print Queue to Tape

If you need to dump files in the print queue to tape, use the SPXTAPE DUMP command. For more information, see "Dumping Standard Spool Files to Tape" on page 215.

Loading Files in the Print Queue from Tape

If you need to load files from tape back onto the system's print queue, use the SPXTAPE LOAD command. For more information, see "Restoring Standard Spool Files from Tape" on page 217.

Controlling the Active File on a Printer

In addition to managing the queue of spool files waiting to be printed, you must be able to control the active file on a printer. The active file is the file that a printer is now printing.

To control the active file on a printer, you need to be able to:

- Backspace or forward space the file
- Cause the file to print single-spaced
- Print another copy of the file
- Stop the file from printing.

Backspacing and Forward Spacing the Active File on a Printer

If a printer starts producing unreadable output, you need to reprint part of the file it is printing. To cause the printer to go back a number of pages, enter:

```
backspace rdev nnn
```

where:

rdev
is the real device number of the printer.

nnn
is the number of pages you want the printer to go back.

If you want to reprint the entire file, you can cause the printer to start over at the beginning of the file. To do this, enter:

```
backspace rdev file
```

If you want to reprint from the end of the file, enter:

```
backspace rdev nnn eof
```

where:

rdev
is the real device number of the printer.

nnn
is the number of pages from the end of the file where you want the printer to begin.

For example, if you want to reprint the last 50 pages of the file on printer 621, enter:

```
backspace 621 50 eof
```

If you want to reprint most of the file, you can cause the printer to go back to the beginning of the file and space forward a number of pages. To do this, enter:

```
backspace rdev file
forward rdev nnn
```

where:

rdev
is the real device number of the printer.

nnn
is the number of pages you want the printer to go forward from the beginning.

Example: Backspacing the Active File on a Printer

A problem on printer 003 caused the print to run together starting on page 30 of the output. The printer is now at page 40 and stopped. After fixing the problem, enter:

```
backspace 003 file
forward 003 29
```

These commands cause the printer to start printing again at page 29 of the file. You can later discard the unwanted output.

Note: The *prtid* option is *not* valid for the BACKSPACE command. Therefore, this command is meaningless for advanced function printers and their servers.

Causing the Active File to Print Single-Spaced

If a printer starts printing one line per page, a user has probably specified that the file contains carriage control characters when in fact it does not. To force the output to be single-spaced (and more readable), enter:

```
space rdev
```

where *rdev* is the real device number of the printer. The printer now starts to single-space the output of this file.

Note: The *prtid* option is *not* valid for the SPACE command, so this command is meaningless for advanced function printers and their servers.

Printing Another Copy of the Active File

If a file is waiting to be printed, you can use the CHANGE command to create more than one copy. See "Changing the Number of Copies of a File in the Print Queue" on page 157.

If the file is now on the printer and the user needs more copies, enter:

```
repeat rdev nnn
```

where:

rdev
is the real device number of the printer.

nnn
is the number of *additional* copies you need.

Example: Printing Another Copy of the Active File

A user originally requested two copies of a file that is now on printer 003. He calls you and says he needs four copies. To get the additional copies, enter:

```
repeat 003 2
```

Note: The *prtid* option is *not* valid for the REPEAT command, so this command is meaningless for advanced function printers and their servers.

Stopping the Active File from Printing

If you want to stop the file that is on a printer from printing, enter:

```
flush rdev hold
```

where *rdev* is the real device number of the printer. In response, CP stops printing the file, puts it back in the print queue in system hold, and starts printing the next available file. To print the interrupted file later, or on another printer, you must take it out of system hold by entering:

```
change userid printer nnnn nosys
```

where:

userid
is the user identification of the user.

nnnn
is the file's spool file identification number.

Controlling Spool Files in the Reader Queue

z/VM gives you options with the commands you use to manipulate spool files in the reader queue. Spool files are collections of data to be processed. Spool files in the reader queue are files that have been read in from a card reader or that were created by a virtual unit record device.

Spool files in the reader queue are destined for user virtual readers; they are primarily the responsibility of those users. As a system operator, you may perform some of the following functions for files destined for service virtual machines. But, in general, you do not need to manipulate spool files in the reader queue.

To control files in the reader queue, you need to be able to:

- Find out what files are in the reader queue
- Put one or more files at the top of the reader queue
- Change one or more attributes associated with a file in the reader queue
- Put one or more files in the reader queue in system hold
- Purge one or more files from the reader queue
- Transfer one or more files from the reader queue.

Finding Out What Files Are in the Reader Queue

To find out how many files are in the reader queue, enter:

```
query files system
```

CP's response to this command indicates how many spool files are in the reader, printer, and punch spool file queues.

In an SSI cluster, if you specify the AVAIL operand, this command does not display information about files that are locked by another system. If you do not specify the AVAIL operand, this command displays information about all files on the system from which the command was processed.

To list the files in the reader queue, enter:

```
query reader all
```

CP's response to this command lists, in the order of processing, all the spool files in the reader queue. The response looks similar to this:

```
OWNERID FILE CLASS RECORDS  CPY HOLD DATE  TIME     NAME     TYPE   DIST
MAINT   1598 A PUN 00000173 001 NONE 09/18 11:37:32 TESTPUN1 CARDS  1:16-3
MAINT   1603 A RDR 00000084 001 NONE 09/18 11:38:57 TESTRDR1 CARDS  1:16-3
MAINT   1632 A PUN 00000032 001 NONE 09/18 11:39:02 TESTPUN2 CARDS  1:16-3
MAINT   1645 A PUN 00000576 001 NONE 09/18 12:42:83 TESTPUN3 CARDS  1:16-3
MAINT   1668 A PUN 00000893 001 NONE 09/18 12:46:43 TESTPUN4 CARDS  1:16-3
MAINT   1672 T RDR 00000344 001 NONE OPEN- 000C     TESTSYS  CARDS  1:16-3
MAINT   1688 D RDR 00049655 001 NONE OPEN- SYS      TESTRDR6 CARDS  1:16-3
MAINT   1689 A RDR 00000040 001 NONE OPEN- 0012     TREALRDR CARDS  1:16-3
```

In an SSI cluster, one response to the QUERY READER/PRINT/PUNCH ALL command can differ from the response on a non-SSI system. The response looks similar to the following:

```
ORIGINID FILE CLASS RECORDS  CPY HOLD DATE  TIME    NAME     TYPE     DIST
userid   spid c typ nnnnnnnn nnn NONE LOCK- sysname filename filetype distcode
```

or

```
ORIGINID FILE CLASS RECORDS  CPY HOLD DATE  TIME    NAME     TYPE     DIST
userid   spid c typ nnnnnnnn nnn NONE COPY- sysname filename filetype distcode
```

To list the files in the reader queue, along with each file's security label, enter:

```
query reader seclabel
```

CP's response to this command lists, in the order of processing, all the spool files in the reader queue, along with each file's security label. For a class D user, the response looks like this:

```
OWNERID FILE CLASS RECORDS  CPY HOLD USERFORM OPERFORM DEST  SECLABEL
MAINT   1593 A RDR 00000173 001 NONE PLAIN    PLAIN    BLDG2 009BDF
MAINT   1603 A RDR 00000084 001 NONE PLAIN    PLAIN    BLDG1 12345678
MAINT   1632 A RDR 00000032 001 USER PLAIN    PLAIN    BLDG2 543219E
MAINT   1645 A RDR 00000576 001 SYS  PLAIN    PLAIN    BLDG1 89FED7
MAINT   1668 A RDR 00000893 001 USYS PLAIN    PLAIN    BLDG3 7EF9C1
```

For a class G user, the response looks like this:

```
ORIGINID FILE CLASS RECORDS  CPY HOLD FORM  DEST    KEEP MSG SECLABEL
MAINT    1593 A RDR 00000273 001 NONE PLAIN BLDG3   ON   ON  019BCF
MAINT    1603 A RDR 00000096 001 NONE PLAIN BLDG2   OFF  OFF 87654321
MAINT    1632 A RDR 00000032 001 USER PLAIN BLDG1   ON   OFF 54319E
MAINT    1645 A RDR 00000276 001 SYS  PLAIN BLDG1   ON   ON  82FED9
```

For a class D user, the response looks like this if an external security manager (ESM) denies access to the information for certain spool files:

```
OWNERID FILE CLASS RECORDS  CPY HOLD USERFORM OPERFORM DEST  SECLABEL
MAINT   1593 A RDR 00000173 001 NONE PLAIN    PLAIN    BLDG2 009BDF
MAINT   1603 A RDR ******** *** NONE *****    *****    ***** ********
MAINT   1632 A RDR 00000032 001 USER PLAIN    PLAIN    BLDG2 543219E
MAINT   1645 A RDR ******** *** SYS  *****    *****    ***** ********
MAINT   1668 A RDR 00000893 001 USYS PLAIN    PLAIN    BLDG3 7EF9C1
```

For a class G user, the response looks like this if an ESM denies access to the information for certain spool files:

```
ORIGINID FILE CLASS RECORDS  CPY HOLD FORM  DEST    KEEP MSG SECLABEL
MAINT    1593 A RDR ******** *** NONE ***** *****   ***  *** ********
MAINT    1603 A RDR 00000867 001 USER PLAIN BLDG2   OFF  OFF 1673238L
MAINT    1632 A RDR 00009643 001 SYS  PLAIN BLDG2   ON   OFF PROJARO
MAINT    1645 A RDR ******** *** USYS ***** *****   ***  *** ********
```

Note: When an external security manager (ESM) is installed and security label checking is enabled, the ESM is called to verify the user's access to each file. If the user is denied access to a specific file, some of the information about that file is hidden from the user. The only fields of the response that contain information are ORIGINID, FILE, CLASS, HOLD, DATE, and TIME. All other fields contain asterisks. If a security label is not assigned to the file, the SECLABEL field contains the word NONE.

In the response to the QUERY READER *SECLABEL* command, the information is as follows:

- **ORIGINID|OWNERID** is the user ID of the file owner. The file owner is the user who either created the file or put it in the reader queue.
- **FILE** is the spool file identification number, a number CP assigned to the file when it was created.
- **CLASS** tells you the class of the spool file and how the spool file was created. The class can be a letter (A through Z) or a number (0 through 9). The origin may be one of the following:
  - **PRT** (virtual printer)
  - **RDR** (real reader)
  - **PUN** (virtual punch)
  - **CON** (virtual console)
  - **SYS** (CP—for example, an abend dump file).
- **RECORDS** is the number of logical records in the file.

- **CPY** is the number of copies of the file.
- **HOLD** is the status of the file, where:
  - **USER** means the file is in user hold.
  - **SYS** means the file is in system hold.
  - **USYS** means the file is in both user and system hold.
  - **NONE** means the file is not in hold status.
- **DATE** and **TIME** are one of the following:
  - The date and time the file was created.
  - **OPEN- vdev** if a file is still open, where *vdev* is the virtual device number of the virtual device at which the file is queued.
  - **OPEN- SYS** if a file is open but not associated with a virtual device (for example, if the file is being processed by the SPXTAPE command).
  - **OPEN- rdev** if a file is still being read on a real card reader, where *rdev* is the card reader's real device number.
  - **LOCK- sysname** if a file is locked by another member of the SSI cluster, where *sysname* is the identification of the system that is using the file. The *sysname* variable appears under TIME.
  - **COPY- sysname** if a file in an SSI cluster is unavailable for processing because communications with the system that owns the file cannot be made. The *sysname* variable is the identification of the system that has a copy of the file and appears under TIME.
- **FORM** is the name of the paper type understood by the user.
- **USERFORM** is the name of the form associated with the file as understood by the user.
- **OPERFORM** is the name of the form associated with the file as understood by the operator. The form names under **OPERFORM** are the ones you use when you enter a START command for a card punch.
- **DEST** is the destination value of the file, where:
  - *dddddddd* means that the file will be punched by a specific reader or group of readers indicated by the 1 - 8 character destination value.
  - **OFF** means that the file will only be processed by a reader that is specifically started or defaulted to OFF.
  - **ANY** means that the file can be selected on any CP-owned reader, regardless of the reader's DEST setting, as long as the other selection criteria match.
- **NAME** is the file name of the file, if any.
- **TYPE** is the file type of the file, if any.
- **DIST** is a code your installation may use to distribute output.
- **KEEP** is the keep status of the file, where:
  - **ON** means the file is in keep status.
  - **OFF** means the file is not in keep status.
- **MSG** is the message status of the file, where:
  - **ON** means the file is in message status.
  - **OFF** means the file is not in message status.

  If the file is in message status, CP sends a message to the file's owner when you punch the file.
- **SECLABEL** is the security label associated with each file. It is a 1 - 8 character alphanumeric value. The SECLABEL information is displayed only if an ESM is installed, security label checking is enabled, and the SECLABEL option is specified on the QUERY READER command.

For information about other options you can specify with the QUERY READER command, see the QUERY READER/PRINTER/PUNCH command in the *z/VM: CP Commands and Utilities Reference*.

Putting Files at the Top of the Reader Queue

If you want to put a particular file at the top of the reader queue, enter:

```
order userid reader nnnn
```

where:

userid
is the ID of the owner of the spool file.

nnnn
is the spool file identification number of the file.

To put several files at the top of the reader queue, enter:

```
order userid reader nnn1 nnn2 nnn3 ....
```

where:

userid
is the owner of the spool files.

nnn1 nnn2 nnn3
are spool file identification numbers of the files.

For information about other options you can specify with the ORDER command, see the *z/VM: CP Commands and Utilities Reference*.

Changing the Attributes of Files in the Reader Queue

The responses to the QUERY READER ALL, QUERY READER, and QUERY READER EXP commands show all of the attributes a file in the reader queue can have.

To change the attributes of reader files, you use options with the CHANGE command. For more information about the CHANGE command, see the *z/VM: CP Commands and Utilities Reference*.

Note: If an external security manager (ESM) is installed, you may not be authorized to use the CHANGE command. For additional information, contact your security administrator.

Purging Files from the Reader Queue

At a user's request, you may remove one or more files from the reader queue. To purge particular files, enter:

```
purge userid reader nnn1 nnn2 ...
```

where:

userid
is the owner of the reader files.

nnn1 nnn2
are spool file identification numbers of the files you want to purge from the reader queue.

CP then erases the files.

Transferring Files from the Reader Queue

The TRANSFER command transfers files from the reader queue of another user. The operator (as a privilege class D user) can transfer any spool files in the system, regardless of ownership or origin.

Note: If an ESM is installed, you may not be authorized to use the TRANSFER command. For additional information, contact your security administrator.

To transfer spool files from a reader, enter:

```
transfer rdr class c|userid1|all to|from all|*|userid2 rdr|prt|pun
```

where:

CLASS *c*
: is the class of the spool files you want to transfer.

userid1
: is the user whose files are to be transferred or reclaimed.

ALL
: specifies the transfer of all files in the queue.

TO|FROM *
: indicates to or from your own user ID.

userid2
: identifies the user to whom the files are transferred.

Example 1: Transferring Files from the Reader to the Punch

All the class B files on your reader queue need to be transferred to your punch queue. To transfer the class B files, enter:

```
transfer rdr class b to * pch
```

Example 2: Transferring a Spool File from Your Reader Queue to the Reader Queue of a User

Spool file with spool ID 1673 needs to be transferred from your reader queue to the reader queue of USERA. Note that the file is in the reader queue, which is the default. To transfer the spool files, enter:

```
transfer 1673 to usera
```

Dumping Files in the Reader Queue to Tape

If you need to dump files in the reader queue to tape, use the SPXTAPE DUMP command. For more information, see "Dumping Standard Spool Files to Tape" on page 215.

Loading Files in the Reader Queue from Tape

If you need to load files from tape back on to the system's reader queue, use the SPXTAPE LOAD command. For more information, see "Restoring Standard Spool Files from Tape" on page 217.

Controlling the Active File on a Card Reader

You cannot use CP commands to control the active reader spool file (that is, the card deck a card reader is now reading). All the CP commands that control active spool files apply to output spool files. The BACKSPACE, FORWARD, SPACE, REPEAT, and FLUSH commands apply only to printers and card punches.

Chapter 7. Responding to Errors

This section tells you what to do when errors occur while you are operating z/VM. After reading this chapter, you will understand:

- What happens to the z/VM system, the operator's virtual machine, and other users' virtual machines when z/VM terminates
- How to recover the production system when possible.

If you need only a brief summary of the tasks in this chapter, see "Summary of Recovering the Production System's Virtual Machine" on page 170.

What CP Tries to Do During Error Recovery

The processor and z/VM try to correct most errors without your help. These include intermittent and permanent machine errors and system I/O errors.

However, when an error occurs, CP records it and sends a message to your z/VM primary system console. The message notifies you of the error and tells you whether:

- System operation can continue
- System operation can continue, but with fewer resources
- System restart and recovery is beginning.

If CP can isolate the error condition to a single virtual machine, and system integrity is not in danger, a soft abend occurs. During a soft abend, CP takes a soft abend dump and operation continues.

If the error is more severe, CP terminates with a hard abend. A number of errors can cause a hard abend, and some are severe enough to prevent CP from restarting. But in most instances, CP:

- Tries to save the z/VM spool file and system data file queues
- Takes a hard abend dump
- Restarts itself
- Restarts your operator's virtual machine

During a hard abend, CP does not try to save the operating environments of virtual machines. Users have to log on again and reload the operating systems that were running in their virtual machines.

If CP needs your help to restart itself, it issues a message telling you what you must do to recover the error.

During a software re-IPL, whether it was caused by a system incident or you entered the SHUTDOWN REIPL command, CP attempts to determine whether the operator is logged on the primary system console. If the operator's user ID is disconnected or logged on another terminal, CP leaves it disconnected or disconnects the operator when the system reinitializes.

When CP cannot restart the system, you must bring up the system again from the processor complex's system console. Chapter 4, "Bringing Up the System," on page 25 describes this procedure.

Hard and Soft Abend Dumps

When you perform a system restart from the system console, or when CP encounters a software error, CP takes an abnormal termination (abend) dump. The format of the dump depends on which abend CP takes: hard or soft.

A hard abend occurs when CP detects an error condition that it cannot isolate to a single virtual machine. A hard abend also occurs when CP suspects damage to system integrity.

A soft abend occurs when CP can isolate the error condition to a single virtual machine, and system integrity is not in danger.

An abend dump contains (in order):

1. The system abend code.
2. The following information about the processor on which the error occurred:
   - Processor address
   - General-purpose registers
   - Control registers
   - Floating-point registers
   - Time-of-day clock, clock comparator, and CPU timer
   - Prefix register.

   A soft abend dump contains this information only for the processor on which the error occurred. A hard abend dump repeats this information for any other processors in the configuration.
3. The prefix page of the failing processor.
4. The rest of storage as determined by the type of abend dump (soft or hard) and the dump options in effect at the time of the abend.
   - For a hard abend, this can be all of CP-owned storage. Use the SET DUMP command to change the dump environment.

In certain situations the system programmer might ask you to issue the SNAPDUMP command. This command generates a full system dump identical to a CP hard abend dump without terminating the system. For more information about snapdumps, see *z/VM: Diagnosis Guide*.

Errors That Affect CP Termination and Recovery

Although the processor and z/VM try to correct most errors without your help, they cannot correct the following:

- Errors that cause CP to terminate
- Errors that prevent CP from restarting
- Circumstances that prevent CP from recovering the production system's virtual machine.

Errors That Cause CP to Terminate

CP terminates if one of the following errors occurs:

- A second machine check occurs while the CP is handling the first.
- A channel path error occurs.
- Severe damage to the I/O subsystem occurs.
- Integrity damage occurs in CP-controlled storage.
- A hard abend occurs.
- A soft abend set to be handled like a hard abend occurs.

Errors That Prevent CP from Restarting

CP cannot restart the system after a termination if any of the following are true:

- The error occurs in one of the hardware or software error recovery programs.
- The CP module is damaged.
- Severe damage to the I/O subsystem occurs.
- When CP tried to halt a processor during termination, the processor did not respond.

In this case you might need to use the stand-alone dump facility. For more information, see *z/VM: CP Planning and Administration*.

Backing Off to Another Module

You may need to IPL an alternate module if:

- The current module is unreliable
- You need to test a new level of CP.

You can back off to a different CP module by entering the following SHUTDOWN command:

```
shutdown reipl module modulename
```

where *modulename* is the name of the CP module you wish to load.

For example, suppose you are running a test system that contains program updates. The new system is running very slowly and causing some small system errors. The old, stable, module is called CPLOAD MODULE and resides on the minidisk that the current module was loaded from. To shut the system down and re-IPL with the old module, enter:

```
shutdown reipl module cpload
```

If the module which you wish to load resides on a minidisk other than the one that the current module was loaded from, you can enter the following SHUTDOWN command:

```
shutdown reipl mdisk userid vdev module modulename
```

where *userid* is the owner of the target minidisk, *vdev* is the address of the user's minidisk and *modulename* is as before.

CPU Sparing

Transparent CPU Sparing is a server feature. “Spare” processors are built in. If an active processor fails, the machine substitutes a spare processor automatically and transparently to z/VM. In cases where no spare processor is available, the failing CPU enters check-stop state. The CPU is often able to log out its execution state at the time of the check-stop; this may allow z/VM to resume an executing guest without disruption on another CPU.

The z/VM system can react in several different ways when a machine check occurs which check-stops the failing CPU.

1. Abend (usually MCH005) and go into disabled wait state
2. Abend and then restart automatically
3. Reflect the machine check to the guest and continue operating.

Which scenario occurs depends on what code was running at the time the machine check occurred and the nature of the hardware error as reported in the Machine Check Interrupt Code (MCIC).

Guest Operating System

If a guest was running in SIE at the time of the machine check, z/VM will continue to run on the remaining CPUs. If guest execution was damaged, z/VM will attempt to reflect a suitable machine check to the guest; otherwise, guest execution can be resumed without disruption.

Control Program

If z/VM itself (CP code) was running, then an abend will occur. In this case, z/VM may or may not restart itself. If it does not restart, a disabled wait state will be entered and the operator can restart the system manually.

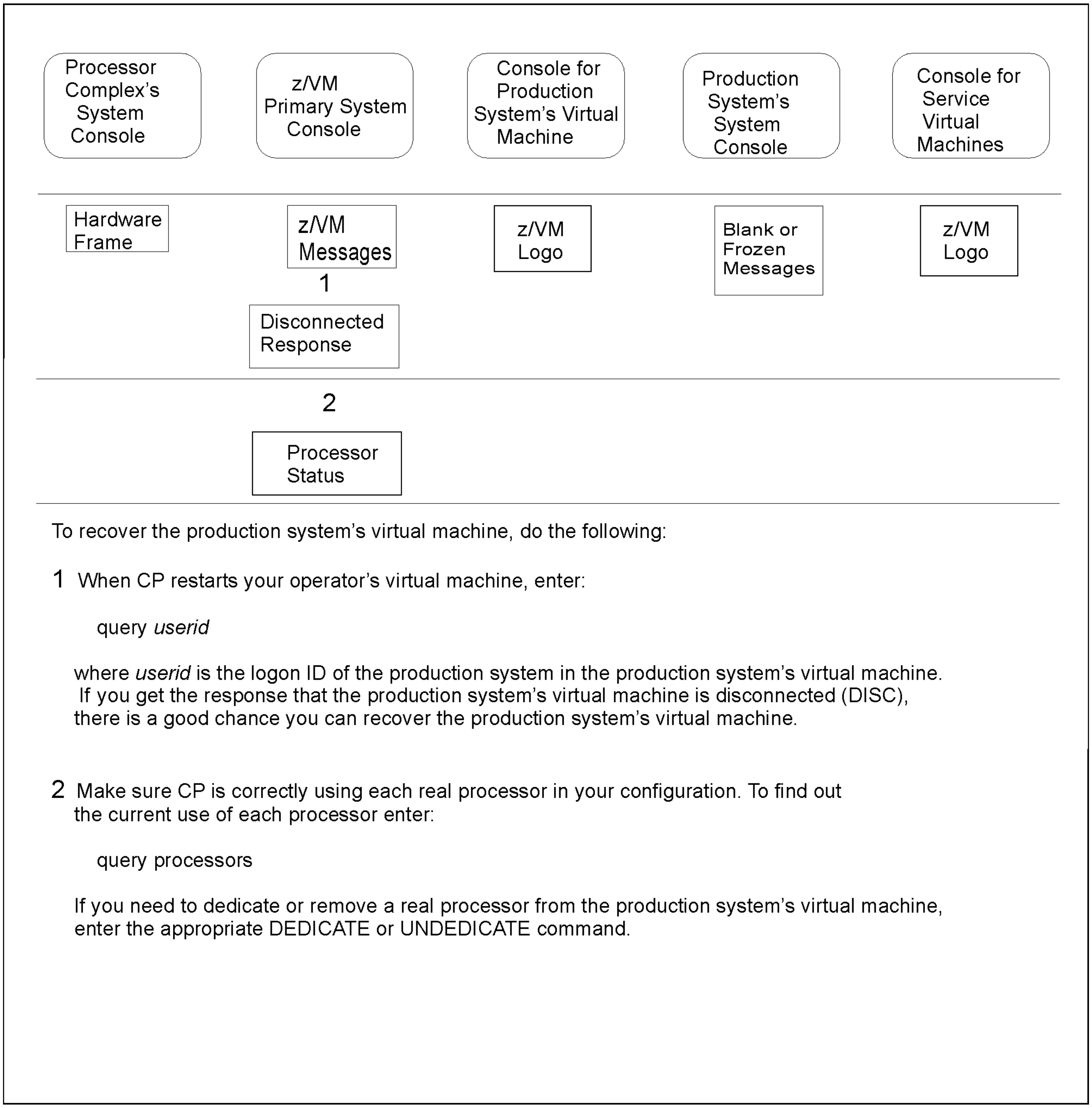
Dedicated Guest Operating System

If a check–stop occurs on a processor that is dedicated to a guest virtual machine, z/VM will determine whether to abend or to reflect the machine check to the guest in the same manner as described in the above paragraphs. In addition the dedicated processor will be undedicated when it is varied offline by the machine check handler. The z/VM operator may issue the CP DEDICATE command for the guest to assign a remaining CPU to it if desired; otherwise, this virtual CPU will share resources with other non-dedicated virtual CPUs.

In any of these cases, a hardware error occurred and this failure should be reported to the proper support personnel.

Summary of Recovering the Production System's Virtual Machine

Figure 8 on page 170 through Figure 10 on page 172 summarize the procedure for recovering the production system's virtual machine.


Processor Complex's System Console
z/VM Primary System Console
Console for Production System's Virtual Machine
Production System's System Console
Console for Service Virtual Machines
Hardware Frame
z/VM Messages
1
Disconnected Response
z/VM Logo
Blank or Frozen Messages
z/VM Logo
2
Processor Status

To recover the production system's virtual machine, do the following:

1 When CP restarts your operator's virtual machine, enter:

```
query userid
```

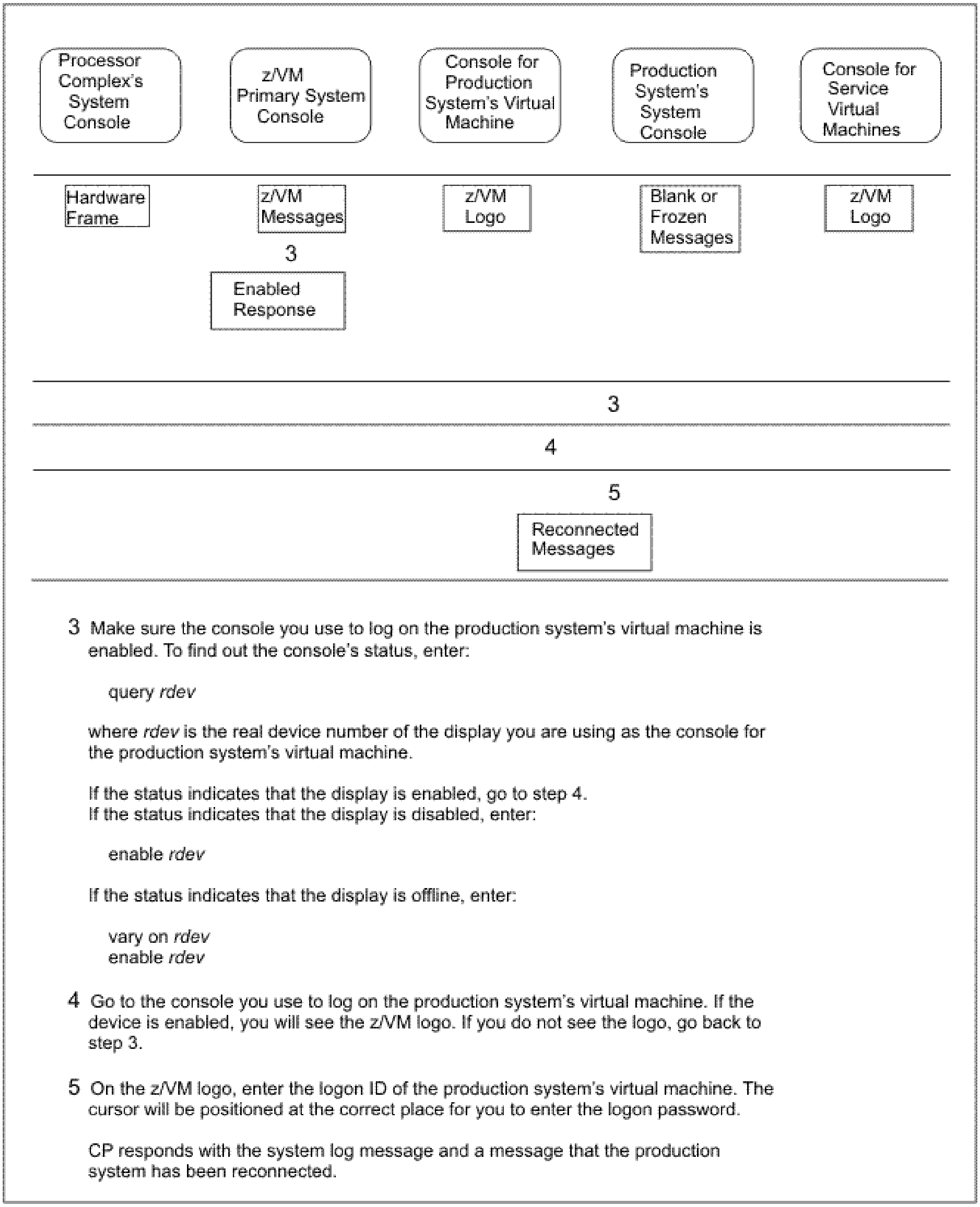
where *userid* is the logon ID of the production system in the production system's virtual machine. If you get the response that the production system's virtual machine is disconnected (DISC), there is a good chance you can recover the production system's virtual machine.

2 Make sure CP is correctly using each real processor in your configuration. To find out the current use of each processor enter:

```
query processors
```

If you need to dedicate or remove a real processor from the production system's virtual machine, enter the appropriate DEDICATE or UNDEDICATE command.

Figure 8. Summary of Recovering the Production System's Virtual Machine (1 of 3)


Processor Complex's System Console
z/VM Primary System Console
Console for Production System's Virtual Machine
Production System's System Console
Console for Service Virtual Machines
Hardware Frame
z/VM Messages
z/VM Logo
Blank or Frozen Messages
z/VM Logo
3
Enabled Response
3
4
5
Reconnected Messages

3 Make sure the console you use to log on the production system's virtual machine is enabled. To find out the console's status, enter:

```
query rdev
```

where *rdev* is the real device number of the display you are using as the console for the production system's virtual machine.

If the status indicates that the display is enabled, go to step 4.
If the status indicates that the display is disabled, enter:

```
enable rdev
```

If the status indicates that the display is offline, enter:

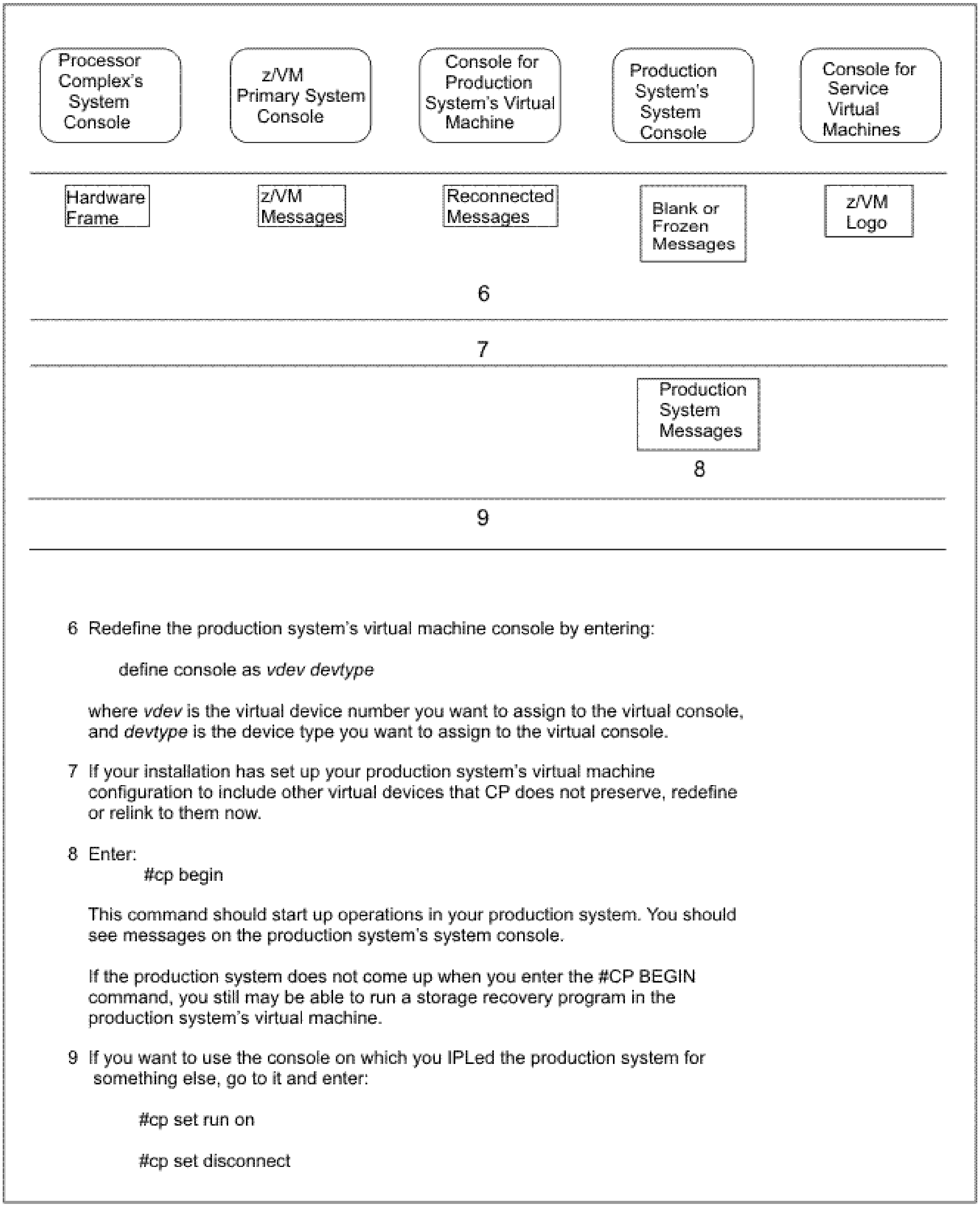
```
vary on rdev
enable rdev
```

4 Go to the console you use to log on the production system's virtual machine. If the device is enabled, you will see the z/VM logo. If you do not see the logo, go back to step 3.

5 On the z/VM logo, enter the logon ID of the production system's virtual machine. The cursor will be positioned at the correct place for you to enter the logon password.

CP responds with the system log message and a message that the production system has been reconnected.

Figure 9. Summary of Recovering the Production System's Virtual Machine (2 of 3)


Processor Complex's System Console
z/VM Primary System Console
Console for Production System's Virtual Machine
Production System's System Console
Console for Service Virtual Machines
Hardware Frame
z/VM Messages
Reconnected Messages
Blank or Frozen Messages
z/VM Logo
6
7
Production System Messages
8
9

6 Redefine the production system's virtual machine console by entering:

```
define console as vdev devtype
```

where *vdev* is the virtual device number you want to assign to the virtual console, and *devtype* is the device type you want to assign to the virtual console.

7 If your installation has set up your production system's virtual machine configuration to include other virtual devices that CP does not preserve, redefine or relink to them now.

8 Enter:

```
#cp begin
```

This command should start up operations in your production system. You should see messages on the production system's system console.

If the production system does not come up when you enter the #CP BEGIN command, you still may be able to run a storage recovery program in the production system's virtual machine.

9 If you want to use the console on which you IPLed the production system for something else, go to it and enter:

```
#cp set run on

#cp set disconnect
```

Figure 10. Summary of Recovering the Production System's Virtual Machine (3 of 3)

Chapter 8. Shutting Down the System

This chapter tells you how to shut down the z/VM system, the production system running in a virtual machine, and any service virtual machines you run. After reading this chapter, you will be able to:

- Inform users about a scheduled shutdown
- Shut down the production system
- Stop any z/VM service virtual machines
- If necessary, stop processing on real devices
- Enter the SHUTDOWN command.

If you need only a brief summary of the tasks in this chapter, see "Summary of Shutting Down the System" on page 176.

Informing Users about a Scheduled Shutdown

Your installation may schedule a shutdown of the z/VM system. As soon as you know when the shutdown is to take place, update the system log message. For information on how to do this, see "Creating and Changing the System Log Message" on page 71.

You should also let your production system users know when the system is to be shut down. If your production system has a system log message, use production system commands to update it with the shutdown information.

Reminding Users of an Impending Shutdown

About an hour before the planned shutdown, send a warning message to all z/VM users that you plan to shut down the system. This gives them time to reach a stopping point and log off.

To send a warning to all users, enter:

```
warning all ***  system shutting down at 20:30  ***
```

Reenter this message at intervals to remind any users who are just logging on that you plan to shut down the system soon.

Just before you shut down the system, send a final warning message to all users:

```
warning all ***  please log off.  system shutting down now  ***
```

You should also enter similar messages at the same intervals to your production system users.

Note:

1. The WARNING command is limited by the length of the input area. If the entire text of a message does not fit in this area, enter another command with the remaining text.
2. If an ESM is installed, you may not be authorized to use the WARNING command. For additional information, contact your security administrator. For additional information, see the *z/VM: CP Commands and Utilities Reference*.
3. If CP does not issue your message the way you entered it, it may be because you are including special line-editing symbols in the text of your message. For example, if logical line editing is in effect, the "#" symbol is your logical line-end symbol; if you include a "#" in your message text, CP cuts off your message.

   To prevent CP from interpreting these symbols as logical line-editing functions, enter:

   ```
   set linedit off
   ```

CP then issues your messages as you enter them.

Shutting Down the Production System

Before you shut down the z/VM system itself, shut down the production system. To do this, enter appropriate production system commands at the production system's system console. Follow your installation's normal shutdown procedures for the production system.

You do not have to log off the production system's virtual machine.

Automating a Shutdown

z/VM allows you to shut down guests automatically, by sending them a shutdown signal and giving them time to respond to that signal. A signal can be sent in response to a hardware event, such as a deactivation of the z/VM image, or as a result of a SHUTDOWN, FORCE, or SIGNAL command.

The amount of time a guest has to shut down can be defined using the SET system configuration statement. This statement can also be used to specify how much time should be reserved to allow z/VM itself to shut down.

Only some guest operating systems have programming support to enable them to receive and react to shutdown signals. For example, z/VM itself has this support. In addition:

- A guest that is enabled for shutdown signals can issue the SET SHUTSIGNAL command to temporarily disable from receiving shutdown signals and later to re-enable for shutdown signals.
- The QUERY SIGNAL command can be used to determine whether a guest is currently enabled to receive shutdown signals.
- The QUERY SHUTDOWN STATUS command can be used:
  - By a guest, to determine whether the entire system is being shut down or whether the guest was the only user signaled to shut down.
  - To tailor how the guest enabled for signals shuts down in an orderly manner.

For more information about automatic guest shutdown, refer to the following:

- The description of the SET system configuration statement in *z/VM: CP Planning and Administration*.
- The descriptions of the following commands in *z/VM: CP Commands and Utilities Reference*: FORCE, QUERY SHUTDOWN STATUS, QUERY SHUTDOWNTIME, QUERY SIGNAL, QUERY SIGNALS, SET SHUTDOWNTIME, SET SHUTSIGNAL, SET SIGNAL, SHUTDOWN, and SIGNAL.

Stopping Service Virtual Machines

Before you shut down the z/VM system itself, shut down any service virtual machines that are running. These virtual machines provide system services and include:

- Accounting virtual machine
- Error recording (EREP) virtual machine
- Symptom record recording virtual machine
- Virtual machines required by z/VM facilities and optional features
- Virtual machines required by supported licensed programs
- SFS file pool server and CRR recovery server virtual machines.

Stopping the Accounting, EREP, and Symptom Virtual Machines

If accounting, error, or symptom record recording is active on your z/VM system, you may log on these virtual machines and then log them off. However, it is not necessary to do this. When you enter the SHUTDOWN command, CP saves any accounting or error recording records that these virtual machines have not yet retrieved. The records are again available at the time of the next warm start.

For information about how to log on the accounting, error, and symptom record recording virtual machines, see “Collecting Accounting Information” on page 183, “Collecting Hardware Error Information” on page 190, and “Collecting Symptom Record Information” on page 202.

Stopping the Virtual Machines for z/VM Facilities and Optional Features

You may need to stop the service machines used by z/VM facilities and optional features, such as:

- DirMaint
- Performance Toolkit
- RSCS
- TCP/IP

For information about when and how to stop these service virtual machines, see the appropriate publications listed in the “Bibliography” on page 279.

Stopping the Licensed Program Virtual Machines

You may need to stop the service machines required by licensed programs. For example, if your installation has installed any of the following licensed programs, you may need to stop the service virtual machines they require:

- VM/Pass-Through
- VTAM Communications Network Applications (VCNA)
- VM/Virtual Telecommunications Access Method (VM/VTAM).

For information about when and how to stop these service virtual machines, see the appropriate licensed program publications.

Stopping the SFS and CRR Server Virtual Machines

If your installation has installed the SFS file pool server and/or CRR recovery server virtual machine, it is important that you notify the server operator of the pending system shut down. The server operator will want to issue the STOP command before system shutdown occurs. For more information about the STOP command, see the *z/VM: CP Planning and Administration*.

Stopping Processing on Real Devices

Before you enter the SHUTDOWN command, use the DRAIN UR command to stop processing on any unit record devices attached to z/VM. Also, if your production system shares any of its DASDs with another system, make sure that shutting down the z/VM system does not disrupt processing on the other system. Also ensure that all data in IBM DASD subsystems with cache and nonvolatile storage (NVS) be transferred to backing storage if any hardware service is planned after system shutdown.

Draining Real Unit Record Devices

If your installation processes spool files on unit record devices attached to z/VM, stop processing on these devices before you shut down the system. To do this, enter:

```
drain ur all
```

In response to this command, CP stops assigning new spool files to the real unit record devices for processing. However, these devices finish processing any active spool files.

As each unit record device finishes processing its active spool file, CP informs you that the device is drained. To keep track of the draining of these devices, enter:

```
query ur active
```

CP's response tells you which devices are still active.

When You Share DASDs with Another System

If your production system shares any of its DASDs with another system, make sure that the shutdown does not disrupt the other system's operation. Normally, this means you must make sure that the DASDs are unavailable for use by the other system. Contact the operator(s) of the other system(s) and follow your installation's normal procedures.

Destaging Data in IBM DASD Subsystems with Cache and NVS

Before performing any service to an IBM DASD subsystem, please refer to “Transferring Data to Backing Storage” on page 103 for information on properly transferring data residing in cache or nonvolatile storage to subsystem backing storage. This action will ensure that no data loss occurs. This is an important consideration because cache data is lost immediately upon the storage control being powered off and NVS data is lost if the control is powered off for more than 48 hours (i.e. battery life).

Entering the SHUTDOWN Command

After the unit record devices are drained and other systems have stopped using any DASDs your production system shares with them, enter:

```
shutdown
```

You can enter SHUTDOWN from any class A virtual machine.

In response to this command, CP stores:

- System status information
- The system log message (only if you did not specify FEATURES ENABLE LOGMSG_FROM_FILE in your system configuration file) file.
- The spool file and system data file queues
- The accounting, EREP, and symptom records in storage.

If you specify a warm start the next time you bring up the system, all of these are again available to the system.

Performing an Unplanned Shutdown

An irrecoverable system error may require you to perform an unplanned shutdown. When this happens, send a warning message to all users of z/VM and the production system. Then, if time permits, drain the unit record devices, quiesce the use of shared DASDs by other systems, and enter the SHUTDOWN command.

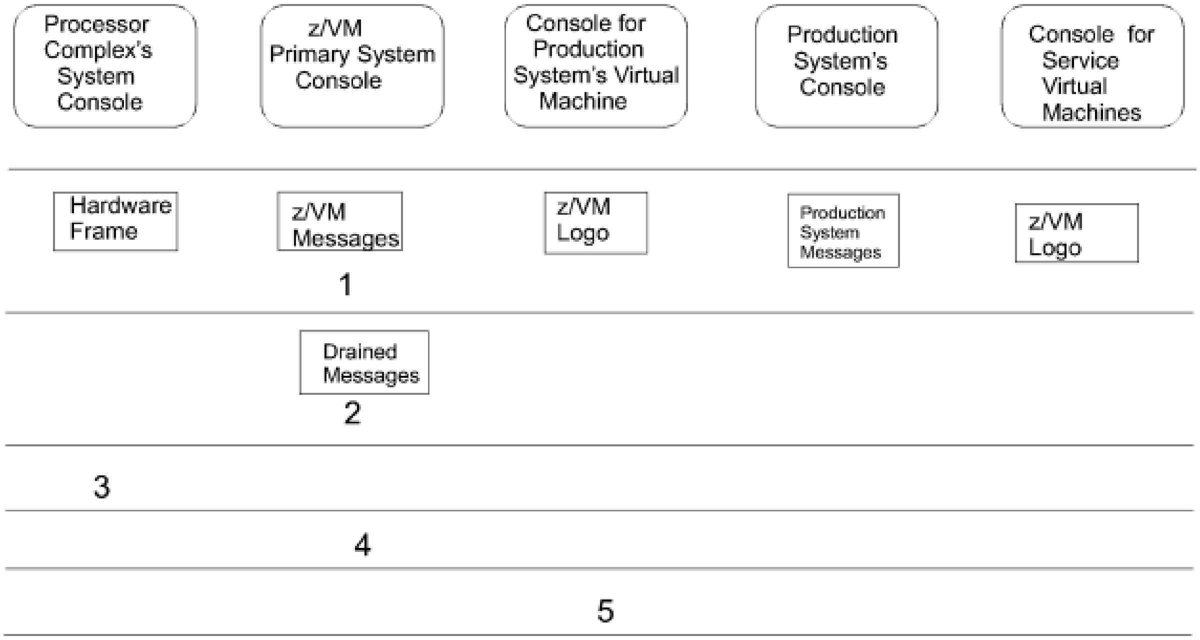
If possible, enter the SHUTDOWN command. During a warm start, CP uses the checkpoint data (for example, the saved spool file queues) that SHUTDOWN creates.

Accessing an Alternate CP Module

If you encounter a system error, you may need to re-IPL another CP module immediately after shutting down the system. To do this, you need to use the REIPL option of the SHUTDOWN command. See “Backing Off to Another Module” on page 169.

Summary of Shutting Down the System

Figure 11 on page 177 summarizes the procedure for shutting down a z/VM system.


Processor Complex's System Console
z/VM Primary System Console
Console for Production System's Virtual Machine
Production System's Console
Console for Service Virtual Machines
Hardware Frame
z/VM Messages
z/VM Logo
Production System Messages
z/VM Logo
1
Drained Messages
2
3
4
5

1 Drain the real unit record devices attached to the z/VM system by entering:

```
drain all
```

2 Wait until CP informs you that all of the devices are drained, or enter:

```
query ur active
```

until all of the devices are drained.

3 Contact the operator(s) of any other systems that share DASD with your production system. Make sure the shutdown will not disrupt their operation.

4 Destage 3990 storage controls with cache and NVS, if hardware service is to be performed after shutdown.

5 Shutdown the z/VM system by entering:

```
shutdown
```

Figure 11. Summary of Shutting Down the System

Chapter 9. Collecting Information about System Operation

This chapter tells you how to collect and print information about system events, performance, accounting, and error recording. If you need only a brief summary of the tasks in this chapter, see "Summary of Collecting Information about System Operation" on page 208.

After reading this chapter, you will be able to:

- State three reasons why system operation includes collecting system data
- Start and print a z/VM console log
- Collect performance information on z/VM
- Collect accounting information on z/VM
- Collect error recording information on z/VM.

Purpose of Collecting Information about System Operation

As the real system operator, your job is to make the real resources of your computer installation available to users so that they can get their work done. Part of your job includes collecting and providing reports of the following kinds of information:

- System events
- System performance
- Accounting
- Error recording.

You provide these reports so your installation can:

- Fix problems that prevent users from using installation resources
- Allocate resources among users in the best way possible
- Plan for future resource requirements.

Collecting Information about System Events

You can use z/VM to help you collect information about system events by starting a console log at your operator's console.

Your production system may also have facilities that can help you collect information about system events.

Starting a Console Log

When you bring up the z/VM system, CP automatically logs activity at the primary system operator's console in a console log. The console log contains all the commands you enter and all of the responses you receive while you are logged on to the operator's virtual machine. It also contains any error or informational messages that CP sends you. If the system abends, CP saves the console log in a spool file, which you can print later. Information in the console log can help support personnel determine the cause of the abend.

"Making Sure CP Is Logging Operator Console Activity" on page 47 discusses how to make sure CP is logging activity at your operator's virtual console. If you are not logged on the primary system operator's user ID, CP may not be automatically logging your console's activity.

To find out whether CP is recording your console's activity in a console log, enter:

```
query virtual console
```

On the first line of CP's response, you see either START or STOP. If it says START, CP is already logging your console activity.

If it says STOP, enter:

```
spool console start
```

CP then starts logging the activity at your operator's console.

Adding Comments to the Console Log

To add comments to your console log, enter:

```
* comment
```

where *comment* is whatever comment you wish to make.

For example, if you want to add a comment to your console spool file that says, “support asked me to enter the following command,” enter:

```
* support asked me to enter the following command
```

The comment and the asterisk are placed in the console log.

Printing the Console Log

To print the console log, enter:

```
close console
```

This command closes the console spool file, places it in the queue of files to be printed, and opens another console spool file. CP normally assigns class T to console spool files.

To print the closed file, follow the procedures in “Printing Spool Files on a 3800” on page 134 or “Printing Spool Files on an Impact Printer” on page 144.

Stopping the Console Log

To stop logging your console's activity in a console log, enter:

```
spool console stop
```

CP then stops logging activity at your console but does not close the current file.

Restarting the Console Log

To resume logging your console's activity, enter:

```
spool console start
```

CP then starts logging activity at your console again, opening a new console spool file if necessary.

Collecting Performance Information

To collect performance information, you may use facilities provided by:

- Performance Toolkit for z/VM

  This optional feature of z/VM provides an enhanced real time performance monitor that allows system programmers and operators to monitor system performance and to analyze bottlenecks. The

Performance Toolkit also provides system console operation in full screen mode. For more information, see the *z/VM: Performance Toolkit Guide*.

- z/VM

  You may enter performance commands and have the responses recorded in the console log.
- CP monitor facility

  You may collect selected performance information in a discontiguous shared segment (DCSS) and use the MONWRITE sample program to write the data to a tape or disk. For more information about this facility, see the *z/VM: CP Planning and Administration*.
- Your production system

  Your production system may have facilities to help you collect information about performance. For more information, see the appropriate publication listed in the Bibliography.
- Your processor complex

  Your processor complex may also have facilities to help you collect performance information. The system activity display (SAD) frames provide information about the use of central processors and channels. For more information, see the appropriate publication listed in the Bibliography.

The next sections discuss how you can use z/VM to help you collect information about performance.

Entering Performance Commands for the Console Log

You can enter CP commands to find out the following about z/VM's performance:

- Load on the system
- Users waiting for I/O devices
- Users waiting for pages
- Users in queue for dispatching
- Resource information for a particular user.

If CP is recording your console's activity in a console log, CP records your commands and its responses.

Finding Out about the Load on the System

To find out what the workload is for the system you are logged on to, enter the following command:

```
indicate load
```

To find out what the workload is for a specific system in the SSI cluster, enter:

```
indicate load at sysname
```

The CP response to the above commands indicates:

- Average processor use (by percentage) for all processors combined, not including dedicated processors
- How much CP is paging (in pages per second)
- Instantaneous approximate percentage of reserved and referenced pages stolen from dispatched users
- Smoothed average number of users in the dispatch, eligible, and dormant lists
- Expansion factor (delay in response time that a virtual machine experiences because of contention for scheduled resources) for those users completing medium-length and long transactions
- How many users are loading users
- Percentage of time the system is running on each processor, identified by processor ID and type (CP, ICF, IFL, or ZIIP)

The value CP displays for the paging rate reflects the number of paging requests that are queued for I/O.

The loading users are users whose transactions were not completed during their previous stay in the dispatch list.

Finding Out Whether a User Is Waiting for an I/O Device

To find out whether a user is waiting for an I/O device, use the INDICATE I/O command. Because it provides a snapshot of the system at a particular instant, enter the command several times before you conclude that there is a problem with a particular device. Enter:

```
indicate i/o
```

CP's response to this command indicates:

- Each user who is waiting for I/O to complete on a device
- The real device number of the device.

Finding Out Which Users Are Waiting for Pages

To find out which users are waiting for pages, enter:

```
indicate paging wait
```

CP's response to this command indicates each user who is waiting for paging to occur.

Finding Out Which Users Are in Queue for Dispatching

To find out which users are in queue for dispatching, enter:

```
indicate queues
```

CP's response to this command indicates the users that are in the dispatch and eligible lists and the virtual CP number. For each user, the response also indicates whether the user is:

- Running, and on which processor (Rnn)
- Waiting for pages (PG)
- Waiting for I/O to complete (IO)
- Waiting for instruction simulation to complete (EX)
- In an enabled wait state (PS).

To find out expanded information about the users in the queue, enter:

```
indicate queues exp
```

CP's response to this command includes, in addition to the information described by the INDICATE QUEUES command, each user's priority, status, and expanded information about processor affinity. Status information indicates whether the user is:

- Receiving HOT SHOT
- Receiving LOCK SHOT
- Receiving INTERACTIVE BIAS
- Past the eligible list deadline
- Exceeding the maximum WSS
- Returning to the eligible list because of preemption
- Returning to the eligible list for exceeding the WSS growth limit
- In the eligible list after having been in the dispatch list for lock shot
- A loading user
- In test-idle state (TI)

- In ready state (R).

Finding Out Resource Information for a Particular User

To find out resource information for a particular user, enter:

```
indicate user userid
```

where *userid* is the user ID of the virtual machine user on which you are checking.

CP's response to this command indicates:

- The user's user ID (USERID).
- The user's virtual machine mode (MACH).
- The user's virtual machine storage size (STOR).
- The virtual device number or name of the named saved system that was on the last IPL command the user entered (IPLSYS).
- The number of devices attached to the user (DEVNUM).
- The number of virtual storage pages that are resident in host storage (RES), in the working set (WS), locked to host real storage (LOCKEDREAL), or reserved in host real storage (RESVD).
- The number of pages, owned by the user, residing on DASD.
- The number of page reads (READS) and page writes (WRITES) for this user since logon.
- The number of the virtual processor to which the following information refers.
- The user's connect time (CTIME), virtual time (VTIME), and virtual and simulation time (TTIME).

  For dedicated processors, the time CP displays for a user may differ from the number shown on a system activity display when certain assists are installed on the real processor. The time displayed on the SAD is the actual time the guest has been running.
- The total number of nonspooled I/O requests the user has made (IO).

  This number resets to zero each time you enter the ACNT command for a user.
- The number of virtual cards the user has read (RDR) and punched (PCH), and the number of virtual lines the user has printed (PRT) since logon or since the last ACNT command was entered.

  Each time you enter the ACNT command for a user, those numbers reset to zero.

Collecting Accounting Information

z/VM normally creates and records accounting records when certain system events occur. Your installation can use these accounting records to analyze overall system operation, to find out if enough resources are available for the system's users, and to bill users for computer services.

Your production system may also have facilities to help you collect accounting information.

How Accounting Information Is Organized

Information about z/VM accounting is organized as follows:

- The *z/VM: CP Planning and Administration* tells you how to set up accounting virtual machines. It also describes the format of each type of accounting record CP creates and records.
- The rest of this section tells you how to control accounting on z/VM. To control the collection of accounting information, you must be able to:
  - Understand how accounting works on z/VM
  - Find out the status of accounting
  - Control accounting record recording
  - Control accounting record retrieval

– Process recorded accounting information.

How Accounting Works on z/VM

On z/VM, accounting involves three steps.

- Accounting records are created by CP and authorized machines. CP records important system events. Authorized virtual machines such as CMS Shared File System servers can record any information they wish, typically resource usage. The creation of accounting records can be controlled or eliminated by using the CP RECORDING command. The accounting records are placed in host storage to await processing.
- Then a special virtual machine called the *accounting virtual machine* must transfer the records from storage to disk. You may have more than one accounting virtual machine.
- Finally, you must process the records on disk in a way that helps your installation analyze z/VM system use.

All three steps are essential to complete accounting:

- CP must create the accounting records; otherwise, system events go unrecorded.
- The accounting virtual machine must move the records to disk; otherwise, you cannot access them to process their information. Further, if the accounting virtual machine does not retrieve the records, they remain in storage, building up as CP creates them. *If you perform a cold start, you lose any accounting records that are in storage.*
- Finally, you must process the records on disk; otherwise, your installation cannot use the information to analyze system operation.

Creating Accounting Records

CP creates and records accounting records when particular system events occur. By default, CP sets accounting on when you bring up z/VM. You can control the setting with the RECORDING command.

Once accounting is on, CP creates an accounting record whenever one of the following events occurs:

- A virtual machine logs off or detaches a virtual processor.
- A user detaches a dedicated device.
- A user releases temporary disk space.
- A virtual machine issues a DIAGNOSE code X'4C'
- A SNA/CCS terminal session ends.
- The system checkpoints (during shutdown, for example).
- You enter the ACNT command.

When one of these events occurs, CP creates and stores an accounting record that describes the event. Then CP notifies the accounting virtual machine of the new record. Accounting records remain in storage until the accounting virtual machine retrieves them.

The default limit for accounting is 20 records. If the number of records in storage reaches that number, CP notifies you, the primary system operator. The buildup of records in storage indicates that retrieval is not active. You can change the limit with the RECORDING command.

Retrieving an Accounting Record

The accounting virtual machine retrieves an accounting record when:

- The virtual machine is defined
- It has started the RETRIEVE program
- It has room on its A disk for new accounting records.

Normally, during system generation, your installation sets up a permanent accounting virtual machine. When you bring up z/VM, CP automatically logs on this virtual machine. However, you may use the

RETRIEVE command to assign the retrieval of accounting records to other virtual machines, if the system programmer has authorized that virtual machine in the IUCV directory control statement. See “Specifying a New Accounting Virtual Machine” on page 187.

After CP logs it on, the permanent accounting virtual machine normally starts retrieving automatically. However, you may also start accounting record retrieval manually. See “Starting Manual Retrieval of Accounting Records” on page 187.

Over time, the accounting virtual machine's A disk fills with accounting records. CP sends a message to you, the primary system operator, when the A disk is 75% full, 90% full, and completely full. You can also log on the accounting virtual machine and check the disk yourself. When the disk is full, you must process some of the old records and erase some files to make room for new ones.

Processing Accounting Records

When enough accounting records accumulate, you can log on the accounting virtual machine and process the records in some way. For example, if your installation writes a program that formats the accounting data, you can run it to produce reports. Your installation's administrator can use these reports to analyze z/VM system use.

You can also dump the accounting file to tape for later processing or send the accounting file to another user for processing. For more information, see “Processing Accounting Information” on page 190.

Finding Out the Status of Accounting

To check the status of accounting record creation and retrieval, enter:

```
query recording
```

In response, CP displays the status of accounting, EREP, and symptom record creation and retrieval. (For more information on error recording and symptom record recording, see “Collecting Hardware Error Information” on page 190 and “Collecting Symptom Record Information” on page 202.) The format of the response is similar to this:

```
RECORDING    COUNT LMT USERID   COMMUNICATION
ACCOUNT  ON  00012 020 DISKACNT INACTIVE
EREP     ON  00000 002 EREP     ACTIVE
SYMPTOM  ON  00000 002 CPSYMP   ACTIVE
```

On the line that says ACCOUNT, the information in the various columns is as follows:

- **RECORDING** is the status of accounting record recording by CP. The status may say ACCOUNT ON or ACCOUNT OFF. In this example, it says ACCOUNT ON, indicating that CP will create and record new accounting records in response to new events.
- **COUNT** is the number of accounting records in storage waiting for retrieval. In this example, the number is 12. Because the number is nonzero, the accounting virtual machine may not be retrieving.
- **LMT** is the number of accounting records that must accumulate in storage before CP notifies you, the primary system operator. In this example, the number is 20, the default limit for accounting.
- **USERID** is the user ID of the accounting virtual machine. In this example it is DISKACNT.
- **COMMUNICATION** is the status of record retrieval by the accounting virtual machine. The status may be INACTIVE, ACTIVE, or ACTIVE PENDING. In this example, the status is INACTIVE, which indicates that the accounting virtual machine is not retrieving. By contrast, the EREP virtual machine and the symptom virtual machine are retrieving, as indicated by their ACTIVE status in the example.

  Persistent ACTIVE PENDING status indicates that the accounting virtual machine is not retrieving properly.

If accounting record retrieval is INACTIVE or ACTIVE PENDING, or if a large number of accounting records are waiting in storage, restart retrieval manually. See “Starting Manual Retrieval of Accounting Records” on page 187.

Controlling Recording of Accounting Records

After finding out its status, you may need to control various aspects of the recording of accounting record. For example, you may need to:

- Stop or start CP creation of accounting records
- Produce specific accounting records
- Control the accounting record limit.

Stopping and Starting the Recording of Accounting Records

By default, CP creates and records accounting records when certain events occur. If you want CP to stop recording accounting records, enter:

```
recording account off
```

In response, CP stops creating accounting records.

When you want CP to start recording accounting records again, you must first purge or retrieve the accounting records queued, then enter:

```
recording account on
```

Purging Accounting Records

To purge any accounting records that are in storage, enter:

```
recording account purge
```

In response, CP purges any accounting records that are in storage.

Producing Specific Accounting Records

Use the ACNT command to produce accounting records for one or more logged-on users, for the system, or both.

A user accounting record shows the system resources a particular user has consumed since the user logged on or since you last entered the ACNT command for this user, whichever is more recent.

A system accounting record shows the system resources charged to the system rather than to a specific user. The record reflects use since you last brought up z/VM, or since you last entered the ACNT command for the system, whichever is more recent.

Example 1—Producing Accounting Records for Specific Users

To produce records that account for the system resources users MVSOPR1 and CMSOPR1 have consumed, enter:

```
acnt mvsopr1 cmsopr1
```

In response, CP produces an accounting record for each user. CP also resets the resource use counters for these two users.

Example 2—Producing System Accounting Records

To produce records that account for system resources not charged to a specific user, enter:

```
acnt system
```

In response, CP produces an accounting record that reflects resource use by the system. CP also resets the resource use counters for the system.

Example 3—Producing Accounting for All System Resource Use

To produce records that account for system resources consumed by both the system and all logged-on users, enter:

```
acnt all
```

In response, CP produces accounting records for all logged-on users, as well as the system. CP also resets all the resource use counters.

Controlling the Accounting Record Limit

By default, CP lets 20 accounting records accumulate in storage before it notifies you that the records are not being retrieved. If you prefer, you may change this limit. For example, if you want to set the limit to 15 records, enter:

```
recording account limit 15
```

Controlling Retrieval of Accounting Records

Once accounting records are being created, you may need to control various aspects of their retrieval. For example, you may need to:

- Specify a new accounting virtual machine
- Start accounting record retrieval manually
- Stop accounting record retrieval
- Disassociate a user ID from accounting record retrieval
- Resume accounting record retrieval
- Find out how much room is on the accounting virtual machine's A disk.

Specifying a New Accounting Virtual Machine

Normally, your installation sets up particular virtual machines to be the accounting virtual machines. If you want to have additional virtual machines collect accounting records, do the following:

1. Make sure the new accounting virtual machine has an A disk available to receive the accounting records.
2. Make sure the system programmer authorizes the new virtual machine to receive the records in the IUCV directory statement. The directory entry of the user entering the RETRIEVE utility must have an IUCV directory control statement authorizing the virtual machine to connect to the CP system service which supports the type of record being collected. *ACCOUNT must be specified for ACCOUNT records.
3. Follow the steps outlined in the next section, "Starting Manual Retrieval of Accounting Records" on page 187.

Starting Manual Retrieval of Accounting Records

Ordinarily, the accounting virtual machine starts retrieving automatically when you bring up z/VM. But there may be times when you need to start retrieval manually. For example, you may find out through error messages or by entering the QUERY RECORDING command that the accounting virtual machine has stopped retrieving. Or your installation may not set up the accounting virtual machine to start retrieval automatically. In either case, you may start retrieval manually, as follows:

1. If necessary, disconnect from the operator's virtual machine by entering:

   ```
   disconnect
   ```

2. Log on the accounting virtual machine.
3. Clear any activity in the virtual machine (including retrieval) by entering:

```
#cp system reset
```

4. Load CMS into storage by entering:

   ```
   ipl 190
   ```

   or, if your installation has installed CMS in a named saved system,

   ```
   ipl cmsname
   ```

   where *cmsname* is the name of your CMS system. If you IPL a named system, (CMS) 16 MB of storage may suffice. However, if you IPL 190, then you will need 20 MB of storage.
5. Make sure there is room on the accounting virtual machine's A disk for more accounting records. To find out how full the A disk is, enter:

   ```
   query disk a
   ```

   If the A disk is almost full, process some of the accounting records. Then erase the old files to free space on the A disk.
6. Start retrieval of accounting records by entering:

   ```
   retrieve account
   ```

7. Disconnect the accounting virtual machine by entering:

   ```
   #cp disconnect
   ```

   You may now use the display for another virtual machine.
8. If necessary, reconnect the operator's virtual machine.

Note that in the preceding commands, the pound sign (#) represents the logical line-end symbol in effect for your virtual machine. If you have redefined the logical line-end symbol, substitute your symbol for the #. If you do not know the line-end character, use the CP QUERY TERMINAL command to display it.

Stopping Retrieval of Accounting Records

Some tasks require you to stop accounting record retrieval. For example, before you can process records stored on the accounting virtual machine's A disk or check how full the A disk is, you must stop retrieval.

To stop retrieval of accounting records:

1. If necessary, disconnect from the operator's virtual machine by entering:

   ```
   disconnect
   ```

2. Log on the accounting virtual machine.
3. Stop accounting retrieval by entering:

   ```
   #cp external
   ```

   In response, you receive a prompt: HCPRET592A ENTER END OR SUMMARY

   Enter:

   ```
   end
   ```

   This stops the retrieval of accounting information. CP continues to store new accounting records in real storage until you resume accounting record retrieval.

   Note: If you enter SUMMARY instead of END, CP displays the number of accounting records retrieved since the last RETRIEVE command, and retrieval continues. To stop retrieval, you must enter the #CP EXTERNAL command again and respond to the prompt with END.

Disassociating a User ID from the Retrieval of Accounting Records

1. Stop the recording of ACCOUNT records for this user ID. To do this from a user ID authorized for the class **A** or **B** version of the CP RECORDING command, enter:

   ```
   recording account off qid userid
   ```

   Or, the user ID being removed (must be authorized for the class **C**, **E**, or **F** version of the recording command), enter:

   ```
   recording account off
   ```

2. If there are accounting records queued in host storage for this user ID and you want to save the data, log on the user ID to be deleted and enter:

   ```
   retrieve account
   ```

   This will place the accounting records in the proper file for later processing. Disconnect this user by entering:

   ```
   #cp disconnect
   ```

   Log back on the authorized user ID.
3. If the accounting records queued in host storage for this user ID are not wanted, they may be purged. To do this from a user ID authorized for the class **A** or **B** version of the CP RECORDING command, enter:

   ```
   recording account purge qid userid
   ```

   Or, the user ID being removed (must be authorized for the class **C**, **E**, or **F** version of the recording command), enter:

   ```
   recording account purge
   ```

   This will PURGE all the accounting records queued in host storage for this user ID.
4. Verify the record count is zero and recording is off for this user ID. To do this, enter:

   ```
   query recording
   ```

   Note: The deleted user ID remains in the warm start data and in the output of the QUERY RECORDING command until z/VM is restarted with the COLD option.
5. If the user ID is specified on the SYSTEM_USERIDS statement in the system configuration file, remove that user ID.

   Note: The user ID OPERACCT is the default if no user ID for accounting is specified in the system configuration file.

Resuming Retrieval of Accounting Records

Ordinarily, after you have processed accounting records or checked on the A disk, CMS is still running in the accounting virtual machine. So, to resume accounting record retrieval, you do not need to go through the whole procedure described in "Starting Manual Retrieval of Accounting Records" on page 187. Instead, perform the following steps:

1. Restart accounting record retrieval by entering:

   ```
   retrieve account
   ```

2. Disconnect the accounting virtual machine by entering:

   ```
   #cp disconnect
   ```

You can now use the display for another virtual machine.

3. If necessary, reconnect the operator's virtual machine.

Finding Out How Full the Accounting A Disk Is

When the accounting virtual machine's A disk is full, it cannot retrieve any more records. If your installation uses the IBM-supplied PROFILE EXEC to start up the accounting virtual machine, you receive a message about the A disk's status when the virtual machine IPLs CMS. Also, CP sends you, the primary system operator, a message when the A disk is 75% full, 90% full, and completely full. Between messages, you may wish to check the disk yourself.

To see how much room there is on the accounting virtual machine's A disk:

1. Stop accounting record retrieval.

   To do this, see "Stopping Retrieval of Accounting Records" on page 188.
2. Enter:

   ```
   query disk a
   ```

   In response, CMS tells you how full the A disk is.
3. Resume accounting record retrieval.

   To do this, see "Resuming Retrieval of Accounting Records" on page 189.

Processing Accounting Information

From time to time, you must process the accounting records stored on the accounting virtual machine's A-disk and make room for new ones. You may do this when you notice the A-disk is nearly full or at intervals set by your installation. In either case, you must:

1. Stop accounting record retrieval.

   To do this, see "Stopping Retrieval of Accounting Records" on page 188.
2. Process the accounting information.

   You can process the accounting records on the A-disk in one of several ways:

   - You can run a program to format the accounting data if such a program is available.
   - You can use the CMS command, TAPE DUMP, to dump the accounting files to tape for later processing.
   - You can send the files to another user for processing.

     To send a file to another virtual machine user, first set up your virtual punch so that CP sends all punch files to the reader of the other virtual machine. Enter:

     ```
     spool punch to userid reader
     ```

     where *userid* is the user ID of the virtual machine to which you are sending the file.

     Then use the CMS command, DISK DUMP, to send the file to the user.

   After you have processed the accounting records, use the CMS command, ERASE, to erase the old files. Erasing the files makes room on the A-disk for new records.
3. Resume accounting record retrieval.

   To do this, see "Resuming Retrieval of Accounting Records" on page 189.

Collecting Hardware Error Information

CP normally generates error recording (EREP) records when certain hardware errors occur. Your installation can use these EREP records to help support personnel diagnose and fix hardware problems.

You can also use the SET MITIME command to control how often CP notifies you of missing interrupts on a device. A console log that contains these messages can help support personnel diagnose problems with the device.

Finally, your production system may have facilities to help you collect hardware error information.

How Error Recording Information Is Organized

Information about z/VM error recording is organized as follows:

- The *z/VM: CP Planning and Administration* tells you how to set up an EREP virtual machine.
- The *Environmental Record Editing and Printing Program User's Guide and Reference* book tells you about the environmental record editing and printing (EREP) program, its options, and the format of each type of EREP record. You use the EREP program to format and print EREP records.
- The *z/VM: CP Commands and Utilities Reference* tells you how to invoke the EREP program through the CPEREPXA utility.
- The rest of this section tells you how to control error recording on z/VM. To control the collection of EREP information, you must be able to:
  - Understand how error recording works on z/VM
  - Find out the status of error recording
  - Control EREP record creation
  - Control EREP record retrieval
  - Process recorded EREP information.

How Error Recording Works on z/VM

On z/VM, error recording involves three steps.

- First, CP must create and store EREP records.
  - CP generates error records when creating a hard abend, soft abend, SA dump or VM dump.
  - For VM dumps, virtual machines can pass error records with guest information, which CP updates and records.
- Then a special virtual machine called the *EREP virtual machine* must transfer the records from storage to disk. You may have more than one EREP virtual machine.
- Finally, you must process the records on disk in a way that helps your installation diagnose and fix hardware problems.

All three steps are essential to complete error recording:

- CP must create the EREP records; otherwise, system events go unrecorded.
- The EREP virtual machine must move the records to disk; otherwise, you cannot access them to process their information. Further, if the EREP virtual machine does not retrieve the records, they remain in storage, building up as CP creates them. *If you perform a cold start, you lose any EREP records that are in storage.*
- Finally, you must process the records on disk; otherwise, your installation cannot use the information to diagnose and fix hardware problems.

EREP Record Recording

CP creates and records EREP records when particular system events occur. By default, CP sets error recording on when you bring up z/VM. You can control the setting with the RECORDING command.

Once error recording is on, CP creates and records an EREP record whenever one of the following events occurs:

- A hardware error occurs while CP is active.

- A hardware error occurs while a virtual machine is active in virtual supervisor state and has previously issued the SET SVC76 CP command.

  CP does not create EREP records for virtual machines in virtual problem state. A virtual machine creates its own EREP records if the user enters the CP SET SVC76 z/VM command. Therefore, CP does not create EREP records for these virtual machines.

Note: Sometimes the same error can be presented to CP for recording twice, once by the hardware as part of normal CP hardware error recording and once by a virtual machine as part of SVC 76 simulation. CP records such errors only once. Therefore, when a virtual machine sets SVC 76 to the CP setting, CP creates a complete and unduplicated set of hardware error records.

When an error occurs that directly affects z/VM, CP creates one or more of the following types of EREP records:

- MCHREC—Machine check record
- MDRREC—Miscellaneous data record
- MITREC—XA mode missing interrupt record
- OBRREC—Outboard recording record
- SLHREC—Subchannel logout record
- CRWREC—Channel report word record.

At a virtual machine's request, CP creates one or more of the following types of EREP records:

- OBRREC—Outboard recording record
- MITREC—XA mode missing interrupt record
- MDRREC—Miscellaneous data record
- DDRREC—Device reconfiguration record
- Software error records.

All these record types are described in the *Environmental Record Editing and Printing Program User's Guide and Reference* book.

After CP creates and stores the EREP record, CP notifies the EREP virtual machine of the new record. EREP records remain in storage until the EREP virtual machine retrieves them.

The default limit for EREP is two records. If the number of records in storage reaches that number, CP notifies you, the primary system operator. The buildup of records in storage indicates that retrieval is not active. You can change the limit with the RECORDING command.

EREP Record Retrieval

The EREP virtual machine retrieves an EREP record when all of the following are true:

- The virtual machine is defined
- It has started the RETRIEVE program
- It has room on its A disk for new EREP records.

Normally during system generation, your installation sets up a permanent EREP virtual machine. When you bring up z/VM, CP automatically logs on this virtual machine. However, you may use the RETRIEVE command to assign the retrieval of EREP records to other virtual machines if the system programmer has authorized that virtual machine in the IUCV directory control statement. See "Specifying a New EREP Virtual Machine" on page 195.

After CP logs it on, the permanent EREP virtual machine normally starts retrieving automatically. However, you can also start EREP record retrieval manually. See "Starting EREP Record Retrieval Manually" on page 196.

Over time, the EREP virtual machine's A disk fills up with EREP records. CP sends a message to you, the primary system operator, when the A disk is 75% full, 90% full, and completely full. You can also log on the EREP virtual machine and check the disk yourself. When the disk is full, you must process some of the old records and erase some files to make room for new ones.

EREP Record Processing

When enough EREP records accumulate, you can log on the EREP virtual machine and process them, using the EREP program. Your installation's support personnel can use the resulting reports to diagnose and fix hardware problems.

You can also dump the EREP file to tape for later processing or send the EREP file to another user for processing. For more information, see “Processing EREP Information” on page 198.

Finding Out the Status of Error Recording

To check the status of EREP record creation and retrieval, enter:

```
query recording
```

In response, CP displays the status of accounting, EREP, and symptom record creation and retrieval. (For more information on accounting and symptom record recording, see “Collecting Accounting Information” on page 183 and “Collecting Symptom Record Information” on page 202.) The format of the response is similar to this:

```
RECORDING    COUNT LMT USERID   COMMUNICATION
ACCOUNT  ON  00000 020 DISKACNT ACTIVE
EREP     ON  00001 002 EREP     INACTIVE
SYMPTOM  ON  00000 002 CPSYMP   ACTIVE
```

On the line that says EREP, the information in the various columns is as follows:

- **RECORDING** is the status of EREP record creation by CP. The status may say EREP ON or EREP OFF. In this example, it says EREP ON, indicating that CP will create new EREP records in response to new events.
- **COUNT** is the number of EREP records in storage waiting for retrieval. In this example, the number is 1. Because the number is nonzero, the EREP virtual machine may not be retrieving.
- **LMT** is the number of EREP records that must accumulate in storage before CP notifies you, the primary system operator. In this example, the number is 2, the default limit for EREP.
- **USERID** is the user ID of the EREP virtual machine. In this example it is EREP.
- **COMMUNICATION** is the status of record retrieval by the EREP virtual machine. The status may be INACTIVE, ACTIVE, or ACTIVE PENDING. In this example, the status is INACTIVE, which indicates that the EREP virtual machine is not retrieving. By contrast, the accounting virtual machine and the symptom virtual machine are retrieving, as indicated by their ACTIVE status in the example.

  Persistent ACTIVE PENDING status indicates that the EREP virtual machine is not retrieving properly.

If EREP record retrieval is INACTIVE or ACTIVE PENDING, or if a large number of EREP records are waiting in storage, restart retrieval manually. See “Starting EREP Record Retrieval Manually” on page 196.

Controlling EREP Record Recording

After finding out its status, you may need to control various aspects of EREP record recording. For example, you may need to:

- Stop or start CP creation of EREP records
- Control the EREP record limit
- Intensify error recording for a device.

Stopping and Starting EREP Record Recording

By default, CP creates and records EREP records when certain events occur. If you want CP to stop recording EREP records, enter:

```
recording erep off
```

In response, CP stops creating EREP records.

When you want CP to start creating EREP records again, you must first purge or retrieve the EREP records queued, then enter:

```
recording erep on
```

Purging EREP Records

To purge any EREP records that are in storage, enter:

```
recording erep purge
```

In response, CP purges any EREP records that are in storage.

You can also control the recording of recoverable processor errors. If you do not want CP to create EREP records for these errors, or you (the primary system operator) do not wish to receive messages when they occur, enter:

```
set mode retry quiet
```

When you want CP to resume creating EREP records for recoverable processor errors, enter:

```
set mode retry record
```

Controlling the EREP Record Limit

By default, CP lets two EREP records accumulate in storage before it notifies you that the records are not being retrieved. If you prefer, you may change this limit. For example, if you want to set the limit to 12 records, enter:

```
recording erep limit 12
```

Intensifying Error Recording for an I/O Device

If an I/O device is causing frequent errors, you can intensify error recording for the device. To do this, enter:

```
set record on rdev limit nn byte ww bit y and byte xx bit z
```

where:

rdev
is the real device number of the I/O device.

nn
is the number of I/O errors that must occur before a record is created.

y
is a bit in sense byte *ww*.

z
is a bit in sense byte *xx*.

For an error to be counted against the limit *nn*, the sense bytes produced by the error must match the sense bytes specified.

In response, CP creates an EREP record the next time an I/O error occurs that sets the sense bytes to the specified values. After the error occurs *nn* more times, CP creates another EREP record for the device. CP continues this until 10 EREP records are created or until you enter:

```
set record off
```

Notes:

1. If you want CP to create EREP records when the sense bytes contain either of the values specified on the BYTE operands, use the OR operand on the SET RECORD command instead of the AND operand.
2. Intensive error recording can be in effect for only one device at a time. If you enter the SET RECORD command for another device, intensive error recording ceases for the first device, even if fewer than 10 records have been created.
3. The SET RECORD command may cause error records to be generated for errors that do not require logging. Later when processing the error records using EREP, error messages may be generated indicating that these error records cannot be processed.

Gathering Device Statistics Dynamically

Some devices maintain statistical data counters that may be useful for hardware service management. Miscellaneous data records (record type MDRREC) containing the statistical information are created when the counters overflow. Other events, such as the issuing of a VARY OFFLINE command, will also trigger the creation of a miscellaneous data record for the device. However, these events occur infrequently.

To force the creation of miscellaneous data records (MDRs), use the CP SYNCMDRS command from any virtual machine having a privilege class of A, B, or F:

```
syncmdrs
```

SYNCMDRS will cause MDRs to be created for all online devices that maintain statistical counters. The EREP machine will retrieve these MDRs as part of its usual error record processing.

Controlling EREP Record Retrieval

Once EREP records are being created, you may need to control various aspects of their retrieval. For example, you may need to:

- Specify a new EREP virtual machine
- Start EREP record retrieval manually
- Stop EREP record retrieval
- Disassociate a user ID from EREP record retrieval
- Resume EREP record retrieval
- Find out how much room is on the EREP virtual machine's A disk.

Specifying a New EREP Virtual Machine

Normally, your installation sets up particular virtual machines to be the EREP virtual machines. If you want to have additional virtual machines collect EREP records, then you must do the following:

1. Make sure the new EREP virtual machine has an A disk available to receive the EREP records.
2. Make sure the system programmer had authorized the new virtual machine to receive the records in the IUCV directory statement. The directory entry of the user entering the RETRIEVE utility must have an IUCV directory control statement authorizing the virtual machine to connect to the CP system service which supports the type of record being collected. *LOGREC must be specified for EREP records.
3. Follow the steps outlined in the next section, "Starting EREP Record Retrieval Manually" on page 196.

Starting EREP Record Retrieval Manually

Ordinarily, the EREP virtual machine starts retrieving automatically when you bring up z/VM. But there may be times when you need to start retrieval manually. For example, you may find out through error messages or by entering the QUERY RECORDING command that the EREP virtual machine has stopped retrieving. Or your installation may not set up the EREP virtual machine to start retrieval automatically. In either case, you can start retrieval manually, as follows:

1. If necessary, disconnect from the operator's virtual machine by entering:

   ```
   disconnect
   ```

2. Log on the EREP virtual machine.
3. Clear any activity in the virtual machine (including retrieval) by entering:

   ```
   #cp system reset
   ```

4. Load CMS into storage by entering:

   ```
   ipl 190
   ```

   or, if your installation has installed CMS in a named saved system:

   ```
   ipl cmsname
   ```

   where *cmsname* is the name of your CMS system.
5. Make sure there is room on the EREP virtual machine's A disk for more EREP records. To find out how full the A disk is, enter:

   ```
   query disk a
   ```

   If the A disk is almost full, process some of the EREP records. Then erase the old files to free space on the A disk.
6. Start EREP record retrieval by entering:

   ```
   retrieve erep
   ```

7. Disconnect the EREP virtual machine by entering:

   ```
   #cp disconnect
   ```

   You may now use the display for another virtual machine.
8. If necessary, reconnect the operator's virtual machine.

Stopping EREP Record Retrieval

Some tasks require you to stop EREP record retrieval. For example, before you can process records stored on the EREP virtual machine's A disk or check how full the A disk is, you must stop retrieval.

To stop EREP record retrieval:

1. If necessary, disconnect from the operator's virtual machine by entering:

   ```
   disconnect
   ```

2. Log on the EREP virtual machine.
3. Stop EREP retrieval by entering:

   ```
   #cp external
   ```

   In response, you receive a prompt:

```
HCPRET592A ENTER END OR SUMMARY
```

Enter:

```
end
```

This stops the retrieval of EREP information. CP continues to store new EREP records in real storage until you resume EREP record retrieval.

Note: If you enter SUMMARY instead of END, CP displays the number of EREP records that have been retrieved since the last RETRIEVE command, and retrieval continues. To stop retrieval, you must enter the #CP EXTERNAL command again and respond to the prompt with END.

Disassociating a User ID from the Retrieval of EREP Records

1. Stop the recording of EREP records for this user ID. To do this from a user ID authorized for the class **A** or **B** version of the CP RECORDING command, enter:

   ```
   recording erep off qid userid
   ```

   Or, the user ID being removed (must be authorized for the class **C**, **E**, or **F** version of the recording command), enter:

   ```
   recording erep off
   ```

2. If there are EREP records queued in host storage for this user ID and you want to save the data, log on the user ID to be deleted and enter:

   ```
   retrieve erep
   ```

   This will place the EREP records in the proper file for later processing. Disconnect this user by entering:

   ```
   #cp disconnect
   ```

   Log back on the authorized user ID.

3. If the EREP records queued in host storage for this user ID are not wanted, they may be purged. To do this from a user ID authorized for the class **A** or **B** version of the CP RECORDING command, enter:

   ```
   recording erep purge qid userid
   ```

   Or, the user ID being removed (must be authorized for the class **C**, **E**, or **F** version of the recording command), enter:

   ```
   recording erep purge
   ```

   This will PURGE all the EREP records queued in host storage for this user ID.

4. Verify the record count is zero and recording is off for this user ID. To do this, enter:

   ```
   query recording
   ```

   Note: The deleted user ID remains in the warm start data and in the output of the QUERY RECORDING command until z/VM is restarted with the COLD option.

5. If the user ID is specified on the SYSTEM_USERIDS statement in the system configuration file, remove that user ID.

   Note: The user ID OPEREREP is the default if no user ID for EREP is specified in the system configuration file.

Resuming EREP Record Retrieval

Ordinarily, after you process EREP records or check on the A disk, CMS is still running in the EREP virtual machine. So, to resume EREP record retrieval, you do not have to go through the whole procedure described in "Starting EREP Record Retrieval Manually" on page 196. Instead, perform the following steps:

1. Restart EREP record retrieval by entering:

   ```
   retrieve erep
   ```

2. Disconnect the EREP virtual machine by entering:

   ```
   #cp disconnect
   ```

   You can now use the display for another virtual machine.

3. If necessary, reconnect the operator's virtual machine.

Finding Out How Full the EREP A-Disk Is

When the EREP virtual machine's A disk is full, it cannot retrieve any more records. If your installation uses the IBM-supplied PROFILE EXEC to start up the EREP virtual machine, you receive a message about the A disk's status when the virtual machine IPLs CMS. Also, CP sends you, the primary system operator, a message when the A disk is 75% full, 90% full, and completely full. Between messages, you may wish to check the disk yourself.

To see how much room there is on the EREP virtual machine's A disk:

1. Stop EREP record retrieval.

   To do this, see "Stopping EREP Record Retrieval" on page 196.

2. Enter:

   ```
   query disk a
   ```

   In response, CMS tells you how full the A disk is.

3. Resume EREP record retrieval.

   To do this, see "Resuming EREP Record Retrieval" on page 198.

Processing EREP Information

From time to time, you must process the EREP records stored on the EREP virtual machine's A-disk and make room for new ones. You may do this when you notice the A-disk is nearly full or at intervals set by your installation. In either case, you must:

1. Stop EREP record retrieval.

   To do this, see "Stopping EREP Record Retrieval" on page 196.

2. Process the EREP information.

   You can process the EREP records on the A-disk in one of several ways:

   - You can run the EREP program to format and print the records.

     For more information, see the next section.

   - You can use the CMS command, TAPE DUMP, to dump the EREP files to tape for later processing.
   - You can send the files to another user for processing.

     To send a file to another virtual machine user, first set up your virtual punch so that CP sends all punch files to the reader of the other virtual machine. Enter:

     ```
     spool punch to userid reader
     ```

where *userid* is the user ID of the virtual machine to which you are sending the file.

Then use the CMS command, DISK DUMP, to send the file to the user.

After you process the EREP records, use the CMS command, ERASE, to erase the old files. Erasing the files makes room on the A-disk for new records.

3. Resume EREP record retrieval.

   To do this, see "Resuming EREP Record Retrieval" on page 198.

Running the CPEREPXA Utility

The environmental record editing and printing (EREP) program formats and prints EREP records. On z/VM, you invoke the program through the CPEREPXA utility.

When you run the EREP program, you must specify options. For a description of the EREP options, see the *z/VM: CP Commands and Utilities Reference* and the *Environmental Record Editing and Printing Program User's Guide and Reference*.

You can enter these options in two ways, either interactively or through an options file.

Entering EREP Options Interactively

To run the EREP program and enter options interactively, enter:

```
cperepxa
```

In response, CP sends you the following prompt:

```
ENTER:
```

You can now enter CPEREPXA options according to the rules in the *z/VM: CP Commands and Utilities Reference*.

Entering EREP Options through an Options File

To run the EREP program using an options file:

1. Create the options file.

   The file type for this file can be any 8-character name that is not reserved.

   The following lists the reserved file types. Some are file types used by functions or licensed programs that z/VM does not support.

Reserved file types

$B$B$B$B
$HLPxxxx
AMSERV
ASM3705
ASSEMBLE
AUXxxxx
CMSUT1
CNTRL
COBOL
COPY
DIRECT
DOSLIB
DOSLNK
ESERV
EXEC
EXPAND
FORTRAN
FREEFORT
FTnn001
GCS
GLOBALV
GROUP
HELPxxxx
LDTFDIx
LISTING
LKEDIT
LOADLIB
MACLIB
MACRO
MAP
MEMO
MODULE
NAMES
NETLOG
NOTEBOOK
PLI
PLIOPT
PROC
RPGII
SCRIPT
SYMDMP
SYNONYM
SYSUTx
SYS00x
TESTCOB
TESTFORT
TEXT
TXTLIB
UPDATE
UPDLOG
UPDTxxxx
VSBASIC
VSBDATA
ZAP

2. Enter:

```
cperepxa filename filetype
```

where *filename* and *filetype* are the file name and file type of the options file, respectively.

CPEREPXA uses the options in the file to process the EREP records and does not prompt you for additional options.

Example—Running CPEREPXA with an Options File

The options file you create, with the name RECORD OPTIONS A1, looks like this:

```
00000 * * * TOP OF FILE * * *
00001 ACC=Y,DATE=(83125,83130),DEV=(3350),
00002 LINECT=40,PRINT=PT,SHORT=N,TYPE=CHMO
00003 * * * END OF FILE * * *
```

To run CPEREPXA with this file, enter:

```
cperepxa record options
```

Specifying How Often CP Checks for Missing Interrupts

CP periodically checks I/O devices for missing interrupts. A missing interrupt is an I/O request to which a device has not responded. When CP detects a missing interrupt for a device, CP either handles the condition itself or reflects the condition to the virtual machine requesting the I/O. When CP reflects a missing interrupt condition to a virtual machine, CP also sends messages to the primary system console to tell you that an interrupt is pending for the device. The messages tell you:

- The type and real device number of the device
- Some information that describes the last interrupt that occurred (IRB)
- The user ID of the virtual machine that is waiting for a response from the device (USERID).

When you receive these messages, you may need to reset the device. The virtual machine must then reenter the I/O request.

To find out how often CP is checking for missing interrupts, enter:

```
query mitime
```

CP's response indicates how often it checks for missing interrupts on various types of devices. The rate is expressed in minutes and seconds, as *mm:ss*. If the response says OFF for any device, CP is not checking it for missing interrupts.

Note: Missing interrupt detection is not provided for the Asynchronous Data Mover Facility.

The response to QUERY MITIME includes the following types of devices:

- **DASD** for DASD
- **GRAF** for display devices, except 328x devices
- **TAPE** for tape drives
- **UR** for unit record devices, except 3800
- **SWCH** for dynamic switching devices
- **MISC** for miscellaneous devices (3851, 328x, 3800, and any DASD defined as a mass storage subsystem device).

To change the rate at which CP is checking for missing interrupts, enter:

```
set mitime devtype mm:ss
```

where:

devtype
is DASD, TAPE, GRAF, UR, MISC, SWCH, or the real device number of one or more specific devices.

mm:ss
indicates how often, in minutes and seconds, you want CP to check the specified devices. CP rounds up the value you specify for *mm:ss* to the nearest multiple of 5 seconds.

You can also specify how often CP checks for missing interrupts for a range of devices. For example, if you want CP to check the six DASD with real device numbers 190 through 195 once every 30 seconds instead of once every 15 seconds, enter:

```
set mitime 190-195 00:30
```

If these are the only DASD attached to your system, you can enter the following command to obtain the same result:

```
set mitime dasd 00:30
```

Routing EREP Records to Another User ID

The RETRIEVE command lets you route certain error records to another user ID after a specified time interval has elapsed. Routing of error records is useful if your installation uses a centralized hardware service program.

To route error records, specify the LOGROUTE operand on the RETRIEVE command. For example, to direct error records to user SERVMACH at node CORPVM7 every 60 minutes, specify:

```
retrieve erep logroute to servmach at corpvm7 for 60
```

LOGROUTE can be specified when you manually start RETRIEVE processing or when RETRIEVE processing is automatically started. In the latter case, add the LOGROUTE operand to the RETRIEVE command in the PROFILE EXEC of the EREP machine.

RETRIEVE processing accumulates error records in a CMS file on file mode A of the EREP machine, and uses the CMS SENDFILE command to route the error records to the specified user. The destination can be either a z/VM system or a TSO system.

Accumulated error records are also sent when service alert (type A3) or serial incident (type A2) error records are retrieved.

Collecting Symptom Record Information

z/VM normally includes symptom records in every dump. When CP/ESA is loaded into storage, the CP *SYMPTOM system service can record symptom records in a CMS file separate from dumps, processor errors, or I/O (LOGREC) errors, and can send the records to authorized virtual machines for recording. Your installation can use these symptom records to identify known problems and their fixes without first reading dumps.

Your production system may have facilities to help you collect symptom record information.

How Symptom Record Recording Information Is Organized

Information about z/VM symptom record recording is organized as follows:

- The *z/VM: CP Planning and Administration* tells you how to set up symptom record recording virtual machines.
- The rest of this section tells you how to control symptom record recording on z/VM. To control the collection of symptom information, you must be able to:
  - Understand how symptom record recording works on z/VM
  - Find out the status of symptom record recording
  - Control symptom record creation
  - Control symptom record retrieval
  - Process recorded symptom information.

How Symptom Record Recording Works on z/VM

On z/VM, symptom record recording involves three steps. They are essential to complete symptom record recording.

- CP must create and record the symptom records; otherwise, system events go unrecorded.
  - CP generates symptom records when creating a hard abend, soft abend, SA dumps and VM dumps
  - For VM dumps, virtual machines can pass symptom records with guest information, which CP updates and records.
- The symptom record recording virtual machine must move the records to disk; otherwise, you cannot access them to process their information. Further, if the symptom record recording virtual machine does not retrieve the records, they remain in storage, building up as CP creates them. You may have more than one symptom record recording virtual machine. *If you perform a cold start, you lose any symptom records that are in storage.*
- Finally, you must process the records on disk; otherwise, your installation cannot use the information to identify and fix the problems.

Symptom Record Recording

CP creates symptom records when particular system events occur. By default, CP sets symptom record recording on when you bring up z/VM. You can control the setting with the RECORDING command.

Once symptom record recording is on, CP records a symptom record whenever one of the following events occurs:

- A z/VM component or IBM Licensed Product requests a VMDUMP or that a symptom record be recorded
- A CP abend dump occurs.

When one of these events occurs, CP creates and stores the symptom record. The *SYMPTOM system service collects the symptom records and notifies the authorized symptom record recording virtual machines. The symptom records remain in storage until the symptom record recording virtual machine retrieves them.

The default limit for symptom records is two. If the number of records in storage reaches that number, CP notifies you, the primary system operator. The buildup of records in storage indicates that retrieval is not active. You can change the limit with the RECORDING command.

Symptom Record Retrieval

The symptom record recording virtual machine retrieves a symptom record when all of the following are true:

- The virtual machine is defined
- It starts the RETRIEVE program
- It has room on its A disk for new symptom records.

Normally, during system generation, your installation sets up a permanent symptom record recording virtual machine. When you bring up z/VM, CP automatically logs on this virtual machine. However, you may use the RETRIEVE command to assign the retrieval of symptom records to other virtual machines if the system programmer authorizes that virtual machine in the IUCV directory control statement. See “Specifying a New Symptom Record Recording Virtual Machine” on page 205.

After CP logs it on, the permanent symptom record recording virtual machine normally starts retrieving automatically. However, you can also start symptom record retrieval manually. See “Stopping and Starting Symptom Record Recording” on page 204.

Over time, the symptom record recording virtual machine's A disk fills up with symptom records. CP sends a message to you, the primary system operator, when the A disk is 75% full, 90% full, and completely full. You can also log on the symptom record recording virtual machine and check the disk yourself. When the disk is full, you must process some of the old records and erase some files to make room for new ones.

Symptom Record Processing

When enough symptom records accumulate, you can log on the symptom record recording virtual machine and use the VIEWSYM command to identify known problems.

You can also dump the symptom record files to tape for later processing or send the symptom record files to another user for processing. For more information, see “Processing Symptom Record Information” on page 207.

Finding Out the Status of Symptom Record Recording

To check the status of symptom record creation and retrieval, enter:

```
query recording
```

In response, CP displays the status of accounting, EREP, and symptom record creation and retrieval. (For more information on accounting and error recording, see “Collecting Accounting Information” on page 183 and “Collecting Hardware Error Information” on page 190.) The format of the response is similar to this:

```
RECORDING     COUNT LMT USERID   COMMUNICATION
ACCOUNT  ON   00000 020 DISKACNT ACTIVE
EREP     ON   00000 002 EREP     ACTIVE
SYMPTOM  ON   00001 002 CPSYMP   INACTIVE
```

On the line that says SYMPTOM, the information is as follows:

- **RECORDING** is the status of symptom record creation by CP. The status may say SYMPTOM ON or SYMPTOM OFF. In this example, it says SYMPTOM ON, indicating that CP creates new symptom records in response to new events.
- **COUNT** is the number of symptom records in storage waiting for retrieval. In this example, the number is 1. Because the number is nonzero, the symptom record recording virtual machine may not be retrieving.

- **LMT** is the number of symptom records that must accumulate in storage before CP notifies you, the primary system operator. In this example, the number is 2, the default limit for symptom records.
- **USERID** is the user ID of the symptom record recording virtual machine. In this example it is CPSYMP.
- **COMMUNICATION** is the status of record retrieval by the symptom record recording virtual machine. The status may be INACTIVE, ACTIVE, or ACTIVE PENDING. In this example, the status is INACTIVE, which indicates that the symptom record recording virtual machine is not retrieving. By contrast, the accounting virtual machine and the EREP virtual machine are retrieving, as indicated by their ACTIVE status in the example.

  Persistent ACTIVE PENDING status indicates that the symptom record recording virtual machine is not retrieving properly.

If symptom record retrieval is INACTIVE or ACTIVE PENDING, or if a large number of symptom records are waiting in storage, restart retrieval manually. See "Starting Manual Retrieval of Symptom Records" on page 205.

Controlling Symptom Record Recording

After finding out its status, you may need to control various aspects of symptom record recording. For example, you may need to:

- Stop or start CP creation of symptom records
- Control the symptom record limit.

Stopping and Starting Symptom Record Recording

By default, CP creates and records symptom records when certain events occur. If you want CP to stop recording symptom records, enter:

```
recording symptom off
```

In response, the CP *SYMPTOM system service stops collecting symptom records.

When you want CP to start creating symptom records again, you must first purge or retrieve the symptom records queued, then enter:

```
recording symptom on
```

Purging Symptom Records

To purge any symptom records that are in storage, enter:

```
recording symptom purge
```

In response, CP purges any symptom records that are in storage.

Controlling the Symptom Record Limit

By default, CP lets two symptom records accumulate in storage before it notifies you that the records are not being retrieved. If you prefer, you can change this limit. For example, if you want to set the limit to 12 records, enter:

```
recording symptom limit 12
```

Controlling Symptom Record Retrieval

Once symptom records are being created, you may need to control various aspects of their retrieval. For example, you may need to:

- Specify a new symptom record recording virtual machine

- Start symptom record retrieval manually
- Stop symptom record retrieval
- Disassociate a user ID from symptom record retrieval
- Resume symptom record retrieval
- Find out how much room is on the symptom record recording virtual machine's A disk.

Specifying a New Symptom Record Recording Virtual Machine

Normally, your installation sets up a particular virtual machine to be the symptom record recording virtual machine. If you want to have additional virtual machines collect symptom records, then you must do the following:

1. Make sure the new symptom record recording virtual machine has an A disk available to receive the symptom records.
2. Make sure the system programmer authorizes the new virtual machine to receive the records in the IUCV directory statement. The directory entry of the user entering the RETRIEVE utility must have an IUCV directory control statement authorizing the virtual machine to connect to the CP system service which supports the type of record being collected. *SYMPTOM must be specified for SYMPTOM records. For more information about IUCV, see the *z/VM: CP Programming Services*.
3. Follow the steps outlined in the next section, “Starting Manual Retrieval of Symptom Records” on page 205.

Starting Manual Retrieval of Symptom Records

Ordinarily, the symptom record recording virtual machine starts retrieving automatically when you bring up z/VM. However, there may be times when you need to start retrieval manually. For example, you may find out through error messages or by entering the QUERY RECORDING command that the symptom recording virtual machine has stopped retrieving. You may also start retrieval manually, as follows:

1. If necessary, disconnect from the operator's virtual machine by entering:

   ```
   disconnect
   ```

2. Log on the symptom record recording virtual machine.
3. Clear any activity in the virtual machine (including retrieval) by entering:

   ```
   #cp system reset
   ```

4. Load CMS into storage by entering:

   ```
   ipl 190
   ```

   or, if your installation has installed CMS in a named saved system,

   ```
   ipl cmsname
   ```

   where *cmsname* is the name of your CMS system.
5. Make sure there is room on the symptom record recording virtual machine's A disk for more symptom records. To find out how full the A disk is, enter:

   ```
   query disk a
   ```

   If the A disk is almost full, process some of the symptom records. Then erase the old files to free space on the A disk.
6. Start symptom record retrieval by entering:

   ```
   retrieve symptom
   ```

7. Disconnect the symptom record recording virtual machine by entering:

```
#cp disconnect
```

You can now use the display for another virtual machine.

8. If necessary, reconnect the operator's virtual machine.

Stopping Symptom Record Retrieval

A task might require you to stop symptom record retrieval. For example, before you can process records stored on the symptom record recording virtual machine's A disk you must stop retrieval.

To stop symptom record retrieval:

1. If necessary, disconnect from the operator's virtual machine by entering:

   ```
   disconnect
   ```

2. Log on the symptom record recording virtual machine.
3. Stop symptom record retrieval by entering:

   ```
   #cp external
   ```

   Note: If the symptom record recording virtual machine enters the CMS DEBUG environment when you enter #CP EXTERNAL, enter GO to return to CMS.

   In response, you receive a prompt:

   ```
   HCPRET592A ENTER END OR SUMMARY
   ```

   Enter:

   ```
   end
   ```

   This stops the retrieval of symptom record information. CP continues to store new symptom records in real storage until you resume symptom record retrieval.

   Note: If you enter SUMMARY instead of END, CP displays the number of symptom records that have been retrieved since the last RETRIEVE command, and retrieval continues. To stop retrieval, you must enter the #CP EXTERNAL command again and respond to the prompt with END.

Disassociating a User ID from the Retrieval of Symptom Records

1. Stop the recording of SYMPTOM records for this user ID. To do this from a user ID authorized for the class **A** or **B** version of the CP RECORDING command, enter:

   ```
   recording symptom off qid userid
   ```

   Or, the user ID being removed (must be authorized for the class **C**, **E**, or **F** version of the recording command), enter:

   ```
   recording symptom off
   ```

2. If there are symptom records queued in host storage for this user ID and you want to save the data, log on the user ID to be deleted and enter:

   ```
   retrieve symptom
   ```

   This will place the symptom records in the proper file for later processing. Disconnect this user by entering:

   ```
   #cp disconnect
   ```

   Log back on the authorized user ID.

3. If the symptom records queued in host storage for this user ID are not wanted, they may be purged. To do this from a user ID authorized for the class **A** or **B** version of the CP RECORDING command, enter:

   ```
   recording symptom purge qid userid
   ```

   Or, the user ID being removed (must be authorized for the class **C**, **E**, or **F** version of the recording command), enter:

   ```
   recording symptom purge
   ```

   This will PURGE all the symptom records queued in host storage for this user ID.
4. Verify the record count is zero and recording is off for this user ID. To do this, enter:

   ```
   query recording
   ```

   Note: The deleted user ID remains in the warm start data and in the output of the QUERY RECORDING command until z/VM is restarted with the COLD option.
5. If the user ID is specified on the SYSTEM_USERIDS statement in the system configuration file, remove that user ID.

   Note: The user ID OPERSYMP is the default if no user ID for SYMPTOM is specified in the system configuration file.

Resuming Symptom Record Retrieval

Ordinarily, after you have stopped symptom record retrieval to view symptom records or to check on the A disk, CMS is still running in the symptom record recording virtual machine. So, to resume symptom record retrieval, you do not have to go through the whole procedure described in “Starting Manual Retrieval of Symptom Records” on page 205. Instead, perform the following steps:

1. Restart symptom record retrieval by entering:

   ```
   retrieve symptom
   ```

2. Disconnect the symptom record recording virtual machine by entering:

   ```
   #cp disconnect
   ```

   You may now use the display for another virtual machine.
3. If necessary, reconnect the operator's virtual machine.

Finding Out How Full the Symptom Record A-Disk Is

When the symptom record recording virtual machine's A disk is full, it cannot retrieve any more records. If your installation uses the IBM-supplied PROFILE EXEC to start up the symptom record recording virtual machine, you, the primary system operator, receive a message about the A disk's status when the virtual machine IPLs CMS. CP sends you a message when the A disk is 75% full, 90% full, and completely full. Between messages, you may wish to check the disk yourself.

To view the symptom records, you would use the LINK command so that a second virtual machine could access or link to the symptom record recording virtual machine's A disk. You would then be able to view the symptom records without interrupting the symptom record retrieval process.

Processing Symptom Record Information

From time to time, you must process the symptom records stored on the symptom record recording virtual machine's A-disk and make room for new ones. You can do this when you notice the A-disk is nearly full or at intervals set by your installation. In either case, you must:

1. Stop symptom record retrieval.

To do this, see "Stopping Symptom Record Retrieval" on page 206.

2. Process the symptom record information.

   You can process the symptom records on the A-disk in one of several ways:

   - You can run a program to format the symptom records if such a program is available.
   - You can use the VIEWSYM command to view selected symptom records. For more information, see the *z/VM: Dump Viewing Facility*.
   - You can use the CMS command, TAPE DUMP, to dump the symptom record files to tape for later processing.
   - You can send the files to another user for processing.

     To send a file to another virtual machine user, first set up your virtual punch so that CP sends all punch files to the reader of the other virtual machine. Enter:

     ```
     spool punch to userid reader
     ```

     where *userid* is the user ID of the virtual machine to which you are sending the file.

     Then use the CMS command, DISK DUMP, to send the file to the user.

   After you process the symptom records, use the CMS command, ERASE, to erase the old files. Erasing the files makes room on the A-disk for new records.

3. Resume symptom record retrieval.

   To do this, see "Resuming Symptom Record Retrieval" on page 207.

Summary of Collecting Information about System Operation

As the real system operator, you are responsible for collecting and providing reports of system events, system performance, accounting, error recording, and symptom record recording information.

You provide these reports so that your installation can:

- Fix problems that prevent users from using installation resources
- Identify and fix known problems
- Allocate resources among users in the best way possible
- Plan for future resource requirements.

To collect information about the system's operation, you may use either z/VM facilities or facilities provided by your production system.

You can use z/VM facilities to collect information as follows:

- To collect information about system events, start a console log for your operator's virtual machine.
- To collect information about performance, enter one of the following commands:
  - INDICATE LOAD
  - INDICATE LOAD AT *sysname*
  - INDICATE LOAD AT *
  - INDICATE I/O
  - INDICATE PAGING WAIT
  - INDICATE QUEUES
  - INDICATE USER
- To collect information about accounting, use the following commands:
  - QUERY RECORDING

  - RECORDING ACCOUNT
  - ACNT
  - RETRIEVE ACCOUNT

- To collect information about hardware errors, use the following commands:
  - QUERY RECORDING
  - RECORDING EREP
  - SET MODE RETRY
  - SET RECORD
  - RETRIEVE EREP
  - CPEREPXA
  - QUERY MITIME
  - SET MITIME

- To collect information about symptom records, use the following commands:
  - QUERY RECORDING
  - RECORDING SYMPTOM
  - RETRIEVE SYMPTOM

Chapter 10. Performing Data Management Services

This chapter discusses z/VM facilities you can use to perform data management services. After reading this chapter, you can:

- Use the major data management services
- Understand the importance of data management
- Locate information about z/VM facilities you can use for data management
- Use the SPXTAPE command to back up standard spool files and system data files
- Use the SPXTAPE command to move files between supported z/VM releases, if necessary.
- Locate information about production system facilities used for data management.

Overview of Data Management Services

As the real system operator, your job is to make the real resources of your computer installation available to users so that they can get their work done. A resource that is very important to users is their data. A significant part of your job, therefore, is providing services to users to help them manage it. These data management services include:

- Controlling user access to data
- Backing up system and user data
- Moving data between systems
- Archiving user data
- Initializing data storage media.

You can use z/VM facilities to help you perform these services. Your production system may also have facilities that can help you manage your installation's data.

Controlling User Access to Data

Part of operating a z/VM system is controlling user access to data. Data is usually considered to be a vital asset. The responsibility for safeguarding it from unauthorized persons rests partly on operations.

As an operator you may have to:

- Control how users enter passwords
- Keep track of tapes and disk packs.

Controlling the Display of Passwords

When a user enters a LOGON command to log on a virtual machine, CP prompts the user for the password. CP masks the password as it is entered, preventing an unauthorized person from reading it on the display screen.

Note: Passwords at your installation may be controlled by an external security manager, such as RACF.

If your installation does not require logon passwords to be kept confidential, you can allow users to enter the password on the same line as the LOGON command. To do this, enter:

```
set password logon include
```

To reset the system to require the more secure procedure, enter:

```
set password logon separate
```

When a user enters an AUTOLOG or XAUTOLOG commands to log on to the system automatically, CP prompts the user for the autolog password. CP masks the password as it is entered, preventing an unauthorized person from reading it on the display screen.

If your installation does not require autolog passwords to be kept confidential, you can specify that users must enter the password on the same line as the AUTOLOG or XAUTOLOG commands. To do this, enter:

```
set password autolog include
```

To reset the system to require the more secure procedure, enter:

```
set password autolog separate
```

z/VM users can share the data they have on minidisks by using the LINK command. z/VM enables your installation to choose whether a user may enter a link password on the input line with the LINK command.

When you first bring up the z/VM system, CP requires all users to enter the link password on a separate line from the LINK command. When a user enters a LINK command, CP prompts the user for the link password. CP masks the password as it is entered, preventing an unauthorized person from reading it on the display screen.

If your installation does not require link passwords to be kept confidential, you can allow users to enter the link password on the same line as the LINK command. To do this, enter:

```
set password link include
```

To reset the system to require the more secure procedure, enter:

```
set password link separate
```

Controlling the Journaling of Incorrect Password Use

When a user enters a LOGON command or a LINK command, CP prompts the user for the password. You can begin or end the creation of accounting records that are generated each time a user enters an incorrect password. To collect accounting records and receive a message when a user enters an incorrect LINK password, enter:

```
set journal link on
```

To halt record creation, enter:

```
set journal link off
```

To collect accounting records and receive a message when a user enters an incorrect LOGON, AUTOLOG, or XAUTOLOG password, enter:

```
set journal logon on
```

To halt record creation, enter:

```
set journal logon off
```

To determine whether CP is generating accounting records, enter:

```
query journal
```

For more information on the SET JOURNAL and QUERY JOURNAL commands, see the *z/VM: CP Commands and Utilities Reference*.

Clearing Temporary Disk (TDISK) Space

You may want to prevent a user from accessing sensitive data remaining on a temporary disk from a previous user. To request that all temporary disks be cleared, specify the ENABLE CLEAR_TDISK operand on the FEATURES statement in the system configuration file. This causes all temporary disk space to be cleared:

- At IPL
- When CP volumes containing TDISK space are attached to the system
- When a user detaches a TDISK

To determine whether CP is clearing TDISK space, enter:

```
query tdiskclr
```

For more information on the QUERY TDISKCLR command, see *z/VM: CP Commands and Utilities Reference*.

Keeping Track of Tapes and Disk Packs

Most installations have a library where they keep tapes and disk packs. Depending on the size of the installation, the responsibilities of the operator who serves as the librarian can be substantial.

In a large installation, a librarian may have to:

- Label tapes and disk packs
- Control the physical location of the tapes and disk packs
- Inventory tapes and disk packs
- Replace defective tapes and packs
- Maintain an adequate supply of scratch tapes and disk packs
- Maintain lists of authorized users
- Follow special security procedures for sensitive tapes and packs
- Conduct periodic audits.

Backing Up System and User Data

Backing up system and user data is another important part of operating a z/VM system. Data is usually considered to be a vital asset. The responsibility for keeping backup copies in case originals are inadvertently destroyed rests partly on operations.

As an operator you may have to back up:

- Data that resides on disk or tape
- Spool files and system data files.

Backing Up Data on Disk or Tape

To back up data on disk or tape, you may have to:

- Copy data from one disk to another
- Dump data from disk to tape
- Restore data from tape to disk
- Copy data from one tape to another.

Copying Data from One Disk to Another

If you need to copy data from one disk to another, use the DASD Dump Restore program that IBM provides with z/VM. For more information, see "Running the DASD Dump Restore Utility (DDRXA)" on page 238.

Dumping Data from Disk to Tape

If you need to dump all or part of the data on DASD to a tape, use the DASD dump restore utility that IBM provides with z/VM. For more information, see "Running the DASD Dump Restore Utility (DDRXA)" on page 238.

If you want to dump one or more CMS files to tape, use the CMS TAPE DUMP command, or the vmfplc2 command.

For example, if you want to dump to tape all of the files on your A disk with a file type of ACCOUNT, and:

- You want CMS to write a tape mark after each file
- You want a printout of the names of the files that were dumped
- The tape drive dedicated to your virtual machine is assigned virtual device number 184. Enter:

```
tape dump * account a (wtm print tap4
```

The asterisk indicates that you want to dump all of the files on your A disk with the file type of ACCOUNT. If you want to dump only a specific file, specify the file's name instead of the asterisk.

Note: The records in these CMS files can be of either fixed or variable length. However, none of the files can be larger than one DASD volume.

Restoring Data to Disk from Tape

If you need to restore data to DASD from a tape, and you used the DASD dump restore utility to create the tape, use the same utility to restore it. For more information, see "Running the DASD Dump Restore Utility (DDRXA)" on page 238.

If you want to load one or more CMS files from a tape onto a CMS minidisk, and you used the CMS TAPE DUMP command to dump them, use the CMS TAPE LOAD command to load them.

For example, if you want to load some CMS files onto your K disk that were previously dumped to tape, and:

- You want CMS to display the names of all of the files that were loaded
- The tape drive on which the tape is mounted is assigned the virtual device number 183. Enter:

```
tape load * * k (term tap3
```

The asterisks indicate that you want CMS to load all of the files. If you want CMS to load only a specific file, specify the file's name and type instead of the asterisks.

Copying Tapes

If you need to copy data from one tape to another, use the DASD dump restore utility that IBM provides with z/VM For more information, see "Running the DASD Dump Restore Utility (DDRXA)" on page 238.

Backing Up Standard Spool Files and System Data Files

Much of your installation's data resides on disk, a significant portion of it in the spooling system as standard spool files and system data files. Standard spool files include files in the reader, printer, and punch queues. System data files include image libraries, named saved systems, saved segments, national language support files (such as message repository files), system trace files, and UCR files. Although UCR files might still exist, they are no longer supported or processed by CP.

You may want to back up standard spool files under one of the following circumstances:

- When you plan to perform a clean start or cold start (because all standard spool files are lost)
- When CP runs out of either spooling space or spool file identification numbers to assign to new spool files.

In either of these situations, you can dump some or all of your system's standard spool files to tape. Then, after you bring the system back up, or when you have less spooling activity, you can restore the files for processing as usual.

You can do the same thing with system data files. z/VM does not save system data files over a clean start. Although z/VM does save system data files over a cold start, it is still a good idea to make backup copies of them. Although you may not have many of these files, it might take users or support personnel a long time to recreate them. Having a copy of the files on tape allows you to restore them quickly in case of a DASD or other unexpected error.

To accomplish either of these tasks, use the SPXTAPE command. It allows you to dump standard spool files and system data files to tape and restore them.

Dumping Standard Spool Files to Tape

Use the SPXTAPE DUMP command to save copies of standard spool files on tape. SPXTAPE processes only closed reader, printer, and punch files; open or in-use files cannot be dumped. By default, the files are dumped regardless of their hold status.

To dump all the standard spool files to tape:

1. Find one or more free tape drives (it is recommended that you use at least two drives to allow overlapping tape mounting) and mount scratch tapes on them.

   A free tape drive is one that is not offline, in use, or dedicated to another virtual machine. The tape drive must be fully supported by z/VM. For a list of supported devices, see the *z/VM: General Information*. To find out the status of all of your tape drives, enter:

   ```
   query tapes all
   ```

   If the status of any tape drive is FREE, mount the scratch tape, which has been enabled for writing, and go to the next step. To write-enable a tape reel, make sure the tape has a write-enable ring. To write-enable a tape cartridge, turn the thumbwheel on the cartridge so the white dot does not show.

   If the status of any tape drive is OFFLINE, enter:

   ```
   vary online rdev
   ```

   where *rdev* is the tape drive's real device number. Then mount the scratch tape, enabled for writing, and go to the next step.

2. Attach each tape drive to your virtual machine. Enter:

   ```
   attach rdev to * as vdev
   ```

   where *vdev* is an available virtual device number.

3. Spool your console to wherever you want the SPXTAPE log files to go. For example, spool the console to yourself so the logs are sent to your reader. Enter:

   ```
   spool cons *
   ```

   Note: If you spool your console *after* entering the SPXTAPE command, the logs may be sent to different destinations.

4. If necessary, use the CMS TAPE command to position each tape to where CP can begin dumping files. The tape must be positioned either at the beginning of the volume or following a tape mark.

5. Dump all the standard spool files to tape. Enter:

   ```
   spxtape dump vdev1-vdev2 std all run
   ```

   where *vdev1-vdev2* is the range of virtual device numbers of the tape drives you are using. If you are using only one drive, you can enter a single virtual device number. The RUN operand rewinds and unloads the tape when processing to that volume completes.

In response, CP dumps a copy of each closed reader, printer, and punch spool file to tape without removing the files from their queues on the system. All the available tape drives in the specified range are used concurrently; while one file is still being written to a tape, the next file is written to the next available tape. As it processes the files, CP reports its progress at the interval requested with the PROGress_interval option (or about every 15 seconds by default) by displaying a message on the screen.

Note: If CP is running out of either spooling space or spool file identification numbers to assign to new files, you may wish to have CP purge each file after dumping the file to tape. If you do, use the PURGE operand as follows:

```
spxtape dump vdev1-vdev2 std all run purge
```

CP responds as before, except it removes each dumped file from the queue. CP also tells the file owner (you or another user) it has purged the file.

CP records information about the dump in two or more logs. The command summary log contains information about the progress and status of the logical SPXTAPE command. A logical SPXTAPE command can be either one independent command or a sequence of appended commands (entered with the APPEND operand). The volume log contains information about the files processed by the logical SPXTAPE command that are associated with a particular tape volume. Therefore, several volume logs are created for one logical SPXTAPE command if more than one tape drive is used or if more than one tape volume is mounted on any drive.

If any tape is filled before all the files are dumped, CP sends you a response that:

a. Asks you to mount the next tape on that device
b. Indicates the number of files and spool pages processed so far.

If you are using more than one tape drive, CP continues dumping to the other drives. When you mount and ready the new tape, CP starts dumping to that drive again. When CP has dumped all the files, it tells you the dump function is complete.

6. Remove the tapes from the tape drives.
7. If you want to find out what files have been dumped, you can look at the volume logs. Each volume log has a unique name (file name and file type) related to both the tape volume and the SPXTAPE command that caused the creation of the log. For example, the volume log for the first volume processed on virtual device 181 for an SPXTAPE DUMP command has the name:

```
xxyyD181 hhmmss01
```

where *xxyy* is the month and day the command was issued, D indicates the operation was DUMP, 181 is the virtual device number of the tape drive, *hhmmss* is the time stamp, and 01 is the volume sequence number.

The volume log contains the following entry for each file written to the volume: Some of the entries have been truncated to fit the page.

```
USERID FILE QUEUE FILNAM FILTYP OPNDAT OPNTME ORIGID SZ SEG_STAT HLDSTAT RCCOUNT C FORM DEST
LSTPGE
userid file queue filnam filtyp opndat opntme origid sz COMPLETEhldstat rccount c form dest
lstpge
```

Note: The heading is included only once for each uninterrupted series of entries.

If the SEG_STAT field contains PARTIAL instead of COMPLETE, the file is split between this volume and one or more other volumes, and this is *not* the part that completes the file (the last part dumped). If all parts of the file have dumped correctly, the volume log entry for the last part dumped contains COMPLETE in the SEG_STAT field.

The preceding steps tell you how to dump all the standard spool files to tape. You can also select the spool files to be dumped by queue, user ID, spool file identification number or range, class, destination value, form name, file name pattern, file type pattern, hold status, SSI copy status, or combinations of these attributes. A file name or file type pattern can be either a complete file name or file type or a string containing wild cards (* and %). The following example shows you how to dump particular classes of files from the reader and printer queues.

For more information about the SPXTAPE logs and the options you can specify on the SPXTAPE DUMP command, see the *z/VM: CP Commands and Utilities Reference*.

Example—Dumping Particular Classes of Reader and Printer Spool Files to Tape

You are planning to shut down the system and bring up a new maintenance level of z/VM with a cold start. Your installation decides to save the class S files in the reader queue and the class T files in the printer queue and restore them after the cold start. Because there is no shortage of spooling space, you do not have to purge any files from the system.

To dump only the class S reader files and class T printer files to tape:

1. Find one or more free tape drives. Enter:

   ```
   query tapes all
   ```

   CP's response tells you the status of tape drives 5C0 and 61B is FREE. Both are 3490s, so you mount a scratch tape on each.
2. Attach the tape drives to your virtual machine. Virtual device numbers 181 and 182 are available, so you enter:

   ```
   attach 5c0 to * as 181
   attach 61B to * as 182
   ```
3. Spool your console. To spool the console to yourself, enter:

   ```
   spool cons *
   ```
4. If necessary, use the CMS TAPE command to position each tape to the beginning of the volume or following a tape mark.
5. Start the dump. Enter:

   ```
   spxtape dump 181-182 reader class s append
   spxtape dump 181-182 printer class t run
   ```

   Specifying the APPEND operand on the first command combines these two commands into a single logical command. CP begins processing the files as soon as you enter the first command in the sequence. CP displays progress responses, requests to mount new tapes (if necessary), and a final response when the dump function is complete. Each file dumped is recorded in the volume log for that tape volume.
6. Remove the tapes from the tape drives.

Restoring Standard Spool Files from Tape

Use the SPXTAPE LOAD command to restore copies of z/VM standard spool files that were previously saved on tape by SPXTAPE DUMP.

To restore all the standard spool files from tape:

1. Find one or more free tape drives (it is recommended that you use at least two drives to allow overlapping tape mounting) and mount the tapes that hold the spool files to be loaded.

   The order in which you mount the tapes does not matter, as long as the files were dumped with one logical SPXTAPE command and you use one logical SPXTAPE command to load them. A logical SPXTAPE command can be either one independent command or a sequence of appended commands (entered with the APPEND operand). If a file is split across two or more tape volumes, CP fits the parts together correctly.

A free tape drive is one that is not offline, in use, or dedicated to another virtual machine. The tape drive must be fully supported by z/VM. For a list of supported devices, see the *z/VM: General Information*. To find out the status of all of your tape drives, enter:

```
query tapes all
```

If the status of any tape drive is FREE, mount a dump tape, which has been enabled for reading only, and go to the next step. To enable a tape reel as read-only, make sure the tape does not have a write-enable ring. To enable a tape cartridge as read-only, turn the thumbwheel on the cartridge so the white dot shows.

If the status of any tape drive is OFFLINE, enter:

```
vary online rdev
```

where *rdev* is the tape drive's real device number. Then mount the tape, enabled for reading only, and go to the next step.

2. Attach each tape drive to your virtual machine. Enter:

   ```
   attach rdev to * as vdev
   ```

   where *vdev* is an available virtual device number.
3. Spool your console to wherever you want the SPXTAPE log files to go. For example, spool the console to yourself so the logs are sent to your reader. Enter:

   ```
   spool cons *
   ```

   Note: If you spool your console *after* entering the SPXTAPE command, the logs may be sent to different destinations.
4. If necessary, use the CMS TAPE command to position each tape to where CP can begin reading data. The tape must be positioned either at the beginning of the volume or at the beginning of a tape file (following the tape mark at the end of the previous tape file).
5. If you want to find out what standard spool files are on tape before you do the load, and the volume logs for the dump are not available, you can use the SPXTAPE SCAN command.

   a. To scan the tape, enter:

      ```
      spxtape scan vdev1-vdev2 std all run
      ```

      where *vdev1-vdev2* is the range of virtual device numbers for the tape drives you are using. If you are using only one drive, you can enter a single virtual device number. The RUN operand rewinds and unloads the tape when processing to that volume completes.

      Note: If you have only one tape to scan on each tape drive, you may prefer to use the REWIND operand, which rewinds the tape but does not unload it from the drive. If the tape contains multiple tape files dumped by multiple logical SPXTAPE DUMP commands, you must use multiple logical SPXTAPE SCAN commands to scan them. In that case, you may want to use the LEAVE operand, which leaves the tape positioned to process the next tape file.

      In response, CP scans each reader, printer, and punch file in the selected tape file on each tape. As it processes the files, CP reports its progress at the interval requested with the PROGress_interval option (or about every 15 seconds by default) by displaying a message on the screen.

      CP records information about the scan in two or more logs. The command summary log contains information about the progress and status of the logical SPXTAPE command. The volume log contains information about the files processed by the logical SPXTAPE command that are associated with a particular tape volume. Therefore, several volume logs are created for one logical SPXTAPE command if more than one tape drive is used or if more than one tape volume is mounted on any drive.

   b. If you entered the SPXTAPE SCAN command with the RUN disposition, CP sends you a response when processing of a tape volume is complete that:

i) Asks you to mount the next tape on that device

ii) Indicates the number of files and spool pages processed so far.

If there are other tapes to be processed, mount the next tape. If you are using more than one tape drive, CP continues scanning on the other drives. When you mount and ready the new tape, CP starts scanning on that drive again.

If there are no other tapes to be mounted, and all tapes currently mounted have finished processing (you have received the "mount next tape" response for each drive), use the SPXTAPE END command to end the operation. Enter:

```
spxtape end vdev1-vdev2
```

where *vdev1-vdev2* is same virtual device range you specified on the SPXTAPE SCAN command. The tape is positioned according to the disposition specified on the SPXTAPE SCAN command.

c. To see the results of the scan, look at the volume logs. Each volume log has a unique name (file name and file type) related to both the tape volume and the SPXTAPE command that caused the creation of the log. For example, the volume log for the second volume processed on virtual device 181 for an SPXTAPE SCAN command has the name:

```
xxyyS181 hhmmss02
```

where *xxyy* is the month and day the command was issued, S indicates the operation was SCAN, 181 is the virtual device number of the tape drive, *hhmmss* is the time stamp, and 02 is the volume sequence number.

The volume log contains the following entry for each file scanned on the volume: Some of the entries have been truncated to fit the page.

```
USERID FILE QUEUE FLNAM FLTYP OPNDAT OPNTIM ORIGID SZ SEG_ST HLDSTAT RCOUNT C FORM DEST
LSTPAGE
userid file queue flnam fltyp opndat opntim origid sz COMPLThldstat rccount c form dest
lstpage
```

Note: The heading is included only once for each uninterrupted series of entries.

If the SEG_STAT field contains PARTIAL instead of COMPLETE, it means one of the following:

- The file is split between this volume and one or more other volumes, and this part is *not* the one that completes the file (the last part scanned). If all parts of the file have scanned correctly, the volume log entry for the last part scanned contains COMPLETE in the SEG_STAT field.

  Note: The last part scanned may contain the end of the file, the beginning, or some section in between, depending on the order in which the tape volumes were mounted.

- CP encountered an I/O error or some other problem trying to read the file from the tape. In that case, you also receive error messages describing the problem. The error messages are recorded in the command summary log.

6. If necessary, use the CMS TAPE command to position each tape to the beginning of the tape file you want to load.

7. Load all the standard spool files. Enter:

```
spxtape load vdev1-vdev2 std all run
```

where *vdev1-vdev2* is the range of virtual device numbers of the tape drives you are using. If you are using only one drive, you can enter a single virtual device number. The RUN operand rewinds and unloads the tape when processing of that volume completes.

Note: If the tape contains tape files dumped by multiple logical SPXTAPE DUMP commands, you must use multiple logical SPXTAPE LOAD commands to load them. In that case, you may prefer to use the LEAVE operand, which leaves the tape positioned to process the next tape file.

In response, CP reads each reader, printer, and punch file from the selected tape file on each tape and adds it to the appropriate spooling system queue. All the available tape drives in the specified range are used concurrently; while one file is still being loaded from a tape, the next file is read from the next available tape. As it processes the files, CP reports its progress at the interval requested with the PROGress_interval option (or about every 15 seconds by default) by displaying a message on the screen. As it loads each file, CP assigns a new spool file identification number to it. CP also sends a response to the owner of the file (if the owner is logged on and has IMSG set ON).

CP records information about the load in two or more logs. The command summary log contains information about the progress and status of the logical SPXTAPE command. The volume log contains information about the files processed by the logical SPXTAPE command that are associated with a particular tape volume. Therefore, several volume logs are created for one logical SPXTAPE command if more than one tape drive is used or if more than one tape volume is mounted on any drive.

8. If you entered the SPXTAPE LOAD command with the RUN disposition, CP sends you a response when processing of a tape volume is complete that:

   a. Asks you to mount the next tape on that device
   b. Indicates the number of files and spool pages processed so far.

   If there are other tapes to be processed, mount the next tape. If you are using more than one tape drive, CP continues loading from the other drives. When you mount and ready the new tape, CP starts loading from that drive again.

   If there are no other tapes to be mounted, and all tapes currently mounted have finished processing (you have received the "mount next tape" response for each drive), use the SPXTAPE END command to end the operation. Enter:

   ```
   spxtape end vdev1-vdev2
   ```

   where *vdev1-vdev2* is same virtual device range you specified on the SPXTAPE LOAD command. The tape is positioned according to the disposition specified on the SPXTAPE LOAD command.
9. Remove the tapes from the tape drives.
10. If you want to find out what files have been loaded, you can look at the volume logs. Each volume log has a unique name (file name and file type) related to both the tape volume and the SPXTAPE command that caused the creation of the log. For example, the volume log for the second volume processed on virtual device 182 for an SPXTAPE LOAD command has the name:

    ```
    xxyyL182 hhmmss02
    ```

    where *xxyy* is the month and day the command was issued, L indicates the operation was LOAD, 182 is the virtual device number of the tape drive, *hhmmss* is the time stamp, and 02 is the volume sequence number.

    The volume log contains the following entry for each file loaded from the volume: Some of the entries have been truncated to fit the page.

    ```
    USERID FILE QUEUE FLNAM FLTYP OPNDAT OPNTIM ORIGID SZ SEG_ST HLDSTAT RECCNT C FORM DEST
    LSTPG
    userid file queue flnam fltyp opndat opntim origid sz CMPLETEhldstat reccnt c form dest
    lstpag
    ```

    Note: The heading is included only once for each uninterrupted series of entries.

    If the SEG_STAT field contains PARTIAL instead of COMPLETE, it means of one the following:

    - The file is split between this volume and one or more other volumes, and this part is *not* the one that completes the file (the last part loaded). If all parts of the file have loaded correctly, the volume log entry for the last part loaded contains COMPLETE in the SEG_STAT field.

      Note: The last part loaded may contain the end of the file, the beginning, or some section in between, depending on the order in which the tape volumes were mounted.
    - CP encountered an I/O error or some other problem trying to read the file from the tape or write it to DASD. In that case, you also receive error messages describing the problem. The error messages are recorded in the command summary log.

The preceding steps tell you how to restore all the standard spool files dumped to tape. You can also select the spool files to be loaded by queue, user ID, spool file identification number or range, class, destination value, form name, file name pattern, file type pattern, hold status, or combinations of these attributes. A file name or file type pattern can be either a complete file name or file type or a string containing wild cards (* and %). The following example shows you how to restore a particular class of files to the reader and printer queues.

For more information about the SPXTAPE logs and the other options you can specify on the SPXTAPE LOAD command, see the *z/VM: CP Commands and Utilities Reference*.

Example—Loading a Particular Class of Reader and Printer Spool Files from Tape

You bring up a new maintenance level of z/VM with a cold start. Before you shut down the z/VM system, you dumped all the standard spool files to tape. But your installation decides to restore only the class A reader files and printer files.

To load only the class A reader and printer files from tape:

1. Find one or more free tape drives. Enter:

   ```
   query tapes all
   ```

   CP’s response tells you the status of tape drives 5C0 and 61B is FREE. Both are 3490s, so you mount a dump tape on each.
2. Attach the tape drives to your virtual machine. Virtual device numbers 181 and 182 are available, so you enter:

   ```
   attach 5c0 to * as 181
   attach 61b to * as 182
   ```

3. Spool your console. To spool the console to yourself, enter:

   ```
   spool cons *
   ```

4. If necessary, use the CMS TAPE command to position each tape to the beginning of the tape file that contains the files you want to load.
5. Start the load. Enter:

   ```
   spxtape load 181-182 reader class a append
   spxtape load 181-182 printer class a run
   ```

   Specifying the APPEND operand on the first command combines these two commands into a single logical command. CP begins processing the files after you enter the last command in the sequence. CP displays progress responses and requests to mount new tapes. Each file loaded is recorded in the volume log for that tape volume.

   Note: The previous procedure assumes the files were saved on tape using one logical SPXTAPE DUMP command. If independent SPXTAPE DUMP commands were used to save the files (for example, if they were saved at different times), you must use independent (not appended) SPXTAPE LOAD commands to restore them. Wait for the first command to finish processing before you enter the second command. The disposition operand (RUN, REWIND, or LEAVE) you use on each command depends on whether the files are contained on one tape volume or multiple tape volumes.
6. When there are no more tapes to be processed (all tapes currently mounted have finished processing and there are no more tapes to be mounted), end the SPXTAPE LOAD operation. Enter:

   ```
   spxtape end 181-182
   ```

7. Remove the tapes from the tape drives.

Dumping System Data Files to Tape

Use the SPXTAPE DUMP command to save copies of system data files on tape.

To dump all the system data files to tape:

1. Find one or more free tape drives (it is recommended that you use at least two drives to allow overlapping tape mounting) and mount scratch tapes on them.

   A free tape drive is one that is not offline, in use, or dedicated to another virtual machine. The tape drive must be fully supported by z/VM. For a list of supported devices, see the *z/VM: General Information*. To find out the status of all of your tape drives, enter:

   ```
   query tapes all
   ```

   If the status of any tape drive is FREE, mount the scratch tape, which has been enabled for writing, and go to the next step. To write-enable a tape reel, make sure the tape has a write-enable ring. To write-enable a tape cartridge, turn the thumbwheel on the cartridge so the white dot does not show.

   If the status of any tape drive is OFFLINE, enter:

   ```
   vary online rdev
   ```

   where *rdev* is the tape drive's real device number. Then mount the scratch tape, enabled for writing, and go to the next step.
2. Attach each tape drive to your virtual machine. Enter:

   ```
   attach rdev to * as vdev
   ```

   where *vdev* is an available virtual device number.
3. Spool your console to wherever you want the SPXTAPE log files to go. For example, spool the console to yourself so the logs are sent to your reader. Enter:

   ```
   spool cons *
   ```

   Note: If you spool your console *after* entering the SPXTAPE command, the logs may be sent to different destinations.
4. If necessary, use the CMS TAPE command to position each tape to where CP can begin dumping files. The tape must be positioned either at the beginning of the volume or following a tape mark.
5. Dump all the system data files to tape: Enter:

   ```
   spxtape dump vdev1-vdev2 sdf all run
   ```

   where *vdev1-vdev2* is the range of virtual device numbers of the tape drives you are using. If you are using only one drive, you can enter a single virtual device number. The RUN operand rewinds and unloads the tape when processing to that volume completes.

   In response, CP dumps a copy of each image library, named saved system, saved segment, national language support file, and system trace file to tape without removing the files from their queues on the system. All the available tape drives in the specified range are used concurrently; while one file is still being written to a tape, the next file is written to the next available tape. As it processes the files, CP reports its progress at the interval requested with the PROGress_interval option (or about every 15 seconds by default) by displaying a message on the screen.

   CP records information about the dump in two or more logs. The command summary log contains information about the progress and status of the logical SPXTAPE command. A logical SPXTAPE command can be either one independent command or a sequence of appended commands (entered with the APPEND operand). The volume log contains information about the files processed by the logical SPXTAPE command that are associated with a particular tape volume. Therefore, several volume logs are created for one logical SPXTAPE command if more than one tape drive is used or if more than one tape volume is mounted on any drive.

   If any tape is filled before all the files are dumped, CP sends you a response that:

   a. Asks you to mount the next tape on that device
   b. Indicates the number of files and spool pages processed so far.

If you are using more than one tape drive, CP continues dumping to the other drives. When you mount and ready the new tape, CP starts dumping to that drive again. When CP has dumped all the files, it tells you the dump function is complete.

6. Remove the tapes from the tape drives.
7. If you want to find out what files have been dumped, you can look at the volume logs. Each volume log has a unique name (file name and file type) related to both the tape volume and the SPXTAPE command that caused the creation of the log. For example, the volume log for the first volume processed on virtual device 181 for an SPXTAPE DUMP command has the name:

   ```
   xxyyD181 hhmmss01
   ```

   where *xxyy* is the month and day the command was issued, D indicates the operation was DUMP, 181 is the virtual device number of the tape drive, *hhmmss* is the time stamp, and 01 is the volume sequence number.

   The volume log contains the following entry for each file written to the volume: some of the entries have been truncated to fit the page.

   ```
   USERID FILE QUEUE FLNAM FLTYP OPNDAT OPNTIM ORIGID SZ SEG_ST HLDSTAT RECCNT C FORM DEST LSTPG
   userid file queue flnam fltyp opndat opntim origid sz CMPLETEhldstat reccnt c form dest
   lstpag
   ```

   Note: The heading is included only once for each uninterrupted series of entries.

   If the SEG_STAT field contains PARTIAL instead of COMPLETE, the file is split between this volume and one or more other volumes, and this is *not* the part that completes the file (the last part dumped). If all parts of the file have dumped correctly, the volume log entry for the last part dumped contains COMPLETE in the SEG_STAT field.

The preceding steps tell you how to dump all the system data files to tape. You can also select the system data files to be dumped by queue, user ID (system trace files only), spool file identification number or range, class, file name pattern, file type pattern, or combinations of these attributes. A file name or file type pattern can be either a complete file name or file type or a string containing wild cards (* and %). The following example shows you how to dump a particular class of image libraries and national language support files.

For more information about the SPXTAPE logs and the options you can specify on the SPXTAPE DUMP command, see the *z/VM: CP Commands and Utilities Reference*.

Example—Dumping a Class of Image Libraries and National Language Support Files to Tape

Your installation decides to save, for backup purposes, all the active (class A) image libraries and national language support files.

To dump only the class A image libraries and national language support files to tape:

1. Find one or more free tape drives. Enter:

   ```
   query tapes all
   ```

   CP's response tells you the status of tape drives 5C0 and 61B is FREE. Both are 3490s, so you mount a scratch tape on each.
2. Attach the tape drives to your virtual machine. Virtual device numbers 181 and 182 are available, so you enter:

   ```
   attach 5c0 to * as 181
   attach 61b to * as 182
   ```

3. Spool your console. To spool the console to yourself, enter:

   ```
   spool cons *
   ```

4. If necessary, use the CMS TAPE command to position each tape to the beginning of the volume or following a tape mark.
5. Start the dump. Enter:

   ```
   spxtape dump 181-182 img class a append
   spxtape dump 181-182 nls class a run
   ```

   Specifying the APPEND operand on the first command combines these two commands into a single logical command. CP begins processing the files as soon as you enter the first command in the sequence. CP displays progress responses, requests to mount new tapes (if necessary), and a final response when the dump function is complete. Each file dumped is recorded in the volume log for that tape volume.
6. Remove the tapes from the tape drives.

Restoring System Data Files from Tape

Use the SPXTAPE LOAD command to restore copies of z/VM system data files that were previously saved on tape by SPXTAPE DUMP.

To restore all the system data files from tape:

1. Find one or more free tape drives (it is recommended that you use at least two drives to allow overlapping tape mounting) and mount the tapes that hold the system data files to be loaded.

   The order in which you mount the tapes does not matter, as long as the files were dumped with one logical SPXTAPE command and you use one logical SPXTAPE command to load them. A logical command can be either one independent command or a sequence of appended commands (entered with the APPEND operand). If a file is split across two or more tape volumes, CP fits the parts together correctly.

   A free tape drive is one that is not offline, in use, or dedicated to another virtual machine. The tape drive must be fully supported by z/VM. For a list of supported devices, see the *z/VM: General Information*. To find out the status of all of your tape drives, enter:

   ```
   query tapes all
   ```

   If the status of any tape drive is FREE, mount a dump tape, which has been enabled for reading only, and go to the next step. To enable a tape reel as read-only, make sure the tape does not have a write-enable ring. To enable a tape cartridge as read-only, turn the thumbwheel on the cartridge so the white dot shows.

   If the status of any tape drive is OFFLINE, enter:

   ```
   vary online rdev
   ```

   where *rdev* is the tape drive's real device number. Then mount the tape, enabled for reading only, and go to the next step.
2. Attach each tape drive to your virtual machine. Enter:

   ```
   attach rdev to * as vdev
   ```

   where *vdev* is an available virtual device number.
3. Spool your console to wherever you want the SPXTAPE log files to go. For example, spool the console to yourself so the logs are sent to your reader. Enter:

   ```
   spool cons *
   ```

   Note: If you spool your console *after* entering the SPXTAPE command, the logs may be sent to different destinations.

4. If necessary, use the CMS TAPE command to position each tape to where CP can begin reading data. The tape must be positioned either at the beginning of the volume or at the beginning of a tape file (following the tape mark at the end of the previous tape file).
5. If you want to find out what system data files are on tape before you do the load, and the volume logs for the dump are not available, you can use the SPXTAPE SCAN command.

   a. To scan the tape, enter:

      ```
      spxtape scan vdev1-vdev2 sdf all run
      ```

      where *vdev1-vdev2* is the range of virtual device numbers for the tape drives you are using. If you are using only one drive, you can enter a single virtual device number. The RUN operand rewinds and unloads the tape when processing to that volume completes.

      Note: If you have only one tape to scan on each tape drive, you may prefer to use the REWIND operand, which rewinds the tape but does not unload it from the drive. If the tape contains multiple tape files dumped by multiple logical SPXTAPE DUMP commands, you must use multiple SPXTAPE SCAN commands to scan them. In that case, you may want to use the LEAVE operand, which leaves the tape positioned to process the next tape file.

      In response, CP scans each image library, named saved system, saved segment, national language support file, system trace file, and any existing UCR file in the selected tape file on each tape. As it processes the files, CP reports its progress at the interval requested with the PROGress_interval option (or about every 15 seconds by default) by displaying a message on the screen.

      CP records information about the scan in two or more logs. The command summary log contains information about the progress and status of the logical SPXTAPE command. The volume log contains information about the files processed by the logical SPXTAPE command that are associated with a particular tape volume. Therefore, several volume logs are created for one logical SPXTAPE command if more than one tape drive is used or if more than one tape volume is mounted on any drive.

   b. If you entered the SPXTAPE SCAN command with the RUN disposition, CP sends you a response when processing of a tape volume is complete that:

      i) Asks you to mount the next tape on that device

      ii) Indicates the number of files and spool pages processed so far.

      If there are other tapes to be processed, mount the next tape. If you are using more than one tape drive, CP continues scanning on the other drives. When you mount and ready the new tape, CP starts scanning on that drive again.

      If there are no other tapes to be mounted, and all tapes currently mounted have finished processing (you have received the "mount next tape" response for each drive), use the SPXTAPE END command to end the operation. Enter:

      ```
      spxtape end vdev1-vdev2
      ```

      where *vdev1-vdev2* is same virtual device range you specified on the SPXTAPE SCAN command. The tape is positioned according to the disposition specified on the SPXTAPE SCAN command.

   c. To see the results of the scan, look at the volume logs. Each volume log has a unique name (file name and file type) related to both the tape volume and the SPXTAPE command that caused the creation of the log. For example, the volume log for the second volume processed on virtual device 181 for an SPXTAPE SCAN command has the name:

      ```
      xxyyS181 hhmmss02
      ```

      where *xxyy* is the month and day the command was issued, S indicates the operation was SCAN, 181 is the virtual device number of the tape drive, *hhmmss* is the time stamp, and 02 is the volume sequence number.

      The volume log contains the following entry for each file scanned on the volume: Some of the entries have been truncated to fit the page.

```
USERID FILE QUEUE FLNAM FLTYP OPNDAT OPNTIM ORIGID SZ SEG_ST HLDSTAT RECCNT C FORM DEST
LSTPG
userid file queue flnam fltyp opndat opntim origid sz CMPLETEhldstat reccnt c form dest
lstpag
```

Note: The heading is included only once for each uninterrupted series of entries.

If the SEG_STAT field contains PARTIAL instead of COMPLETE, it means one of the following:

- The file is split between this volume and one or more other volumes, and this part is *not* the one that completes the file (the last part scanned). If all parts of the file have scanned correctly, the volume log entry for the last part scanned contains COMPLETE in the SEG_STAT field.

  Note: The last part scanned may contain the end of the file, the beginning, or some section in between, depending on the order in which the tape volumes were mounted.

- CP encountered an I/O error or some other problem trying to read the file from the tape. In that case, you also receive error messages describing the problem. The error messages are recorded in the command summary log.

6. If necessary, use the CMS TAPE command to position each tape to the beginning of the tape file you want to load.
7. Load all the system data files. Enter:

   ```
   spxtape load vdev1-vdev2 sdf all run
   ```

   where *vdev1-vdev2* is the range of virtual device numbers of the tape drives you are using. If you are using only one drive, you can enter a single virtual device number. The RUN operand rewinds and unloads the tape when processing of that volume completes.

   Note:

   a. If the tape contains tape files dumped by multiple logical SPXTAPE DUMP commands, you must use multiple logical SPXTAPE LOAD commands to load them. In that case, you may prefer to use the LEAVE operand, which leaves the tape positioned to process the next tape file.
   b. You can prevent the loading of any file that duplicates a file already on the system. Use the NODUP operand, as follows:

      ```
      spxtape load vdev1-vdev2 sdf all run nodup
      ```

      However, the NODUP operand requires the system to do *considerable* additional processing. Before using this operand, see the information in the description of the SPXTAPE LOAD command in the *z/VM: CP Commands and Utilities Reference*.

   In response, CP reads each image library, named saved system, saved segment, national language support file, system trace file, and any existing UCR file from the tape and adds it to the appropriate spooling system queue. All the available tape drives in the specified range are used concurrently; as one file is still being loaded from a tape, the next file is read from the next available tape. As it processes the files, CP reports its progress at the interval requested with the PROGress_interval option (or about every 15 seconds by default) by displaying a message on the screen. As it loads each file, CP assigns a new spool file identification number to it. CP also sends a response to the owner of the file (if the owner is logged on and has IMSG set ON).

   CP records information about the load in two or more logs. The command summary log contains information about the progress and status of the logical SPXTAPE command. The volume log contains information about the files processed by the logical SPXTAPE command that are associated with a particular tape volume. Therefore, several volume logs are created for one logical SPXTAPE command if more than one tape drive is used or if more than one tape volume is mounted on any drive.
8. If you entered the SPXTAPE LOAD command with the RUN disposition, CP sends you a response when processing of a tape volume is complete that:

   a. Asks you to mount the next tape on that device
   b. Indicates the number of files and spool pages processed so far.

If there are other tapes to be processed, mount the next tape. If you are using more than one tape drive, CP continues loading from the other drives. When you mount and ready the new tape, CP starts loading from that drive again.

If there are no other tapes to be mounted, and all tapes currently mounted have finished processing (you have received the "mount next tape" response for each drive), use the SPXTAPE END command to end the operation. Enter:

```
spxtape end vdev1-vdev2
```

where *vdev1-vdev2* is the same virtual device range you specified on the SPXTAPE LOAD command. The tape is positioned according to the disposition specified on the SPXTAPE LOAD command.

9. Remove the tapes from the tape drives.
10. If you want to find out what files have been loaded, you can look at the volume logs. Each volume log has a unique name (file name and file type) related to both the tape volume and the SPXTAPE command that caused the creation of the log. For example, the volume log for the second volume processed on virtual device 182 for an SPXTAPE LOAD command has the name:

    ```
    xxyyL182 hhmmss02
    ```

    where *xxyy* is the month and day the command was issued, L indicates the operation was LOAD, 182 is the virtual device number of the tape drive, *hhmmss* is the time stamp, and 02 is the volume sequence number.

    The volume log contains the following entry for each file loaded from the volume: Some of the entries have been truncated to fit the page.

    ```
    USERID FILE QUEUE FLNAM FLTYP OPNDAT OPNTIM ORIGID SZ SEG_ST HLDSTAT RECCNT C FORM DEST
    LSTPG
    userid file queue flnam fltyp opndat opntim origid sz CMPLETEhldstat reccnt c form dest
    lstpag
    ```

    Note: The heading is included only once for each uninterrupted series of entries.

    If the SEG_STAT field contains PARTIAL instead of COMPLETE, it means of one the following:

    - The file is split between this volume and one or more other volumes. and this part is *not* the one that completes the file (the last one loaded). If all parts of the file have loaded correctly, the volume log entry for the last part loaded contains COMPLETE in the SEG_STAT field.

      Note: The last part loaded may contain the end of the file, the beginning, or some section in between, depending on the order in which the tape volumes were mounted.
    - CP encountered an I/O error or some other problem trying to read the file from the tape or write it to DASD. In that case, you also receive error messages describing the problem. The error messages are recorded in the command summary log.

    Note: If you used the NODUP operand on the SPXTAPE LOAD command, duplicate files are not loaded but are indicated by entries in the volume log with DUP_FILE in the SEG_STAT field.
11. To see what system data files are on the system, you can check the various system data file queues. If you did not use the NODUP operand on the SPXTAPE LOAD command, this will show if you you loaded any duplicate files. If you find duplicate files, contact the person who maintains your installation's system data files to find out which files to purge.

    To check the named saved systems and saved segments, enter:

    ```
    query nss
    ```

    To see whether any named saved system is defined using the VMGROUP option, enter:

    ```
    query nss map
    ```

    To check the image libraries, enter:

```
query img
```

To check the national language support files, enter:

```
query nls
```

To check the system trace files, enter:

```
query trf
```

In response to each of these commands, CP lists all the files in the specified system data file queue.

Note: If an external security manager (ESM) is installed on your system, the RECS, FILENAME, and FILETYPE fields in the response may contain asterisks. Asterisks are used to mask fields the user is not authorized to view while logged on with the current security label. For additional information, contact your security administrator.

To get only a count of a particular type of system data file, use the COUNT operand. For example, to get a count of all the class S (skeleton) named saved systems and saved segments, enter:

```
query nss class s count
```

To get a count of all the system data files, enter:

```
query sdf
```

In response to this command, CP displays a count of each type of system data file.

The preceding steps tell you how to restore all the system data files dumped to tape. You can also select the system data files to be loaded by queue, user ID (system trace files only), spool file identification number or range, class, file name pattern, file type pattern, or combinations of these attributes. A file name or file type pattern can be either a complete file name or file type or a string containing wild cards (* and %). The following example shows you how to restore particular classes of image libraries, named saved systems, and saved segments.

For more information about the SPXTAPE logs and the options you can specify on the SPXTAPE LOAD command, see the *z/VM: CP Commands and Utilities Reference*.

Example—Loading Particular Classes of Image Libraries, Named Saved Systems, and Saved Segments from Tape

You previously backup up all the system data files. Now you want to restore only the class A image libraries and the class A and S named saved systems and saved segments.

To load only the class A image libraries and class A and S named saved systems and saved segments from tape:

1. Find one or more free tape drives. Enter:

   ```
   query tapes all
   ```

   CP’s response tells you the status of tape drives 5C0 and 61B is FREE. Both are 3490s, so you mount a dump tape on each.

2. Attach the tape drives to your virtual machine. Virtual device numbers 181 and 182 are available, so you enter:

   ```
   attach 5c0 to * as 181
   attach 61b to * as 182
   ```

3. Spool your console. To spool the console to yourself, enter:

   ```
   spool cons *
   ```

4. If necessary, use the CMS TAPE command to position each tape to the beginning of the tape file that contains the files you want to load.
5. Start the load. Enter:

   ```
   spxtape load 181-182 img class a append
   spxtape load 181-182 nss class as run
   ```

   Specifying the APPEND operand on the first command combines these two commands into a single logical command. CP begins processing the files after you enter the last command in the sequence. CP displays progress responses and requests to mount new tapes. Each file loaded is recorded in the volume log for that tape volume.

   Note: The previous procedure assumes the files were saved on tape using one logical SPXTAPE DUMP command. If independent SPXTAPE DUMP commands were used to save the files (for example, if they were saved at different times), you must use independent (not appended) SPXTAPE LOAD commands to restore them. Wait for the first command to finish processing before you enter the second command. The disposition operand (RUN, REWIND, or LEAVE) you use on each command depends on whether the files are contained on one tape volume or multiple tape volumes.
6. When there are no more tapes to be processed (all tapes currently mounted have finished processing and there are no more tapes to be mounted), end the SPXTAPE LOAD operation. Enter:

   ```
   spxtape end 181-182
   ```
7. Remove the tapes from the tape drives.

Moving User Data between Systems

You may need to transfer user data between one or more z/VM systems. This is a particularly important migration task when your installation is in transition from one production operating system to another.

Moving z/VM Spool Files to another z/VM system

To move spool files from one z/VM system to another:

1. On the currently used z/VM system, use the SPXTAPE DUMP command to write the spool files to tape.
2. On the target z/VM, use the SPXTAPE LOAD command to restore the spool files from the tape.

To load spool files from a tape:

1. Find a free tape drive and mount the tape that holds the dumped spool files.

   A free tape drive is one that is not offline, in use, or dedicated to another virtual machine. The tape drive must be fully supported by z/VM. For a list of supported devices, see the *z/VM: General Information*. To find out the status of all of your tape drives, enter:

   ```
   query tapes all
   ```

   If the status of any tape drive is FREE, mount the tape, which has been enabled for reading only, and go to the next step. To enable a tape reel as read-only, make sure the tape does not have a write-enable ring. To enable a tape cartridge as read-only, turn the thumbwheel on the cartridge so the white dot shows.

   If the status of any tape drive is OFFLINE, enter:

   ```
   vary online rdev
   ```

   where *rdev* is the tape drive's real device number. Then mount the tape, enabled for reading only, and go to the next step.
2. Restore to the system all the spool files from the reader queue that are on the tape. Enter:

   ```
   spxtape load vdev reader all rewind
   ```

where *vdev* is the tape drive's virtual device number.

In response, CP loads from the tape each reader file and places it on the system's reader queue. As it loads each file, CP assigns a new spool file identification number and sends both you and the file owner a message. If you are the owner of a file that CP loads, you receive two messages. When CP has loaded all the files, it tells you the load function is complete and rewinds the tape.

3. Restore to the system all the spool files from the print queue that are on the tape. Enter:

   ```
   spxtape load vdev printer all rewind
   ```

   where *vdev* is the tape drive's virtual device number.

   In response, CP loads from the tape each print file and places it on the system's print queue. As it loads each file, CP assigns a new spool file identification number and sends both you and the file owner a message. If you are the owner of a file that CP loads, you receive two messages. When CP has loaded all of the files, it tells you the load function is complete and rewinds the tape.
4. Restore to the system all the spool files from the punch queue that are on the tape. Enter:

   ```
   spxtape load vdev punch all rewind
   ```

   where *vdev* is the tape drive's virtual device number.

   In response, CP loads from the tape each punch file and places it on the system's punch queue. As it loads each file, CP assigns a new spool file identification number and sends both you and the file owner a message. If you are the owner of a file that CP loads, you receive two messages. When CP has loaded all of the files, it tells you the load function is complete and rewinds the tape.
5. Check the spool file queues to make sure CP restored all of the files.

   To check the files in the system's reader queue, enter:

   ```
   query reader
   ```

   To check the files in the system's print queue, enter:

   ```
   query printer
   ```

   To check the files in the system's punch queue, enter:

   ```
   query punch
   ```

   In response, CP lists all of the files in the reader, print, and punch queues, respectively.
6. Remove the tape from the tape drive.

For the other options you can specify on the SPXTAPE LOAD command, see the *z/VM: CP Commands and Utilities Reference*.

Archiving User Data

Another part of operating a z/VM system is archiving user data. Because the resources of the installation are limited, you need to archive the old data to make room for current data. The responsibility for setting archiving policies and performing archiving operations rests on operations. As an operator you may have to use z/VM archiving facilities and use production system archiving facilities.

z/VM allows you to archive data by storing it on tape. For more information on how you can use z/VM facilities to move data to tape, see "Backing Up System and User Data" on page 213.

Some production systems have products that help you perform archiving operations.

Initializing DASD Volumes

Finally, you sometimes have to initialize the storage media on which data resides, specifically, DASD volumes. This is particularly important when the volume formerly contained sensitive data. The responsibility for safeguarding data from unauthorized persons rests partly on operations.

If you need to initialize a DASD volume so that it can be used as a z/VM system volume, use the Device Support Facilities program (ICKDSF). It is the preferred method of CP volume maintenance and is required for certain DASD type. For more information on running this program operation, see “Formatting and Allocating a System Volume” on page 114.

If you need to initialize a DASD volume so that it can be used in another operating system's environment, you should also use the Device Support Facilities program (ICKDSF). For more information on running this program, see “Running the Device Support Facilities Program” on page 245.

Chapter 11. Running Utility Programs

This chapter provides guidance and reference information for utilities that can be run either stand-alone or under z/VM. After reading this chapter, you can:

- Locate information on running the Input/Output Configuration Program (IOCP)
- Run the Stand-Alone Program Loader Creation Utility (SALIPL)
  - Stand-alone on a real processor
  - Stand-alone in a virtual machine
  - Under CMS in a virtual machine
- Locate information on running the stand-alone dump utility
- Run the DASD dump restore utility (DDRXA)
  - Stand-alone loaded from SAPL
  - Stand-alone on a real processor
  - Stand-alone in a virtual machine
  - Under CMS in a virtual machine
- Locate information on running the Device Support Facilities program
- Find information about utilities that run in a production system.

Running IOCP

An Input/Output Configuration Program (IOCP) defines the real I/O device configuration to your processor complex.

If you have a supported IBM processor complex, you can run IOCP under z/VM. For information about running IOCP under z/VM on supported processors, see the appropriate *IOCP User's Guide*.

Running the Stand-Alone Program Loader Creation Utility (SALIPL)

Use the Stand-Alone Program Loader Creation Utility (SALIPL) to install the Stand-Alone Program Loader (SAPL). SALIPL is available as a module that runs under CMS and as a stand-alone utility.

Note:

1. You cannot use the SALIPL utility to install SAPL to a SCSI DASD. You must use the SALIPL utility while running under CMS.
2. You should not use the SALIPL utility to install SAPL if you specify the DEVICE option on the SALIPL utility. Using the creation utility will cause you to lose the information from any DEVICE option that you specified using SALIPL. For this function, you must use the SALIPL utility under CMS.

Before You Begin

Before you run SALIPL, you must know:

- Where SALIPL is located
- How you want to run SALIPL
- How you formatted SAPL

Before you run SALIPL, you must know its location. The SALIPL executable module is on the system S-disk (190) when z/VM is shipped from IBM. During system generation, your installation puts a copy of the program either on tape or in a file on DASD. To find out where the program is, contact the person at your installation who generates your z/VM system.

Next, decide how you want to run the program. To run the program on a real processor, see "Running SALIPL on a Real Processor" on page 234. To run SALIPL in a virtual machine, see "Running SALIPL in a Virtual Machine" on page 235.

Finally, before you write SAPL on a DASD volume, it must have been formatted either by CMS or by the Device Support Facilities Program. For more information about the Device Support Facilities Program, see "Running the Device Support Facilities Program" on page 245.

Running SALIPL on a Real Processor

To run SALIPL on a real processor, you first have to ensure that the program is loadable.

Making SALIPL Loadable on a Real Processor

If SALIPL resides as a module file on DASD and SAPL is already installed on a DASD volume, it is already loadable. If SALIPL resides on a tape, it is already loadable. If SALIPL resides as an IPL file on a DASD volume, do one of the following:

- Put a copy of the program on tape.
- Create a card deck of the program.

To put a copy of the program on tape, use the CMS FILEDEF and MOVEFILE commands. For example, suppose you want to dump to tape a loadable version of the SALIPL program. The program resides on the K-disk with a file name of IPL SALIPL. To put this file on tape, do the following:

1. Ask the operator to:
   - Mount on a tape drive a scratch tape enabled for writing.
   - Attach the tape drive to your virtual machine as virtual device number 181.
2. Define for the MOVEFILE command the file you want to move to tape. Enter:

   ```
   filedef in disk ipl salipl k
   ```

   If you IPL a named system, (CMS) 16 MB of storage may suffice. However, if you IPL 190, then you will need 20 MB of storage.
3. Define for the MOVEFILE the tape drive to which you want to move the file. Enter:

   ```
   filedef out tap1 (recfm f lrecl 80 block 80
   ```
4. Wait for CP to send you a message that the tape drive has been attached.
5. Move the SALIPL file from disk to tape. Enter:

   ```
   movefile in out
   ```

To create a card deck of the program, first enter the CMS PUNCH command with the NOHEADER option. Then punch the resulting spool file on a real card punch.

Loading SALIPL on a Real Processor

You are now ready to load the program on the real processor.

1. If necessary, shut down the program on the real processor.
2. If SALIPL resides on a tape, mount the tape on a compatible tape drive. If it resides on a card deck, load the deck into a real reader.
3. If you are using SAPL to load SALIPL, you must specify the console to be used by SALIPL by entering its device number in the first four characters of the LOADPARM on your processor console. If you are loading from cards or from tape, this is optional.
4. Using your processor complex's system console, IPL with the device number of the tape, card reader, or DASD volume containing SAPL.

5. If you are using SAPL, you must now specify the location of the minidisk containing the SALIPL module and the name of the module on the SAPL screen and hit the load key (PF10).

You are now ready to start running the program. See "Filling in the SALIPL Menu Screen" on page 237.

Running SALIPL in a Virtual Machine

You can run SALIPL as a CMS command or as a stand-alone utility. For information on the SALIPL command, see *z/VM: CP Commands and Utilities Reference*. To run SALIPL as a stand-alone program, it must be loaded by SAPL. To run SALIPL in a virtual machine, you first have to ensure that the program is loadable.

Making SALIPL Loadable in a Virtual Machine

If SALIPL resides as a MODULE file on DASD and SAPL is already installed on a DASD volume, it is already loadable. Running this utility as stand-alone is not recommended and for the most part is exceptional. The recommended and easiest method is to run SALIPL as a CMS command. For detailed information on running SALIPL under CMS you should refer to the SALIPL utility in the *z/VM: CP Commands and Utilities Reference*.

If SALIPL resides on a tape, it is already loadable. If SALIPL resides as an IPL file on a DASD volume, do one of the following:

- Dump a copy of the program to tape.
- Place a copy of the program in your virtual reader.

To put a copy of the program on tape, use the CMS FILEDEF and MOVEFILE commands. For example, suppose you want to dump to tape a loadable version of the SALIPL program. To put this file on tape, do the following:

1. Ask the operator to:
   - Mount on a tape drive a scratch tape enabled for writing.
   - Attach the tape drive to your virtual machine as virtual device number 181.
2. Define for the MOVEFILE command the file you want to move to tape. Enter:

   ```
   filedef in disk ipl salipl k
   ```

3. Define for the MOVEFILE the tape drive to which you want to move the file. Enter:

   ```
   filedef out tap1 (recfm f lrecl 80 block 80
   ```

4. Wait for CP to send you a message that the tape drive has been attached.
5. Move the SALIPL file from disk to tape. Enter:

   ```
   movefile in out
   ```

To place a copy of the SALIPL program in your virtual reader:

1. Bring up CMS in your virtual machine. Enter:

   ```
   ipl 190
   ```

   Or, if your installation has installed CMS in a named saved system, enter:

   ```
   ipl cmsname
   ```

   where *cmsname* is the name of your CMS system.
2. If SALIPL does not reside on the CMS system S-disk (190), link to the minidisk that contains the program.
3. Route punch spool files to your virtual reader. Enter:

```
spool punch *
```

4. Punch a copy of the SALIPL program. Enter:

   ```
   punch ipl salipl * (noheader
   ```

You now have a copy of the SALIPL program in your virtual reader.

Loading SALIPL in a Virtual Machine

You are now ready to load the program in your virtual machine.

Loading SALIPL from Tape

If SALIPL resides on a tape:

1. Ask the operator to mount the tape on a compatible tape drive and attach the tape drive to your virtual machine as virtual device number 181.
2. Wait for CP to send you a message that the tape drive has been attached.
3. Make sure your virtual machine is in ESA mode. Enter:

   ```
   set machine esa
   ```

4. Make sure your virtual console is in 3270 mode. Enter:

   ```
   terminal conmode 3270
   ```

5. Load the SALIPL program into your virtual machine storage. Enter:

   ```
   ipl 181
   ```

   or

   ```
   ipl 181 loadparm vcon
   ```

   where *vcon* is the virtual device number of your console.

You are now ready to start running the program. See "Filling in the SALIPL Menu Screen" on page 237.

Loading SALIPL from Your Virtual Reader

If SALIPL resides in your virtual reader:

1. Make the file ready to load and prevent CP from purging it from your reader. Enter:

   ```
   change reader spoolid nohold keep
   ```

   where *spoolid* is the spool file identification number that CP assigned the spool file when it placed it in your reader.
2. Put the SALIPL file first in your reader queue. Enter:

   ```
   order reader spoolid
   ```

   where *spoolid* is the spool file identification number of the SALIPL reader file.
3. Make sure the reader can process the SALIPL file. Enter:

   ```
   spool vdev class *
   ```

   where *vdev* is the virtual device number of your virtual reader.
4. Make sure your virtual machine is in ESA mode. Enter:

   ```
   set machine esa
   ```

5. Make sure your virtual console is in 3270 mode. Enter:

   ```
   terminal conmode 3270
   ```

6. Load the SALIPL program into your virtual machine storage. Enter:

   ```
   ipl vdev
   ```

   or

   ```
   ipl vdev loadparm vcon
   ```

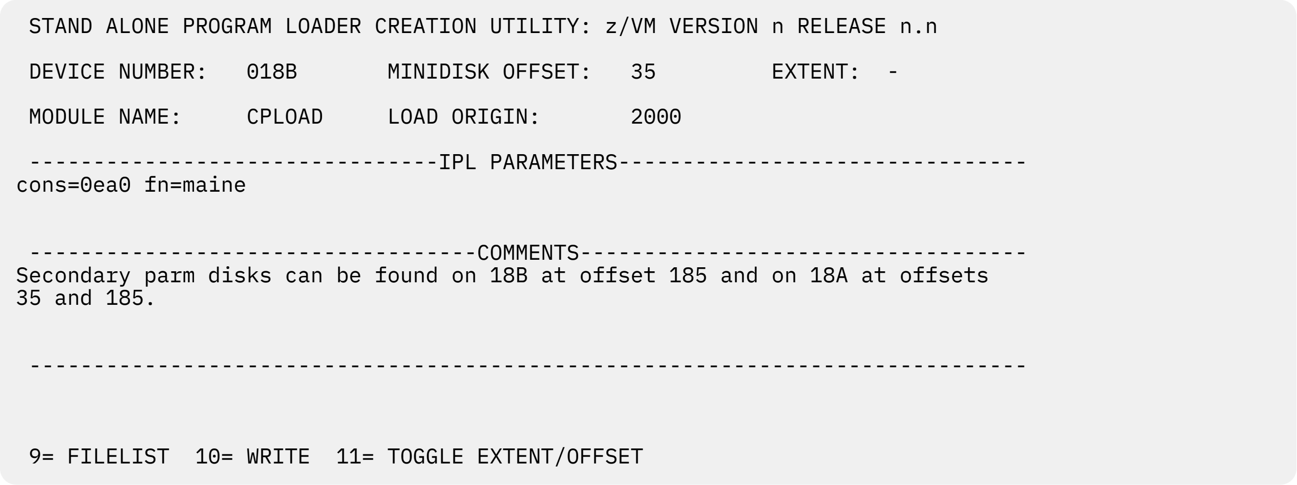
   where *vdev* is the virtual device number of your virtual reader and *vcon* is the virtual device number of your console.

You are now ready to fill in the SALIPL menu screen.

Filling in the SALIPL Menu Screen

If your console was at device number 0009 or 001F or you specified your console address using the LOADPARM, you should get the SALIPL menu screen as soon as SALIPL is loaded into storage. If you do not get this screen, press ENTER without entering any data. Entering a null line tells SALIPL your console's device number; then, it can display the menu screen.

If you have non-console devices with device numbers of 0009 or 001F, or it is possible that a device other than the console you wish to use may generate attention interrupts, it is recommended that you use the LOADPARM to specify the device number of the console you wish to use.



```
 STAND ALONE PROGRAM LOADER CREATION UTILITY: z/VM VERSION n RELEASE n.n

 DEVICE NUMBER:   018B       MINIDISK OFFSET:   35         EXTENT:  -

 MODULE NAME:     CPLOAD     LOAD ORIGIN:       2000

 --------------------------------IPL PARAMETERS--------------------------------
cons=0ea0 fn=maine


 -----------------------------------COMMENTS-----------------------------------
Secondary parm disks can be found on 18B at offset 185 and on 18A at offsets
35 and 185.


 ------------------------------------------------------------------------------



 9= FILELIST  10= WRITE  11= TOGGLE EXTENT/OFFSET
```

Figure 12. Stand-Alone Program Loader Creation Utility

The menu screen appears with defaults already filled in for some of the fields. You can, however, enter information in all fields except the EXTENT field. These fields define where the stand-alone program loader is to be written and what it's default parameters are to be.

The Stand-Alone Program Loader Creation Utility screen contains the following fields:

DEVICE NUMBER
Specifies the device number of the DASD volume that SAPL is to be written to.

Note: You cannot use the SALIPL utility to install SAPL to a SCSI DASD. You must use the SALIPL utility while running under CMS.

MINIDISK OFFSET
Specifies the offset from the beginning of the DASD volume to the start of the CMS minidisk that SAPL is to use by default. Offset is the number of cylinders for count key devices and blocks for fixed block architecture devices.

EXTENT

Specifies the extent number from 1 to 9 (first to ninth). This field may be used only with volumes that have been formatted using the Device Support Facilities Program and has an allocation map containing PARM extents written on them. When a valid extent is entered, the offset corresponding to the extent will be displayed in the MINIDISK OFFSET field.

Note: Only one of the fields, EXTENT or MINIDISK OFFSET, can be altered at a time. If the specified area(s) have not already been CMS- formatted, message 'FATAL DASD I/O ERROR CSW=0E00' may be encountered when PF10 (WRITE) is entered at the SALIPL creation menu screen.

Use PF11 to show you which field is currently alterable.

MODULE NAME

Specifies the name of a module on the minidisk that will be loaded by default by SAPL.

LOAD ORIGIN

Specifies the storage location (HEX address) into which SAPL will load relocatable modules.

Note: If you specify a LOAD ORIGIN value for a CPLOAD MODULE, you should be aware of the following restrictions:

1. The entire CP module, including its required CP data structures, must fit below the 2 GB line.
2. If you specify an odd page address, CP will relocate to the next higher even page address (8 KB boundary).
3. CP is not relocatable. After initially loading at the specified address, CP will relocate itself to location X'2000'.

IPL PARAMETERS

Specifies the default data that will be passed to the loaded program by SAPL. The contents of this field depends on the program loaded by SAPL. For parameters valid for CP, see "Passing IPL Parameters" on page 13.

Once all of the fields have been filled in, press PF10 (WRITE) to write the IPL records (SAPL) to the dasd pack.

Running the Stand-Alone Dump Utility

z/VM: CP Planning and Administration contains information about the stand-alone dump utility. It documents how the system administrator can install and configure the stand-alone dump program. It also contains information about how the system operator can take a stand-alone dump and load the dump for analysis.

Running the DASD Dump Restore Utility (DDRXA)

The DASD dump restore utility (DDRXA) runs:

- Stand-alone loaded from SAPL
- Stand-alone on a real processor
- Stand-alone in a virtual machine
- Under CMS in a virtual machine

You can use DDRXA to:

- Dump data from a DASD to tape
- Restore dumped data from a tape back to a DASD
- Copy data from DASD to DASD or from tape to tape
- Print or display records from DASD or tape

You tell the program what to do through DDRXA control statements. You can either enter the control statements from your console or supply them from the device on which you load the program (the IPL device).

Before You Begin

Before you run DDRXA, you must know its location. During system generation, your installation puts a copy of the program either on tape or in a file on DASD. To find out where the program is, contact the person at your installation who generates your z/VM system.

Also, decide how you want to submit the DDRXA control statements. You can either enter the statements from your console or supply them from the device on which you load the program (the IPL device). For more information, see "Supplying DDRXA Control Statements" on page 243.

Next, decide how you want to run the program. See the appropriate section:

- "Running DDRXA Stand-Alone Loaded from SAPL" on page 239
- "Running DDRXA Stand-Alone on a Real Processor" on page 239
- "Running DDRXA Stand-Alone in a Virtual Machine" on page 240
- "Running DDRXA under CMS in a Virtual Machine" on page 242

Note: The easiest way to run DDRXA is under CMS.

Finally, before you restore or copy important data to a DASD volume, use the Device Support Facilities Program to inspect the volume for defective tracks. For more information, see "Running the Device Support Facilities Program" on page 245.

Running DDRXA Stand-Alone Loaded from SAPL

For more information about the SAPL utility, see "Running the Stand-Alone Program Loader Creation Utility (SALIPL)" on page 233.

To run DDRXA loaded from SAPL, you first have to ensure that the program is available on the SAPL Parm disk.

Making DDRXA Loadable from SAPL

To run DDRXA loaded from SAPL, a copy of the DDR MODULE file must be on your Parm disk. It may already be on the Parm disk as a result of product installation. If not, locate the file (it should be on the system disk accessed at file mode S) and use the COPY command to create a copy on your Parm disk.

Loading DDRXA with SAPL

There are two ways to load DDRXA with SAPL:

- IPL the DASD specifying the LOADPARM parameter with the console address, enter `DDR` in the **MODULE NAME** field, and press PF10. DDR will come up on the console.
- Specify the `MODULE DDR` option when you run the SALIPL utility to install SAPL on the DASD. IPLing the DASD with no LOADPARM parameter will automatically start DDR. If your console address is not 9 or 1F, you must press Enter for DDR to recognize your console address. For information about running SALIPL under CMS, see the description of the SALIPL utility in the *z/VM: CP Commands and Utilities Reference*.

Running DDRXA Stand-Alone on a Real Processor

To run DDRXA stand-alone on a real processor, you first have to ensure that the program is loadable.

Making DDRXA Loadable on a Real Processor

If DDRXA resides on a tape, it is already loadable.

If the DDRXA program resides in a file on a DASD volume, do one of the following:

- Put a copy of the program on tape
- Create a card deck of the program.

To put a copy of the program on tape, use the CMS FILEDEF and MOVEFILE commands. For example, suppose you want to dump to tape a loadable version of the DDRXA program. You must run a file list (filel), or IPL DDRXA, to find the disk where the DDRXA module resides. To put this file on tape, do the following:

1. Ask the operator to:
   - Mount on a tape drive a scratch tape enabled for writing
   - Attach the tape drive to your virtual machine as virtual device number 181.
2. Define for the MOVEFILE command the file you want to move to tape. Enter:

   ```
   filedef in disk ipl ddrxa *   ( use the proper mode for * )
   ```

   If you IPL a named system, (CMS) 16 MB of storage may suffice. However, if you IPL 190, then you will need 20 MB of storage.
3. Define for the MOVEFILE command the tape drive to which you want to move the file. Enter:

   ```
   filedef out tap1 (recfm f lrecl 80 block 80
   ```

4. Wait for CP to send you a message that the tape drive has been attached.
5. Move the DDRXA file from disk to tape. Enter:

   ```
   movefile in out
   ```

To create a card deck of the program, first enter the CMS PUNCH command with the NOHEADER option. Then, punch the resulting spool file on a real card punch.

Loading DDRXA on a Real Processor

You are now ready to load the program on the real processor.

1. If necessary, shut down the z/VM system.
2. If DDRXA resides on a tape, mount the tape on a compatible tape drive. If it resides in a card deck, load the deck into a real card reader.
3. Using your processor complex's system console, IPL the DDRXA program.

You are now ready to start running the program. See "Supplying DDRXA Control Statements" on page 243.

DDRXA and the IPL LOADPARM

Stand-alone DDRXA uses the loadparm passed from the system console IPL panel. Stand-alone DDRXA will use the device number specified in the loadparm field as its console when the first four bytes of the field contain a valid device address (the field is generally eight bytes in length). When the loadparm field is not used to pass a console address, DDRXA will attempt to use device 9, then 01F. If neither is found DDRXA will use the first device that presents an attention interrupt.

Terminals on a D/T3174, D/T3274 or other terminal control unit will display a PRG505 and lightning bolt when DDRXA is IPLed on the hardware. An attention interrupt cannot be generated off these devices to start DDRXA. Failure to use the loadparm field may result in PROG505 appearing on the console that was meant to be used for DDRX. The user may not be able to use DDRXA.

From the Hardware Management Console you can use the integrated 3270 console with stand-alone DDR. The integrated 3270 console is identified to DDR as SYSG in the IPL loadparm. This console provides the ability to operate DDR without any attached 3270 devices.

Running DDRXA Stand-Alone in a Virtual Machine

To run DDRXA stand-alone in a virtual machine, you first have to ensure that the program is loadable.

Making DDRXA Loadable in a Virtual Machine

If DDRXA resides on tape, it is already loadable.

If the DDRXA program resides in a file on a DASD volume, do one of the following:

- Dump a copy of the program to tape
- Place a copy of the program in your virtual reader.

To dump the DDRXA program to tape, use the CMS FILEDEF and MOVEFILE commands.

For example, suppose you want to dump to tape a loadable version of the DDRXA program. The program resides on the K disk with a file name and file type of IPL DDRXA. To put this file on tape, do the following:

1. Ask the operator to:
   - Mount on a tape drive a scratch tape enabled for writing
   - Attach the tape drive to your virtual machine as virtual device number 181.
2. Define for the MOVEFILE command the file you want to move to tape. Enter:

   ```
   filedef in disk ipl ddrxaxa s
   ```

3. Define for the MOVEFILE command the tape drive to which you want to move the file. Enter:

   ```
   filedef out tap1 (recfm f lrecl 80 block 80
   ```

4. Wait for CP to send you a message that the tape drive has been attached.
5. Move the DDRXA file from disk to tape. Enter:

   ```
   movefile in out
   ```

To place a copy of the DDRXA program in your virtual reader:

1. If DDRXA does not reside on the CMS system volume, link to the minidisk that contains the program.
2. Bring up CMS in your virtual machine. Enter:

   ```
   ipl 190
   ```

   Or, if your installation has installed CMS in a named saved system, enter:

   ```
   ipl cmsname
   ```

   where *cmsname* is the name of your CMS system.
3. Route punch spool files to your virtual reader. Enter:

   ```
   spool punch *
   ```

4. Punch a copy of the DDRXA program and name the resulting punch file IPL DDRXA. Enter:

   ```
   punch ipl DDRXA * (noheader
   ```

You now have a copy of the DDRXA program in your virtual reader.

Loading DDRXA in a Virtual Machine

You are now ready to load the program in your virtual machine.

Loading DDRXA from Tape

If DDRXA resides on a tape:

1. Mount the DDRXA tape on a 3422, 3480, or 3490 tape drive and attach the tape drive to your virtual machine as virtual device number 181.

To attach the tape drive to your virtual machine, enter:

```
attach rdev * 181
```

where *rdev* is the real device number of the tape drive.

2. Load the DDRXA program into your virtual machine storage. Enter:

```
ipl 181
```

You are now ready to start running the program. See "Supplying DDRXA Control Statements" on page 243.

Loading DDRXA from Your Virtual Reader

If DDRXA resides in your virtual reader:

1. Make the file ready to load, and prevent CP from purging it from your reader. Enter:

   ```
   change reader spoolid nohold keep
   ```

   where *spoolid* is the spool file identification number that was assigned to the spool file by CP when it placed it in your reader.

   Note: Class D users can assign a security label to the DDRXA file if the following conditions are met:

   - An external security manager (ESM) is installed
   - Security label checking is enabled, and:
     - The user is exempt from ESM authorization checking, or
     - The system has been placed in a tranquil state by the security administrator.

   To assign a security label to the DDRXA file, include the SECLABEL *ssssssss* option on the CHANGE command, where *ssssssss* is a 1 - 8 character alphanumeric value.

   For additional information, contact your security administrator.

2. Put the DDRXA file first in your reader queue. Enter:

   ```
   order reader spoolid
   ```

   where *spoolid* is the spool file identification number of the DDRXA reader file.

3. Make sure the reader can process the DDRXA file. Enter:

   ```
   spool vdev class *
   ```

   where *vdev* is the virtual device number of your virtual reader.

4. Load the DDRXA program into storage. Enter:

   ```
   ipl vdev
   ```

   where *vdev* is the virtual device number of your virtual reader.

You are now ready to start running the program.

Running DDRXA under CMS in a Virtual Machine

To run DDRXA under CMS in a virtual machine, you first have to ensure that the program is available.

Making DDRXA Available in CMS

The DDR MODULE file should be located on your CMS system disk accessed at file mode S. (It should have been placed there during product installation.)

Loading DDRXA under CMS in a Virtual Machine

To load DDRXA under CMS in a virtual machine, issue the DDR command. For detailed information about running DDRXA under CMS, see the description of the DDR utility in the *z/VM: CP Commands and Utilities Reference*.

Supplying DDRXA Control Statements

You can supply control statements to DDRXA in one of two ways:

- You can enter DDRXA control statements from your console after you invoke the program.
- You can supply them from the device on which you load the program (the IPL device). This option requires you to set up the control statements before you invoke the program.

The syntax used with each DDRXA control statement is described in *z/VM: CP Commands and Utilities Reference*.

Entering DDRXA Control Statements from Your Console

Normally, you enter DDRXA control statements interactively from your console.

As soon as you load DDRXA into storage, it begins executing and sends you the following initial prompt:

```
z/VM DASD DUMP/RESTORE PROGRAM
ENTER CARD READER ADDRESS OR CONTROL STATEMENTS
ENTER:
```

If you do not receive this prompt, press the ENTER key without any data. (Enter a null line.) Entering a null line tells DDRXA your console's device number, so that it can send you the prompt. (DDRXA initially assumes that your console has device number 0009 or 001F.) You may then start entering control statements interactively.

Note: When you run DDRXA on a real processor, and you have a nonconsole device at either real device number 0009 or 001F, make sure the device is not operational. Otherwise, results are unpredictable.

In response to the initial prompt, enter the first statement for the first function. Entering statements for different functions is discussed in *z/VM: CP Commands and Utilities Reference*. However, when you enter the statements, remember that:

- DDRXA reads only the first 71 characters of data per statement.
- DDRXA ignores any data you enter after the last possible operand of a statement.

After you enter the first one, DDRXA prompts you for additional statements. When you have entered all the statements for the first function, enter a null line if needed. Null lines delimit groups of statements for DUMP, COPY, and RESTORE functions. You do not need to enter a null line for PRINT or TYPE functions.

DDRXA performs the first function and prompts you for another statement. Enter the statements for the next function in the same manner.

To end the program after DDRXA has completed the last function, enter a null line. The program ends.

Supplying DDRXA Control Statements from the IPL Device

Normally, you enter DDRXA control statements interactively after you IPL the program. However, if DDRXA resides on a DASD volume, you can direct DDRXA to read the control statements from the end of the file you IPL. To do this, you must set up the program and control statements before you IPL, as follows:

1. Edit the DDRXA program and put the control statements at the end of the file.
2. Transfer the program to tape, a card deck, or your virtual reader as usual.

The control statements transfer along with the program. Then you can load the program on the real processor (using the tape or the card deck) or in your virtual machine (using the tape or your virtual reader) as discussed previously.

After you load the program and receive the initial prompt, press ENTER. DDRXA reads all the control statements, completes the requested functions, and ends.

Rules for Supplying DDRXA Control Statements from the IPL Device

When you prepare DDRXA control statements for entry from a tape or from a card reader, keep in mind the following rules:

- DDRXA reads only columns 1 to 71.
- DDRXA ignores any data you enter after the last possible operand of a statement.
- Each group of statements you include must define a single function: DUMP, RESTORE, COPY, PRINT, or TYPE.
- If you want DDRXA to perform the same function on different DASD extents, you must include a statement for each extent. You may not include more than 300 extent statements for the same function.

  For example, suppose you want to dump to tape cylinders 1, 2, 3, 5, 7, 8, 10, 14, and 20 of a DASD. To do this, you must include six DUMP function statements, one for each noncontiguous set of cylinders (each extent).
- If you enter an incorrect statement, or if you enter the function statements in the wrong order, DDRXA sends you an error message. However, DDRXA continues to scan any remaining function statements for syntax errors.
- By default, DDRXA uses device number 00E as a printer. If you do not have a printer with device number 00E, use a SYSPRINT statement to define your printer. If the device specified on the SYSPRINT statement does not exist, DDRXA sends you an error message but checks any remaining statements for syntax errors.
- Start each group of statements that defines a DUMP, COPY, or RESTORE function with an INPUT or an OUTPUT statement.

  INPUT and OUTPUT statements delimit the different functions you want DDRXA to perform on this run. When DDRXA reads an INPUT or OUTPUT statement that follows one or more function statements, it performs the function described by the previous group of statements. It treats any function statements that follow as separate function requests.

DDRXA Prompts and Responses

DDRXA may send you some of the following prompts or responses:

```
ENTER:
```

is the prompt you receive when DDRXA is waiting for you to enter more DDRXA statements from the console.

To start a new function (dump, copy, restore, print, or type), enter a control statement. When DDRXA prompts you again, enter the next statement. Continue until you have entered all the statements needed for the function. Then press the ENTER key without any data (enter a null line). Entering a null line ends input for the function and causes DDRXA to perform the function. When DDRXA prompts you again, you may end the program or enter statements for a new function.

If you wish to end the program, and you have not entered statements for a new function, enter a null line. DDRXA then ends.

```
ENTER CYLINDER EXTENTS
ENTER:

or

ENTER BLOCK EXTENTS
ENTER:
```

is the prompt you receive when DDRXA is waiting for you to define more cylinder or block extents. After you enter the first cylinder or block extent, DDRXA prompts you for the next:

```
ENTER NEXT EXTENT OR NULL LINE
ENTER:
```

When you have entered all of the cylinder or block extents, press ENTER without any data.

```
VOLID READ IS volid1 [NOT volid2]
DO YOU WISH TO CONTINUE?
RESPOND YES, NO OR REREAD:
```

is the prompt you receive when DDRXA has read the DASD volume identifier of your input device. DDRXA waits for you to verify that the device with volume identifier *volid1* is the input device you want to use. If you entered a volume identifier on the INPUT control statement that does not match *volid1*, DDRXA displays the volume identifier you entered as *volid2*.

If you want DDRXA to continue, enter YES. If not, enter NO. If you want DDRXA to read the volume identifier again, enter REREAD.

```
DUMPING volid
COPYING volid
RESTORING volid
PRINTING volid
```

are the responses you receive when DDRXA begins a dump, copy, restore, or print function, respectively. The *volid* variable is the volume identifier described by the INPUT statement.

```
END OF VOLUME CYL xxxx HD xx, MOUNT NEXT TAPE

or

END OF VOLUME BLOCK xxxxxx, MOUNT NEXT TAPE
```

is the response you receive when DDRXA reaches the end of a tape and is waiting for you to mount the next one. DDRXA continues after you mount the next tape.

```
END OF DUMP
END OF COPY
END OF RESTORE
END OF PRINT
```

are the responses you receive when DDRXA completes a DUMP, COPY, RESTORE, or PRINT function, respectively.

```
END OF JOB
```

is the response you receive when DDRXA completes its processing.

Running the Device Support Facilities Program

The Device Support Facilities Program (ICKDSF) is provided with z/VM. It normally resides on the CMS system volume.

You can use it to perform various operations on direct access storage devices (DASD). For example, you can use ICKDSF to inspect a DASD volume for defective tracks, assign alternate tracks if defective ones are encountered, change a volume label or owner ID, or initialize a DASD volume so that it can be used in an operating system environment.

Under z/VM, ICKDSF operates on:

- A user's minidisk
- A dedicated DASD.

To perform these operations, see the detailed descriptions in *Device Support Facilities User's Guide and Reference*.

Running Production System Utility Programs

In addition to utilities provided with z/VM, you may need to run utility programs that are provided with your production system. For example, you may need to run one of the following stand-alone programs:

- The information management system/virtual storage (IMS™/VS) log tape terminator utility
- The stand-alone dump utility for your production system.

For information about these and other production system utility programs, refer to the appropriate documentation for those programs.

Appendix A. Common Operator Commands for z/VM

This appendix summarizes the commands you commonly use as a z/VM operator. The appendix includes the commands to:

- Check the status of users
- Check the status of the SSI cluster
- Check the commands and DIAGNOSE codes you can use
- Check the maximum number of spool files for the system
- Set the maximum number of users for the system
- Check whether a virtual machine is a quick dispatch virtual machine
- Check CP file system status
- Control processors and storage
- Communicate with users
- Control channel paths and local devices
- Control spool files in the print, reader, and punch queues
- Define and set time zones
- Shut down z/VM
- Collect information about system events, performance, accounting, and hardware errors
- Perform data management services.

Note: Some of the listed commands require a privilege class other than A (the IBM default class for System Operator).

Check the Status of Users

Command	Purpose
QUERY USERS	Find out how many users are logged on and dialed. In an SSI cluster, total counts for the cluster are displayed.
QUERY USERS AT *sysname*	Find out how many users are logged on and dialed on a specific member of the SSI cluster.
QUERY USERS AT ALL	Find out how many users are logged on and dialed on each member of the SSI cluster. A response will be issued from each active member.
QUERY *userid* QUERY USERS *userid*	Find out if a specific user is logged on. The response gives the device number of the user's display, or "SSI" if the user is logged on to a remote member of the SSI cluster.
QUERY *userid* EXTENDED QUERY USERS *userid* EXTENDED	Find out additional logon information for a specific user, such as the member of the SSI cluster where the user is logged on.
QUERY *userid* AT *sysname* QUERY USERS *userid* AT *sysname*	Find out if a specific user is logged on to a specific member of the SSI cluster.
QUERY *userid* AT ALL QUERY USERS *userid* AT ALL	Find out all of the logon instances for a specific multiconfiguration virtual machine in the SSI cluster. A response will be issued from each member where the user is logged on.

Command	Purpose
QUERY NAMES	Find out which users are logged on. In an SSI cluster, all of the logged-on users are displayed. The response gives the device number of each user's display. Also included are the logical unit names (lunames) owned by any active VTAM service machine that is connected to the system. Note that the response to this command could tie up your console with many thousands of names.
QUERY NAMES AT *sysname*	Find out which users are logged on to a specific member of the SSI cluster. The response gives the device number of each user's display. Note that the response to this command could tie up your console with many thousands of names.
QUERY NAMES AT ALL	Find out which users are logged on to each member of the SSI cluster. A response will be issued from each active member. The response gives the device number of each user's display. Note that the response to this command could tie up your console with many thousands of names.
INDICATE USER *userid*	Find out a user's system resource usage.
QUERY SHARE *userid*	Find out a user's scheduling share, or find out if a processor is dedicated to the user.
QUERY MAXUSERS	Find out the maximum number of users allowed to log on.
QUERY RESPOOL ALL	Find out what resource pools are defined on the system and how many users are assigned to each pool.
QUERY RESPOOL USER *userid*	Find out if a user is assigned to a resource pool.
QUERY RESPOOL POOL *poolname* MEMBERS	Find out the CPU type and CPU limit setting for a resource pool and list the assigned users.

Check the Status of the SSI Cluster

Command	Purpose
QUERY SSI STATUS	Find out the status of the SSI cluster. The response displays the state and heartbeat information for each member of the cluster.
QUERY SSI CONNECT	Find out the status of the connections to the other members of the SSI cluster.

Check the Commands and DIAGNOSE Codes

Command	Purpose
COMMANDS	Find out what commands you can enter from the ID you are using.
QUERY D8ONECMD	Find out if issuing multiple CP commands through DIAGNOSE code X'08' is allowed.
QUERY DIAGNOSE	Find out the available DIAGNOSE instructions and status.

Check the Maximum Number of Spool Files for the System

Command	Purpose
QUERY MAXSPOOL	Find out the maximum number of spool files for the system.

Set the Maximum Number of Users for the System

Command	Purpose
SET MAXUSERS *nnnn*	Set the maximum number of users who can log on to the system.

Check Whether a Virtual Machine Is a Quick Dispatch Virtual Machine

Command	Purpose
QUERY QUICKDSP *userid*	Find out if the virtual machine is a quick dispatch virtual machine.

Check CP File System Status

Command	Purpose
CPACCESS	Identifies a CMS-formatted minidisk to CP and makes the files on that minidisk available to CP by establishing a file mode letter for the files.
CPRELEASE	Releases a minidisk that was previously accessed using the CPACCESS command or the CP_ACCESS statement in the system configuration file.
CPLISTFILE	Lists information on the files on a minidisk that you identified to CP by using either a CPACCESS command or a CP_ACCESS statement in the system configuration file.
CPTYPE	Causes CP to display selected lines of a file on a minidisk that you identified to CP with either a CPACCESS command or on the CP_ACCESS statement in the system configuration file.
QUERY CPDISKS	Displays what minidisks CP has currently accessed.

Control Processors

Command	Purpose
INDICATE LOAD	Find out which processors are active.
QUERY PROCESSORS	Find out how CP is using the real processors.

Control Storage

Command	Purpose
QUERY STORAGE	Find out the size of online real storage.
SET STORAGE	Increase the size of online real storage.
QUERY FRAMES	Find out how real storage is allocated.
QUERY SXSSTORAGE	Find out what host logical storage addresses are used for the system execution space.
QUERY SXSPAGES	Find out the status of system execution space pages.
QUERY MDC	Find the status of minidisk caching allocation and usage for storage.

Communicate with z/VM Users

Command	Purpose
QUERY LOGMSG	Display the system log message.
SET LOGMSG *nn* * *text*	Change the system log message so that all users see the text of this line when they log on.
SET LOGMSG *nn text*	Change the system log message so that users must enter QUERY LOGMSG to read the text of this line.
SET LOGMSG *nn* NULL	Delete a line from the system log message.
SET LOGMSG NULL	Delete the entire system log message.
MESSAGE ALL *text*	Send a message to all users on the system.
MESSAGE ALL AT ALL *text*	Send a message to every user on every system in the SSI cluster.
MESSAGE ALL AT *sysname text*	Send a message to every user on a specific system in the SSI cluster.
MESSAGE *userid text*	Send a message to a specific user on the system.
MESSAGE *userid* AT ALL *text*	Send a message to a specific user on any system in the SSI cluster.
MESSAGE *userid* AT *sysname text*	Send a message to a specific user on a specific system in the SSI cluster.
WARNING ALL *text*	Send a warning message to all users on the system.
WARNING ALL AT ALL *text*	Send a warning message to every user on every system in the SSI cluster.
WARNING ALL AT *sysname text*	Send a warning message to every user on a specific system in the SSI cluster.
WARNING *userid text*	Send a warning message to a specific user on the system.
WARNING *userid* AT ALL *text*	Send a warning message to a specific user on any system in the SSI cluster.
WARNING *userid* AT *sysname text*	Send a warning message to a specific user on a specific system in the SSI cluster.

Control Channel Paths and Programs

Command	Purpose
HALT *rdev*	Terminate execution of an active channel program on device *rdev*.
QUERY PATHS TO *rdev*	Find out the status of all channel paths installed (with IOCP) to device *rdev*.
QUERY CHPID *nn*	Find out the status of channel path *nn* for all devices to which it is installed.
QUERY CHPIDS	Used to display all 256 of the machine's channel paths and their physical status.
VARY OFFLINE PATH *nn* FROM *rdev*	Logically remove channel path *nn* from device *rdev*.
VARY ONLINE PATH *nn* TO *rdev*	Make channel path *nn* logically available to device *rdev*.

Control Dynamic Switching Devices

Command	Purpose
QUERY SWCH ALL	Find out the status of all dynamic switching devices.
ATTACH *rdev userid vdev*	Dedicate dynamic switching device *rdev* to user *userid* as virtual device *vdev*.
DETACH *rdev userid*	Remove dynamic switching device *rdev* from user *userid*.
VARY ONLINE SUBCHANNEL *rdev*	Enable the subchannel for I/O interrupts
VARY OFFLINE SUBCHANNEL *rdev*	Disable the subchannel for I/O interrupts

Control Local Displays, 3270 Printers, and Communication Lines

Command	Purpose
QUERY *rdev*	Find out the status of files to be printed on the printer *rdev*.
QUERY GRAF ALL	Find out the status of all displays and 3270 printers.
QUERY LINES ALL	Find out the status of all communication lines.
VARY ONLINE *rdev*	Make display, 3270 printer, or communication lines *rdev* available to the system.
ENABLE *rdev*	Enable display, 3270 printer, or communication lines *rdev* to be used by a z/VM user.
ATTACH *rdev userid vdev*	Dedicate display, 3270 printer, or communication lines *rdev* to user *userid* as virtual device *vdev*.
DETACH *rdev userid*	Remove display, 3270 printer, or communication lines *rdev* from user *userid*.
DISABLE *rdev*	Disable display, 3270 printer, or communication lines *rdev* from being used by a z/VM user.
VARY OFFLINE *rdev*	Make a display, 3270 printer, or communication lines *rdev* unavailable to the system.
SET PF*nn* COPY *rdev*	Set up 3270 printer *rdev* to copy a display screen with the PF*nn* key. (This does not apply to communication lines.)
TRANSFER PRT CLASS *c*|*userid1*|all to|from all|* *userid2* rdr|prt|pun	Transfer spool files from the 3270 printer.
SET RDEVICE	Define device characteristics.
VARY ONLINE SUBCHANNEL *rdev*	Enable the subchannel for I/O interrupts
VARY OFFLINE SUBCHANNEL *rdev*	Disable the subchannel for I/O interrupts

Control Local Direct Access Storage Devices (DASD)

Command	Purpose
COMMIT DEVICE *rdev*	Transfer cache fast write data from the cache to addressed device *rdev*.
DESTAGE SUBSYSTEM *rdev*	Force transfer of modified data in subsystem cache or NVS to backing storage device *rdev*.

Command	Purpose
DISCARD PINNED ALL|CACHEFW *rdev*	Discard pinned data in device *rdev* in subsystem cache or nonvolatile storage (NVS).
DUPLEX *rdev* ON|SUSPEND|OFF	Perform duplex control operations (such as establishing or discontinuing a duplex pair).
QUERY DUPLEX *rdev*	Display status of duplex pairs of devices attached to a cache storage subsystem.
QUERY CACHEFW *rdev*	Display the cache fast write status for cache storage subsystem device *rdev*.
QUERY DASDFW *rdev*	Display the DASD fast write status of device *rdev* attached to the cache storage subsystem.
QUERY FENCES *rdev-rdev*	Display a list of internal fences for device *rdev* created by an IBM DASD subsystem.
QUERY NVS *rdev*	Display the status of NVS device *rdev* associated with a DASD storage subsystem.
QUERY PINNED DEVICE|SUBSYSTEM *rdev1...rdevn*	Display the data tracks of one or more DASD *rdev1...rdevn* or the total amount of subsystem pinned and NVS storage data in *rdev1...rdevn*.
QUERY PENDING COMMANDS ALLUSERS *rdev*|ALL	Display the commands for device *rdev* or for all devices (ALL) that you have entered and, optionally, that other users have entered (ALLUSERS) for which the associated asynchronous function has not yet completed.
QUERY RSAW CLUSTER 0|1|ALL *rdev*	Generate and display the RSAW for storage cluster 0, 1, or all in the DASD subsystem for device *rdev*.

SET CACHE DEVICE ON|OFF
SET CACHE SUBSYSTEM ON|OFF|FORCEOFF FOR *rdev*
	Activate or deactivate caching by device or subsystem for device *rdev* in the IBM DASD subsystem. FORCEOFF indicates the cache is forced to be unavailable to device *rdev*.

SET CACHEFW SUBSYSTEM ON|OFF *rdev*	Activate or deactivate the cache fast write function in a DASD subsystem for device *rdev*.
SET DASDFW DEVICE ON|OFF|FORCEOFF *rdev1...rdevn*	Activate or deactivate the DASD fast write function for devices *rdev1...rdevn* in a DASD subsystem.
SET NVS SUBSYSTEM ON|OFF *rdev*	Make NVS available or unavailable within a DASD subsystem for device *rdev*.
QUERY *rdev*	Find out the status of a DASD *rdev*.
QUERY DASD ALL	Find out the status of all DASD.
QUERY CACHE *rdev*	Find out the caching status of a DASD *rdev*.
SET CACHE SUBSYSTEM ON *rdev*	Make the cache available to the subsystem for a DASD *rdev*.
SET CACHE DEVICE ON *rdev*	Activate the cache for a DASD *rdev*.
SET CACHE SUBSYSTEM OFF *rdev*	Make the cache unavailable to the subsystem for a DASD *rdev*.
SET CACHE DEVICE OFF *rdev*	Deactivate the cache for a DASD *rdev*.
QUERY DASD *volid*	Locate the DASD volume *volid*, and find out the status of the DASD on which it is mounted.
VARY ONLINE *rdev*	Make a DASD *rdev* available to the system.
ATTACH *rdev userid vdev*	Dedicate a DASD *rdev* to user *userid* as virtual device *vdev*.

Command	Purpose
ATTACH *rdev userid vdev control*	Dedicate a DASD *rdev* to user *userid* as virtual device *vdev* with the given level of *control*: NOCTL, DEVCTL, SYSCTL. **Note:** Deactivate, activate, and force deactivate DASD fast write are accepted for DEVCTL.
DETACH *rdev userid*	Remove a DASD *rdev* from user *userid*.
ATTACH *rdev* SYSTEM *volid*	Attach the system volume *volid* on a DASD *rdev* to the system.
DETACH *rdev* SYSTEM	Detach the system volume on a DASD *rdev* from the system.
VARY OFFLINE *rdev*	Make a DASD *rdev* unavailable to the system.
QUERY ALLOC ALL	Display a list of CP-owned system volumes that contain temporary disk, directory, paging, or spooling space.
QUERY ALLOC PAGE ALL	Display a list of all CP-owned system volumes that contain paging space.
QUERY ALLOC SPOOL ALL	Display a list of all CP-owned system volumes that contain spooling space.
QUERY ALLOC TDISK ALL	Display a list of all CP-owned system volumes that contain temporary disk space.
QUERY ALLOC DRCT ALL	Display a list of all CP-owned system volumes that contain directory space.
QUERY ALLOC MAP ALL	Display the allocation mapping of all CP-owned system volumes.
QUERY SYSTEM ALL	Find out how many users are using minidisks or temporary disks on system volumes.
CPFMTXA *vdev volid*	Under CMS, format and allocate volume *volid* on virtual device *vdev* for system use.
SET DUMP DASD IPL CP	Define a DASD as the dump device.
SET RDEVICE	Define device characteristics.
VARY ONLINE SUBCHANNEL *rdev*	Enable the subchannel for I/O interrupts
VARY OFFLINE SUBCHANNEL *rdev*	Disable the subchannel for I/O interrupts

Control Local Tape Drives

Command	Purpose
QUERY *rdev*	Find out the status of tape drive *rdev*.
QUERY TAPES ALL	Find out the status of all tape drives.
VARY ONLINE *rdev*	Make tape drive *rdev* available to the system.
ATTACH *rdev userid vdev*	Dedicate tape drive *rdev* to user *userid* as virtual device *vdev*.
DETACH *rdev userid*	Remove tape drive *rdev* from user *userid*.
DETACH *rdev userid* LEAVE	Remove tape drive *rdev* from user *userid*, but do not rewind or unload the tape.
DETACH *rdev userid* UNLOAD	Remove tape drive *rdev* from user *userid* and rewind and unload the tape.

Command	Purpose
VARY OFFLINE *rdev*	Make a tape drive *rdev* unavailable to the system.
SET DUMP *rdev* IPL CP	Define tape drive *rdev* as the dump device.
GIVE *vdev1* to *userid* as *vdev2* RETURN R/W LEAVE	Transfer control of the tape drive *vdev1* to the receiver *userid* in READ/WRITE mode as attached virtual device *vdev2*. The tape drive is to be returned in the position it was in when it was detached by the user
SET RDEVICE	Define device characteristics.
VARY ONLINE SUBCHANNEL *rdev*	Enable the subchannel for I/O interrupts
VARY OFFLINE SUBCHANNEL *rdev*	Disable the subchannel for I/O interrupts

Control Local Channel-to-Channel Adapters (CTCAs)

Command	Purpose
QUERY *rdev*	Find out the status of CTCA *rdev*.
QUERY CTCA ALL	Find out the status of all CTCAs.
VARY ONLINE *rdev*	Make a CTCA *rdev* available to the system.
ATTACH *rdev userid vdev*	Dedicate a CTCA *rdev* to user *userid* as virtual device *vdev*.
DETACH *rdev userid*	Remove a CTCA *rdev* from user *userid*.
VARY OFFLINE *rdev*	Make a CTCA *rdev* unavailable to the system.
VARY ONLINE SUBCHANNEL *rdev*	Enable the subchannel for I/O interrupts
VARY OFFLINE SUBCHANNEL *rdev*	Disable the subchannel for I/O interrupts

Control Local Open Systems Adapters (OSAs)

Command	Purpose
QUERY *rdev*	Find out the status of OSA *rdev*.
QUERY OSA ALL	Find out the status of all OSAs.
VARY ONLINE *rdev*	Make an OSA *rdev* available to the system.
ATTACH *rdev userid vdev*	Dedicate an OSA *rdev* to user *userid* as virtual device *vdev*.
DETACH *rdev userid*	Remove an OSA *rdev* from user *userid*.
VARY OFFLINE *rdev*	Make an OSA *rdev* unavailable to the system.
VARY ONLINE SUBCHANNEL *rdev*	Enable the subchannel for I/O interrupts
VARY OFFLINE SUBCHANNEL *rdev*	Disable the subchannel for I/O interrupts

Control Local Printers

Command	Purpose
QUERY *rdev*	Find out the status of printer *rdev*.
QUERY *prtid*	Find out the status of the virtual machine *prtid* that controls the print server printer.

Command	Purpose
QUERY PRINTER SECLABEL	Find out the security labels of the spool files in the printer queue.
QUERY UR ALL	Find out the status of all printers (and all other unit record devices).
VARY ONLINE *rdev*	Make printer *rdev* available to the system.
ATTACH *rdev userid vdev*	Dedicate printer *rdev* to user *userid* as virtual device *vdev*.
DETACH *rdev userid*	Remove printer *rdev* from user *userid*.
VARY OFFLINE *rdev*	Make a printer *rdev* unavailable to the system.
QUERY IMG	Find out what 3800 image libraries are available.
START UR *rdev* USER SYSTEM *options*	Start printing any user's files on 3800 *rdev* with the given options: CLASS *abcdefgh* FORM *operform* CHARS *cccc* FCB *ffff* MANUAL, AUTO SEP, NOSEP ANY3800, BEG3800, NO3800 FLASH *name*, FLASH OFF IMAGE *name* HOLD, PURGE SECLABEL.
START UR *rdev* USER *userid options*	Start printing only the files owned by user *userid* on a 3800 *rdev* with the given options.
LOADBUF *rdev* UCS *name* VER	Load the UCS buffer *name* for impact printer *rdev* and verify it.
LOADBUF *rdev* FCB *name*	Load the FCB *name* for impact printer *rdev*.
START *rdev* USER SYSTEM *options*	Start printing any user's files on impact printer *rdev* with the given options: CLASS *abcdefgh* FORM *operform* DEST *dest* CHARS *cccc* FCB *ffff* AFP, NOAFP IMAGE *name* SETUP, MANUAL, AUTO SEP, NOSEP NO3800, BEG3800, ANY3800 INDEX *nn* SECLABEL.
START UR *rdev* USER *userid options*	Start printing only the files owned by user *userid* on impact printer *rdev* with the given options.
DRAIN UR *rdev*	Stop printing on printer *rdev* after the current file.
DRAIN *prtid*	Stop printing on the AFP identified as *prtid* that is controlled by the print server virtual machine, after the current file.

Command	Purpose
FLUSH *rdev* HOLD	Stop printing on printer *rdev* immediately; put the current file on system hold.
FLUSH *rdev* ALL	Stop printing on printer *rdev* immediately; purge all copies of the current file.
FLUSH *prtid* HOLD	Stop printing on the print server printer controlled by virtual machine *prtid* immediately; put the current file in system hold.
FLUSH *prtid* ALL	Stop printing on the print server printer controlled by virtual machine *prtid* immediately; purge all copies of the current file.
BACKSPACE *rdev nnn*	Backspace the file printing on printer *rdev nnn* number of pages.
BACKSPACE *rdev* FILE	Backspace the file printing on printer *rdev* to the beginning of the file.
BACKSPACE *rdev nnn* EOF	Backspace the file printing on printer *rdev* to a point *nnn* pages from the end of the file.
SPACE *rdev*	Cause the file printing on printer *rdev* to print single-spaced.
REPEAT *rdev nnn*	Cause the file printing on printer *rdev* to print *nnn* more copies.
SET RDEVICE	Define device characteristics.
VARY ONLINE SUBCHANNEL *rdev*	Enable the subchannel for I/O interrupts
VARY OFFLINE SUBCHANNEL *rdev*	Disable the subchannel for I/O interrupts

Control Spool Files in the Print Queue

Command	Purpose
QUERY FILES	Find out how many files are in the print queue.
QUERY PRINTER	List the files in the print queue.
QUERY PRINTER EXP	Display expanded information about the files in the print queue.
QUERY PRINTER PSF	Display additional information about the files in the print queue.
ORDER *userid* PRINTER *nnnn*	Put file *nnnn* at the top of the print queue.
CHANGE *userid* PRINTER *nnnn* SYS	Put file *nnnn* (in the print queue) on system hold.
CHANGE *userid* PRINTER *nnnn* NOSYS	Release file *nnnn* (in the print queue) from system hold.
CHANGE *userid* PRINTER *nnnn* COPY *cpy*	Change the number of copies of file *nnnn* (in the print queue) to *cpy*.
CHANGE *userid* PRINTER *nnnn* SECLABEL *ssssssss*	Change the security label of file *nnnn* (in the print queue) to *ssssssss*.
CHANGE *userid* PRINTER *nnnn* DEST *dest*	Change the destination value of file *nnnn* (in the print queue) to *dest*.
CHANGE *userid* PRINTER *nnnn* UNCONV	Cancel all preprocessing to file *nnnn* (in the print queue) in order to use different options.
PURGE *userid* PRINTER *nnnn*	Purge file *nnnn* from the print queue.
TRANSFER PRT CLASS *c*|*userid1*|all to|from all|* *userid2* rdr|prt|pun	Transfer spool files from the print queue.

Control Spool Files in the Reader Queue

Command	Purpose
QUERY FILES	Find out how many files are in the reader queue.
QUERY READER ALL	List the files in the reader queue. Note that the response to this command could tie up your console with many thousands of reader files.
QUERY READER SECLABEL	Find out the security labels of the spool files in the reader.
CHANGE *userid* READER *nnnn* SECLABEL *ssssssss*	Change the security label of file *nnnn* (in *userid*'s reader queue) to *ssssssss*.
ORDER *userid* READER *nnnn*	Put file *nnnn* at the top of the reader queue.
PURGE *userid* READER *nnnn*	Purge file *nnnn* from the reader queue.
TRANSFER RDR CLASS *c*|*userid1*|all to|from all|* *userid2* rdr|prt|pun	Transfer spool files from the reader queue.

Control Spool Files in the Punch Queue

Command	Purpose
QUERY FILES	Find out how many files are in the punch queue.
QUERY PUNCH	List the files in the punch queue.
QUERY PUNCH ALL	Display additional information about the files in the punch queue.
QUERY PUNCH SECLABEL	Find out the security labels of the spool files in the punch queue.
ORDER *userid* PUNCH *nnnn*	Put file *nnnn* at the top of the punch queue.
CHANGE *userid* PUNCH *nnnn* SYS	Put file *nnnn* (in the punch queue) in system hold.
CHANGE *userid* PUNCH *nnnn* NOSYS	Release file *nnnn* (in the punch queue) from system hold.
CHANGE *userid* PUNCH *nnnn* COPY *cpy*	Change the number of copies of file *nnnn* (in the punch queue) to number *cpy*.
CHANGE *userid* PUNCH *nnnn* SECLABEL *ssssssss*	Change the security label of file *nnnn* (in the punch queue) to *ssssssss*
PURGE *userid* PUNCH *nnnn*	Purge file *nnnn* from the punch queue.
TRANSFER PUN CLASS *c*|*userid1*|all to|from all|* *userid2* rdr|prt|pun	Transfer spool files from the punch queue.

Defining and Setting Time Zones

Command	Purpose
QUERY TIME	Displays the following: Current time. Time zone. Day of the week. Date. Connect time for the current session. Processor time for the current session.

Command	Purpose
QUERY TIME OFFSET	Find out the difference between the system's current time zone and Coordinated Universal Time (UTC).
QUERY TIMEZONE	Displays the list of active and inactive time zone definitions on your system.
DEFINE TIMEZONE	Defines a new time zone or changes an existing time zone definition.
SET TIMEZONE	Changes the system's active time zone ID and time zone offset. **Note:** Enter this command only when changing to standard or daylight savings time.

Shut Down z/VM

Command	Purpose
WARNING ALL *text*	Send a warning message to all users.
DRAIN UR ALL	Drain all unit record devices.
SHUTDOWN	Shut down z/VM.

Load an Alternate CP Module

Command	Purpose
WARNING ALL *text*	Send a warning message to all users.
DRAIN UR ALL	Drain all unit record devices.
SHUTDOWN REIPL MODULE *modulename*	Shut down z/VM and bring it up with another CP module residing on the same minidisk as the current CP module. To use a CP module on a different minidisk, enter: SHUTDOWN REIPL MDISK *userid vdev* MODULE *modulename*

Collect Information about System Events

Command	Purpose
SPOOL CONSOLE START	Start logging console activity in a console log.
* text	Add comment *text* to a console log.
CLOSE CONSOLE	Ready a console log for printing.
SPOOL CONSOLE STOP	Stop logging console activity in a console log.

Collect Information about System Performance

Command	Purpose
INDICATE LOAD	Find out what the operating load is for the system you are logged on.
INDICATE LOAD AT *sysname*	Find out what the operating load is for a specific system in the SSI cluster.
INDICATE I/O	Find out the users that are waiting for devices.
INDICATE PAGING WAIT	Find out the users that are waiting for pages.

Command	Purpose
INDICATE QUEUES	Find out the users that are in queue for dispatching.
INDICATE QUEUES EXP	Find out the users that are in queue for dispatching with expanded information about their priority, processor affinity, and status.
INDICATE USER *userid*	Find out a user's system resource use.
QUERY SCMEASURE	Display whether subchannel measurement data is being collected for a specific device or range of devices.
SET SCMEASURE	Enable or disable the collecting of subchannel measurement data for a specific device or range of devices.

Collect Accounting Information

Command	Purpose
RECORDING ACCOUNT ON	Start creating accounting records.
ACNT ALL	Produce accounting records for all users and the system.
QUERY RECORDING	Find out the status of accounting.
RETRIEVE ACCOUNT	Start retrieval of the accounting records (issued by the accounting virtual machine).

Collect Hardware Error Information

Command	Purpose
RECORDING EREP ON	Start creating EREP records.
QUERY RECORDING	Find out the status of error recording.
RETRIEVE EREP	Start retrieval of the EREP records (issued by the EREP virtual machine).
CPEREPXA	Process EREP records (issued by the EREP virtual machine).
SYNCMDRS DASD|TAPE|PRINTER|ALL	Create Miscellaneous Data Records (MDRs) for the specified devices.
QUERY MITIME	Find out how often CP checks for missing interrupts.
SET MITIME *dev mm:ss*	Change the rate at which CP checks for missing interrupts for device type *dev* (DASD, GRAF, TAPE, UR, MISC, SWCH, or real device number).

Collect Symptom Record Information

Command	Purpose
RECORDING SYMPTOM ON	Start creating symptom records.
QUERY RECORDING	Find out the status of symptom record recording.
RETRIEVE SYMPTOM	Start retrieval of the symptom records (issued by the symptom record recording virtual machine).

Dump and Load Standard Spool Files and System Data Files

Command	Purpose
SPXTAPE SCAN *vdev* SPOOL REWIND	Scan a tape mounted on tape drive *vdev* for all dumped standard spool files and system data files and rewind the tape.
SPXTAPE SCAN *vdev* STD REWIND	Scan a tape mounted on tape drive *vdev* for all dumped standard spool files and rewind the tape.
SPXTAPE SCAN *vdev* SDF REWIND	Scan a tape mounted on tape drive *vdev* for all dumped system data files and rewind the tape.
SPXTAPE DUMP *vdev* READER	Dump all reader spool files to a tape mounted on tape drive *vdev*.
SPXTAPE LOAD *vdev* READER	Load all reader spool files from a tape mounted on tape drive *vdev*.
SPXTAPE DUMP *vdev* PRINTER	Dump all printer spool files to a tape mounted on tape drive *vdev*.
SPXTAPE LOAD *vdev* PRINTER	Load all printer spool files from a tape mounted on tape drive *vdev*.
SPXTAPE DUMP *vdev* PUNCH	Dump all punch spool files to a tape mounted on tape drive *vdev*.
SPXTAPE LOAD *vdev* PUNCH	Load all punch spool files from a tape mounted on tape drive *vdev*.
QUERY NSS	Find out what named saved systems and saved segments are available.
SPXTAPE DUMP *vdev* NSS	Dump all named saved systems and saved segments to a tape mounted on tape drive *vdev*.
SPXTAPE LOAD *vdev* NSS	Load all named saved systems and saved segments from a tape mounted on tape drive *vdev*.
QUERY IMG	Find out what image libraries are available.
SPXTAPE DUMP *vdev* IMG	Dump all image libraries to a tape mounted on tape drive *vdev*.
SPXTAPE LOAD *vdev* IMG	Load all image libraries from a tape mounted on tape drive *vdev*.
QUERY NLS	Find out what national language support files are available.
SPXTAPE DUMP *vdev* NLS	Dump all national language support files to a tape mounted on tape drive *vdev*.
SPXTAPE LOAD *vdev* NLS	Load all national language support files from a tape mounted on tape drive *vdev*.
QUERY TRF	Find out what system trace files are available.
SPXTAPE DUMP *vdev* TRF	Dump all system trace files to a tape mounted on tape drive *vdev*.
SPXTAPE LOAD *vdev* TRF	Load all system trace files from a tape mounted on tape drive *vdev*.

Commands with Security Label Options

Command	Purpose
QUERY READER SECLABEL	Find out the security labels of the spool files in the reader.
QUERY PRINTER SECLABEL	Find out the security labels of the spool files in the printer queue.
QUERY PUNCH SECLABEL	Find out the security labels of the spool files in the punch queue.
CHANGE *userid* READER *nnnn* SECLABEL *ssssssss*	Change the security label of file *nnnn* (in the reader queue) to *ssssssss*.

Command	Purpose
CHANGE *userid* PRINTER *nnnn* SECLABEL *ssssssss*.	Change the security label of file *nnnn* (in the print queue) to *ssssssss*.
CHANGE *userid* PUNCH *nnnn* SECLABEL *ssssssss*	Change the security label of file *nnnn* (in the card punch or the punch queue) to *ssssssss*.
LOGON *userid* SECLABEL *ssssssss*	Log on a user with a specific security label *ssssssss*.
START *rdev* USER SYSTEM SECLABEL *ssssssss*	Start printing any user's files on printer *rdev* with the security label *ssssssss*.

Appendix B. Printer and Print File Status

This appendix explains the way z/VM presents both printer and printer spool file status. It discusses the responses to the QUERY command for 3800 and impact printers and for files in the print queue.

Understanding Printer Status

To display information about a printer, enter:

```
query rdev
```

where *rdev* is the real device number of the printer.

In response, CP displays the printer's status. Figure 13 on page 263 shows two sample responses, one for a 3800 printer and one for an impact printer.

Sections following Figure 13 on page 263 describe its contents in detail.

```
Sample QUERY
rdev Response for an Active 3800 Printer:
PRT   0B32 DRAINING   SYSTEM   CLASS ABCD        SEC JOBARO
PRT   0B32 FORM TTABT6   MANUAL SEP   ANY3800  LIMIT
PRT   0B32 FLASH      IMAGE IMAG3800 CHARS GF10 FCB 8    HOLD   LPP 060
PRT   0B32 DEST BLDG1    DEST - NONE - DEST - NONE - DEST - NONE -
PRT   0B32 AFP     MARK
PRT   0B32 PRINTING MAINT    FILE 1603 RECS 4436  COPY  001 SEQ 001 CL A PRT

Sample QUERY
rdev Response for an Active Impact Printer:
PRT   0003 SETUP REQ  MAINT    CLASS EFGH        SEC JOBARO
PRT   0003 FORM STANDARD SETUP  SEP   NO3800   DEFFCB LIMIT NONE
PRT   0003 FOLD   IMAGE IMAGIMP1 CHARS FCB FCB8           LPP  068
PRT   0003 DEST BLDG1    DEST - NONE - DEST - NONE - DEST - NONE -
PRT   0003 NOAFP
PRT   0003 PRINTING MAINT    FILE 1610 RECS 5328  COPY  003 SEQ 001 CL A PRT
```

Figure 13. Sample QUERY Response for a Printer

Device Type and Real Device Number

In Figure 13 on page 263, the first two fields of each line show the *device type* and the *real device number*. The sample shows one response for a 3800 printer with real device *number* B32 and another for an impact printer with real device number 003.

Printer State

The *printer state* indicates an active printer's basic state. In Figure 13 on page 263, the 3800's printer state is DRAINING; the impact printer's is SETUP REQ. The printer state may be any of the following:

- **STARTED**—The printer is either ready to print or is actually printing any available files. An available file is a file with characteristics that match those the printer is set up to handle.

  If there is an active file on the printer, CP displays information about it. This information appears after the other printer status.

- **DRAINING**—The printer is getting ready to go idle by finishing the printing of any active file.
- **DRAINED**—The printer is idle and not available to print spool files.
- **INT REQ**—Your intervention is required to handle a condition such as a paper jam, an open door, or lack of paper.

- **SETUP REQ** (impact printers only)—The printer is waiting for you to check the alignment of the paper on the printer.
- **MOUNT REQ**—The printer is waiting for you to change the paper (mount a new form).

You change the printer's state by entering the START or DRAIN commands, by responding to the mount or setup request, or by handling the condition for which your intervention is required.

Assigned User

The *assigned user* shows you whose spool files the printer will print. In Figure 13 on page 263, the 3800 is assigned to SYSTEM. The SYSTEM setting allows the printer to print files belonging to any user. Otherwise, the printer prints only the files belonging to the user whose user ID is displayed. In Figure 13 on page 263, the impact printer is assigned to user MAINT and so would print only MAINT's files.

You change the printer's assigned user with the START command.

Spool File Class

The *spool file classes* field (which appears after the keyword CLASS) shows the class or classes of spool files the printer will print. The *operator form name* (after FORM) indicates the type of paper that should be mounted on the printer. The printer prints only spool files with matching class, security label, and operator form name.

Each installation classifies its printed output differently, but it is common to use a combination of class and form to separate different kinds of print files from each other. For example, a job of one print class/form differs from a job in another print class/form in at least one of the following ways:

- The type of printer (for example, impact or 3800) it is printed on
- The 3800 stacker feature (burster-trimmer-stacker or continuous forms stacker) to be used
- The kind of paper (size and number of parts) it is printed on
- The security label of the file
- The forms overlay required, if any
- The set of characters (print train image or font) used to print it
- The spacing in lines per inch (forms control buffer) used to print it.

A spool file class is an alphanumeric character (A through Z, 0 through 9). CP displays up to eight classes, or an asterisk (*), which indicates that any class file will print. In the example, the 3800 printer prints files of four classes, classes A, B, C, and D, in that order. The impact printer prints class E, F, G, and H files, in that order.

You change the print class with the START command to correspond to the kind of files you want to print next.

Security Label

The SEC field contains a 1 - 8 character alphanumeric value that corresponds to the *security label* of spool files that can be printed on this printer. If a security label is not assigned to the printer, the SEC field is followed by the word NONE. No files can be printed until the printer is STARTed with a valid security label.

In Figure 13 on page 263, the printers will print only files with a security label of JOBARO.

Security labels are used to control print jobs only if an external security manager (ESM) is installed and security label checking is enabled. If you are authorized by the ESM, use the START command to change the security label of the printer to correspond to the security label of the files you want to print next.

Form Name

The operator form name corresponds to the type of paper on which a spool file should print. CP displays either an operator form name that your installation has established, or an asterisk (*), which indicates that files with any form will print. In Figure 13 on page 263, the 3800 printer will print files with the operator form name TTABT6. The impact printer, on the other hand, will print only the files that require standard paper.

You change the form name with the START command to correspond to the kind of files you want to print next.

Form Mode

The *form mode* shows how the printer acts when no more files of the current form are left to print.

The MANUAL setting allows you to choose the next form you want to mount on the printer. The printer idles until another file of the current form joins the print queue or until you start the printer with a new form. Figure 13 on page 263 shows the 3800 printer with the MANUAL setting.

When AUTO is set, CP automatically chooses the next form for you. When there are no more files in queue of the current form, CP starts the printer with the form of the next available file. CP then prompts you to mount the new form.

Both 3800 and impact printers may display MANUAL and AUTO settings. But SETUP mode applies only to impact printers; the example shows what it looks like.

In SETUP mode, the printer waits for you to align the paper based on the position of the alignment page it has printed out. The alignment page is the first page of the active file, except that the letters on the page appear as Xs and the numbers appear as 9s. Pressing the START and STOP buttons on the printer causes the printer to print out another alignment page. The printer stays in SETUP mode until you enter a START command without the SETUP option.

You change the printer's form mode with the START command.

Separator Page Mode

The *separator page mode* (in both examples in Figure 13 on page 263, SEP) shows whether the printer prints separator pages between files. The SEP setting causes the printer to produce separator pages. These header and trailer pages make it easier to burst the output. When NOSEP is set, the printer prints one file after another without separation.

You change the printer's separator page mode with the START command.

Note: If an ESM is installed and security label checking is enabled, the NOSEP option is not valid. Files must be printed with separator pages.

3800 Load CCW Mode

The *3800 load CCW mode* shows you whether or not the printer will print files with 3800 load CCWs. Figure 13 on page 263 shows ANY3800 mode for the 3800, and NO3800 mode for the impact printer.

A *3800 load CCW* is a special instruction imbedded in a spool file. It tells a 3800 to load data that defines the spacing, characters, or copy modification data that the printer is to use to print the file. CP keeps track of whether a spool file has any 3800 load CCWs, and if so, whether they appear only at the beginning, or throughout the file.

The ANY3800 setting allows the printer to print any kind of print file, whether it has 3800 load CCWs or not.

The NO3800 setting allows the printer to print only those files that have no 3800 load CCWs.

The BEG3800 setting allows the printer to print files with no 3800 load CCWs, or files with 3800 load CCWs that appear only at the beginning of the file.

ANY3800 is the normal setting for a 3800, while NO3800 is the normal setting for an impact printer. If you must, you can print BEG3800 and ANY3800 files on an impact printer, but doing so does not take advantage of any required 3800 features. An impact printer ignores 3800 load CCWs.

You change the printer's 3800 load CCW mode with the START command.

DEFFCB/FILEFCB

The DEFFCB specification tells CP to use the default FCB, which has been loaded to the printer. The default FCB can be loaded to the printer using the LOADBUF command or the START command.

The FILEFCB specification tells CP to send LOAD FCB CCWs that were imbedded in the SPOOL file to the printer.

DEFFCB and FILEFCB are only valid for non-1403 impact printers. You can change the printer's LOAD FCB CCW load option with the START command.

LIMIT

The LIMIT specification shows whether CP has been instructed to select files for the printer based on the overall size of the spool file. If LIMIT is not displayed as NONE, CP will multiply the copy count of a spool file by its record count and if the number exceeds the number displayed after LIMIT, the file will not be selected for printing. You can change the LIMIT value by entering the START command.

Forms Overlay Name (3800 Only)

The *forms overlay name* (which appears after the FLASH keyword) shows you whether or not a 3800 printer prints files with a forms overlay.

Some print files may require the use of a forms overlay, a feature of the 3800. When this feature is active, the 3800 flashes an image onto selected pages of a print file through a film negative mounted in a frame. In the sample 3800 status, Figure 13 on page 263, the keyword FLASH is followed by blanks, which indicates that the printer prints only files that do not require a forms overlay. CP does not display forms overlay status for an impact printer.

You change a 3800's forms overlay name with the START command.

Image Library Name

The *image library name* (which appears after the IMAGE keyword) indicates which image library a printer will use.

An *image library* is a set of modules that define the spacing, characters, and copy modification data you may select. When a spool file that is printing calls for one of these modules, the printer looks for it in the specified image library. Figure 13 on page 263 shows IMAG3800 for the 3800 printer and IMAGIMP1 for the impact printer.

You change the image library name with the START command.

Default Character Set and FCB Names

The *default character set name* (which appears after the keyword CHARS) and *default FCB name* (appearing after the keyword FCB) show the characters and spacing a printer will use for files that do not specify a character set or FCB.

If spool files do not specify their own character set(s) or FCB, the printer uses these default values. The printer also uses these defaults to print the separator pages between files, if any. The default character set and FCB usually correspond to the class or form, or both, set up on the printer.

The number of lines to be printed on the separator page is determined by the FCB loaded on the printer. The default separator page contains 58 lines of data. If the page length defined by the FCB is less than the default separator page length, the separator page data must be customized in order to fit on a single page. This may be done using CP Exit points that are provided in separator page processing. See the *z/VM: CP Exit Customization* for details.

In Figure 13 on page 263, both printers use a character set of GF10 and a vertical spacing of 8 lines per inch. (This information is not displayed for a 1403-type printer unless it is equipped with a UCS feature.)

You change a printer's default character set and FCB name with the START command.

Load Check Action (3800 Only)

The *load check action* setting shows what a 3800 printer does when it encounters a load check error while printing a file.

A *load check error* indicates a problem with a spool file's 3800 load CCW. The HOLD setting causes CP to put the file in system hold. The PURGE setting causes CP to delete the file. The HOLD setting for the 3800 printer in Figure 13 on page 263 indicates that CP will put a file in system hold. CP does not display a load check action for an impact printer.

You change the load check action setting for a 3800 with the START command.

Fold or Nofold Status (Impact Printers Only)

FOLD indicates that lowercase characters are being translated into uppercase for printing and NOFOLD indicates they are not.

Spool File Destination Value

The *dest field* (which appears after the keyword DEST) indicates those destinations the printer will print. The *dest* variable has a 1 - 8 character value and there can be up to four destination values. In Figure 13 on page 263, both printers will print those spool files with a destination of BLDG1.

The first printer's DEST value is always shown as OFF unless it has been specifically set by the START command. If the remaining DEST values have not been set (by the START command), they are shown as - NONE -.

Advanced Function Printer Status

Although a CP printer can select spool files having AFP characteristics, it is best to enter the START command with the NOAFP option. This avoids any erroneous output.

Active File Printing Response Line

If the printer is not in a drained state, there may be a file that is either actually printing or ready to print. This is the *active file*.

If there is an active file on the printer, CP displays information about it after the other printer status, which will look something like the PRINTING response lines at the bottom of Figure 13 on page 263. In addition to the device type and real device number, the PRINTING response line shows, for the active file:

- The *file owner*—the user ID of the user who owns or created the file
- The *spool file identification number*—a number CP assigns to the file when it is created
- The *file record count*—the number of logical records in the spool file, expressed in units, thousands (K), or millions (M)
- The *file copy number*—the number of the copy now printing, which is useful for files that have more than one copy
- The *file sequence number*—a number that appears on the separator pages (if any), marking the start and end of the printed file. CP maintains a separate list of sequence numbers for each spooled printer.

In Figure 13 on page 263, the active file on the 3800 is file 1603. This file belongs to user MAINT and contains 4436 logical records. Copy 001 is about to print, and sequence number 001 will appear on the file's separator pages.

The active file on the impact printer in Figure 13 on page 263 is file 1610. It also belongs to user MAINT, but it contains 5328 logical records. The printer is about to print copy 003. The separator pages will also show sequence number 001.

Understanding Print File Status

To display information about all of the spool files in the print queue, enter one or all of the following commands:

```
query printer
query printer exp
query printer all
query printer psf
query printer seclabel
```

In response, CP displays information about all the files that are waiting to be printed. To display information about only a particular spool file in the print queue, enter:

```
query printer userid nnnn
```

where *nnnn* is the spool file identification number for the file.

Figure 14 on page 269 is an example of a CP response to these commands. The sections that follow Figure 14 on page 269 describe the responses in detail. However, because the information CP displays for one file is the same as the information CP displays for all files, the discussion covers only the responses to the QUERY PRINTER, QUERY PRINTER EXP, QUERY PRINTER ALL, QUERY PRINTER PSF, and QUERY PRINTER SECLABEL commands, rather than the information about each file.

```
Sample Response to the QUERY PRINTER Command:
OWNERID  FILE CLASS RECORDS  CPY HOLD USERFORM OPERFORM DEST     KEEP MSG
OPERATOR 1598 T CON 00006789 001 NONE LISTING  STANDARD OFF      OFF  OFF
OPERATOR 1602 T CON 00000171 001 USER LISTING  STANDARD OFF      OFF  OFF
MAINT    1603 B PRT 00004436 001 NONE LISTING  STANDARD ANY      ON   OFF
MAINT    1608 B PRT 00045373 003 NONE LISTING  STANDARD ANY      ON   OFF
FINANCE1 1612 S PRT 00004833 001 NONE BILLING1 BILLFORM BLDG1    ON   OFF
FINANCE1 1615 R PRT 00000524*020 NONE MEMO     TTABT6   BLDG1    OFF  OFF
MAINT    1629 B PRT 00047682 001 NONE LISTING  STANDARD ANY      ON   OFF

Sample Response to the QUERY PRINTER EXP Command:
OWNERID  FILE CLASS RECORDS  FLASH FCB  MDFY   FLSHC LOAD CHARS               SIZE
OPERATOR 1598 T CON 00006789                         NO                       0004
OPERATOR 1602 T CON 00000171                         NO                       0004
MAINT    1603 B PRT 00004436                         BEG                      0004
MAINT    1608 B PRT 00045373                         ANY                      0004
FINANCE1 1612 S PRT 00004833 BILL  BFCB BMOD 1 001   ANY  BT10 BT10 BB10 BB10 0004
FINANCE1 1615 R PRT 00000524            CONF 1       BEG                      0004
MAINT    1629 B PRT 00047682                         ANY                      0004

Sample Response to the QUERY PRINTER ALL Command:
OWNERID  FILE CLASS RECORDS  CPY HOLD DATE  TIME     NAME     TYPE     DIST
OPERATOR 1598 T CON 00006789 001 NONE OPEN- 0009                       OPER
OPERATOR 1602 T CON 00000171 001 USER 08/03 09:08:55                   OPER
MAINT    1603 B PRT 00004436 001 NONE 08/03 10:59:34 TESTIM1  LISTING  1:16-3
MAINT    1608 B PRT 00045373 003 NONE 08/03 11:02:27 TESTIM2  LISTING  1:16-3
FINANCE1 1612 S PRT 00004833 001 NONE 08/03 11:03:43 AUGBILL  REPORT   2:12-1
FINANCE1 1615 R PRT 00000524*020 NONE 08/03 11:10:28 AUGRPT   MEMO     2:12-1
MAINT    1629 B PRT 00047682 001 NONE 08/03 11:22:59 TESTIM3  LISTING  1:16-3

Sample Response to the QUERY PRINTER PSF Command:
OWNERID  FILE CLASS RECORDS  USERFORM OPERFORM DEST     CONVERSION SPECIAL PURGE
OPERATOR 1598 T CON 00006789 LISTING  STANDARD OFF      NOTCONV    NO      NO
OPERATOR 1602 T CON 00000171 LISTING  STANDARD OFF      NOTCONV    NO      NO
MAINT    1603 B PRT 00004436 LISTING  STANDARD ANY      COMPLETE   YES     NO
MAINT    1608 B PRT 00045373 LISTING  STANDARD ANY      COMPLETE   YES     YES
FINANCE1 1612 S PRT 00004833 BILLING1 BILLFORM BLDG1    NOTCONV    NO      NO
FINANCE1 1615 R PRT 00000524 MEMO     TTABT6   BLDG1    NOTCONV    NO      NO
MAINT    1629 B PRT 00047682 LISTING  STANDARD ANY      ACTIVE     YES     NO

Sample Response to the QUERY PRINTER SECLABEL Command for a Class D User:
OWNERID FILE CLASS RECORDS  CPY HOLD USERFORM OPERFORM DEST  SECLABEL
MAINT   1593 A PRT 00000173 001 NONE PLAIN    PLAIN    BLDG2 009BDF
MAINT   1603 A PRT 00000084 001 NONE PLAIN    PLAIN    BLDG1 12345678
MAINT   1632 A PRT 00000032 001 USER PLAIN    PLAIN    BLDG2 543219E
MAINT   1645 A PRT 00000576 001 SYS  PLAIN    PLAIN    BLDG1 89FED7
MAINT   1668 A PRT 00000893 001 USYS PLAIN    PLAIN    BLDG3 7EF9C1

Sample Response to the QUERY PRINTER SECLABEL Command for a Class G User:
ORIGINID FILE CLASS RECORDS  CPY HOLD FORM  DEST    KEEP MSG SECLABEL
MAINT    1593 A PRT 00000273 001 NONE PLAIN BLDG3   ON   ON  019BCF
MAINT    1603 A PRT 00000096 001 NONE PLAIN BLDG2   OFF  OFF 87654321
MAINT    1632 A PRT 00000032 001 USER PLAIN BLDG1   ON   OFF 54319E
MAINT    1645 A PRT 00000276 001 SYS  PLAIN BLDG1   ON   ON  82FED9
```

Figure 14. Sample QUERY Responses for Files in the Print Queue

QUERY PRINTER Response

In the response to the QUERY PRINTER command, the information is as follows:

- **ORIGINID|OWNERID** is the user ID of the file owner. The file owner is the user who either created the file or put it in the print queue.
- **FILE** is the spool file identification number, a number CP assigned to the file when it was created.
- **CLASS** tells you the class of the spool file and how the spool file was created. The class can be a letter (A through Z) or a number (0 through 9). The origin may be one of the following:
  - **PRT** (virtual printer)
  - **RDR** (real reader)
  - **PUN** (virtual punch)
  - **CON** (virtual console)
- **RECORDS** is the number of logical records in the file.

- **CPY** is the number of copies CP will print.

  If there is no asterisk in front of the number, CP prints the entire file the specified number of times. But if an asterisk precedes the number, and you print the file on a 3800, CP prints each page of the file the specified number of times. On a 3800, printing many copies of each page is faster than printing many copies of the whole file.
- **HOLD** is the status of the file, where:
  - **USER** means the file is in user hold.
  - **SYS** means the file is in system hold.
  - **USYS** means the file is in both user and system hold.
  - **NONE** means the file is not in hold status.

  If the file is in system hold or user hold status, you cannot print it.
- **FORM** is the name of the paper type as understood by the user.
- **USERFORM** is the name of the form associated with the file as understood by the user.
- **OPERFORM** is the name of the form associated with the file as understood by the operator. The form names under *OPERFORM* are the ones you use when you enter a START command for a printer.
- **DEST** is the destination value of the file, where:
  - *dddddddd* means that the file will be printed by a specific printer or group of printers indicated by the 1 - 8 character destination value.
  - **OFF** means that the file will be processed only by a printer that is specifically started or defaulted to OFF.
  - **ANY** means that the file can be selected on any CP-owned printer, regardless of the printer's DEST setting, as long as the other selection criteria match.
- **KEEP** is the keep status of the file, where:
  - **ON** means the file is in keep status.
  - **OFF** means the file is not in keep status.

  If the file is in keep status, you may print the file, but the system also keeps a copy in user hold.
- **MSG** is the message status of the file, where:
  - **ON** means the file is in message status.
  - **OFF** means the file is not in message status.

  If the file is in message status, CP sends a message to the file's owner when you print the file.

QUERY PRINTER EXP Response

In the response to the QUERY PRINTER EXP command, the information is as follows:

- **OWNERID** is the user ID of the file owner.
- **FILE** is the spool file identification number.
- **CLASS** tells you the class of the spool file and how the spool file was created.
- **RECORDS** is the number of logical records in the file.
- **FLASH** is the name of the forms overlay to load in a 3800 when you print the file.

  When the forms overlay feature is active, the 3800 flashes an image onto selected pages of a print file through a film negative mounted in a frame. The number under FLSHC gives the number of copies the 3800 prints while the forms overlay is in place.
- **FCB** is the name of the FCB module to be used to control the vertical spacing on a 3800 printer. The module must be present in the image library for which the 3800 is set up.
- **MDFY** is the name of the copy modification module a 3800 should use when it prints the file.

The copy modification feature allows a user to specify that the 3800 is to add or delete text from selected copies of an output file. The number after the name gives the copy modification character arrangement table number, which helps the 3800 locate the correct data. The module must be present in the image library for which the 3800 is set up.

- **FLSHC** is the number of copies the 3800 prints while a forms overlay is in place.
- **LOAD** is the position of 3800 load CCWs in the file, if any.

  A **3800 load CCW** is a special instruction imbedded in a spool file. It tells a 3800 to load data that defines the spacing, characters, or copy modification data needed to print the file. CP keeps track of whether a spool file has any 3800 load CCWs, and if so, whether they appear only at the beginning or throughout the file. Accordingly, the keyword under *LOAD* may be:

  – **NO** (no 3800 load CCWs in the file)

  – **BEG** (3800 load CCWs at the beginning of the file only)

  – **ANY** (3800 load CCWs throughout the file).
- **CHARS** is the name of the character set a 3800 should use to print the file. A file may request up to four character sets, which must be present in the 3800 image library for which the printer is set up.
- **SIZE** is the number of 4 KB DASD blocks allocated for data for the file.

QUERY PRINTER ALL Response

In the response to the QUERY PRINTER ALL command, the information is as follows:

- **OWNERID** is the user ID of the file owner.
- **FILE** is the spool file identification number.
- **CLASS** tells you the class of the spool file and how the spool file was created.
- **RECORDS** is the number of logical records in the file.
- **CPY** is the number of copies CP will print.
- **HOLD** is the hold status of the file.
- **DATE** and **TIME** are one of the following:

  – The date and time the file was created

  – **PRT- (recs)** if a file is being printed, where *recs* indicates the number of logical records left to print

  – **OPEN- vdev** if a file is still open, where *vdev* is the virtual device number of the virtual device at which the file is queued

  – **OPEN- SYS** if a file is open but not associated with a virtual device (for example, if the file is being processed by the SPXTAPE command).

- **NAME** is the file name of the file, if any.
- **TYPE** is the file type of the file, if any.
- **DIST** is a code your installation may use to distribute output.

QUERY PRINTER PSF Response

In the response to the QUERY PRINTER PSF command, the information is as follows:

- **OWNERID** is the user ID of the file owner.
- **FILE** is the spool file identification number.
- **CLASS** tells you the class of the spool file and how the spool file was created.
- **RECORDS** is the number of logical records in the file.
- **USERFORM** is the name of the form associated with the file as understood by the user.
- **OPERFORM** is the name of the form associated with the file as understood by the operator.

- **DEST** is the 1 - 8 character destination value of the file.
- **CONVERSION** is the conversion status of the file, where:
  - **COMPLETE** means that the print server has read the spool file, has marked it as being converted and has created a corresponding CMS file containing the actual data and print controls used to print the file.
  - **ACTIVE** means that the file is currently being converted to a corresponding CMS file.
  - **NOTCONV** means that the file has not yet been converted to a corresponding CMS file.
- **SPECIAL** indicates whether the file contains CCWs for advanced function printer data streams, where:
  - **YES** means that the file does contain CCWs.
  - **NO** means that the file does not contain CCWs.
- **PURGE** indicates whether the file has been purged from CP and is waiting for the print server to recognize this status, where:
  - **YES** means that the file has been purged.
  - **NO** means that the file has not been purged.

QUERY PRINTER SECLABEL Response

The following information is returned in response to the QUERY PRINTER SECLABEL command:

- **OWNERID** is the user ID of the file owner. The file owner is the user who either created the file or put it in the print queue.
- **FILE** is the spool file identification number, a number CP assigned to the file when it was created.
- **CLASS** tells you the class of the spool file and how the spool file was created. The class can be a letter (A through Z) or a number (0 through 9). The origin may be one of the following:
  - **PRT** (virtual printer)
  - **RDR** (real reader)
  - **PUN** (virtual punch)
  - **CON** (virtual console).
- **RECORDS** is the number of logical records in the file.
- **CPY** is the number of copies CP will print.

  If there is no asterisk in front of the number, CP prints the entire file the specified number of times. But if an asterisk precedes the number, and you print the file on a 3800, CP prints each page of the file the specified number of times. On a 3800, printing many copies of each page is faster than printing many copies of the whole file.
- **HOLD** is the status of the file, where:
  - **USER** means the file is in user hold.
  - **SYS** means the file is in system hold.
  - **USYS** means the file is in both user and system hold.
  - **NONE** means the file is not in hold status.

  If the file is in system hold or user hold status, you cannot print it.
- **USERFORM** is the name of the form associated with the file as understood by the user.
- **OPERFORM** is the name of the form associated with the file as understood by the operator. The form names under *OPERFORM* are the ones you use when you enter a START command for a printer.
- **DEST** is the destination value of the file, where:
  - *dddddddd* means that the file will be printed by a specific printer or group of printers indicated by the 1 - 8 character destination value.

  - **OFF** means that the file will be processed only by a printer that is specifically started or defaulted to OFF.
  - **ANY** means that the file can be selected on any CP-owned printer, regardless of the printer's DEST setting, as long as the other selection criteria match.
- **SECLABEL** specifies the security label of the print file. If a security label is not assigned to the file, the SECLABEL field contains the word NONE.

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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 grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

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.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information may contain sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., in the United States and/or other countries. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on IBM Copyright and trademark information (https://www.ibm.com/legal/copytrade).

The registered trademark Linux is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Terms and Conditions for Product Documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal Use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial Use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see:

- The section entitled **IBM Websites** at IBM Privacy Statement (https://www.ibm.com/privacy)
- Cookies and Similar Technologies (https://www.ibm.com/privacy#Cookies_and_Similar_Technologies)

Bibliography

This topic lists the publications in the z/VM library. For abstracts of the z/VM publications, see *z/VM: General Information*.

Where to Get z/VM Information

The current z/VM product documentation is available in IBM Documentation - z/VM (https://www.ibm.com/docs/en/zvm).

z/VM Base Library

Overview

- *z/VM: License Information*, GI13-4377
- *z/VM: General Information*, GC24-6286

Installation, Migration, and Service

- *z/VM: Installation Guide*, GC24-6292
- *z/VM: Migration Guide*, GC24-6294
- *z/VM: Service Guide*, GC24-6325
- *z/VM: VMSES/E Introduction and Reference*, GC24-6336

Planning and Administration

- *z/VM: CMS File Pool Planning, Administration, and Operation*, SC24-6261
- *z/VM: CMS Planning and Administration*, SC24-6264
- *z/VM: Connectivity*, SC24-6267
- *z/VM: CP Planning and Administration*, SC24-6271
- *z/VM: Getting Started with Linux on IBM Z*, SC24-6287
- *z/VM: Group Control System*, SC24-6289
- *z/VM: I/O Configuration*, SC24-6291
- *z/VM: Running Guest Operating Systems*, SC24-6321
- *z/VM: Saved Segments Planning and Administration*, SC24-6322
- *z/VM: Secure Configuration Guide*, SC24-6323

Customization and Tuning

- *z/VM: CP Exit Customization*, SC24-6269
- *z/VM: Performance*, SC24-6301

Operation and Use

- *z/VM: CMS Commands and Utilities Reference*, SC24-6260
- *z/VM: CMS Primer*, SC24-6265
- *z/VM: CMS User's Guide*, SC24-6266
- *z/VM: CP Commands and Utilities Reference*, SC24-6268

- *z/VM: System Operation*, SC24-6326
- *z/VM: Virtual Machine Operation*, SC24-6334
- *z/VM: XEDIT Commands and Macros Reference*, SC24-6337
- *z/VM: XEDIT User's Guide*, SC24-6338

Application Programming

- *z/VM: CMS Application Development Guide*, SC24-6256
- *z/VM: CMS Application Development Guide for Assembler*, SC24-6257
- *z/VM: CMS Application Multitasking*, SC24-6258
- *z/VM: CMS Callable Services Reference*, SC24-6259
- *z/VM: CMS Macros and Functions Reference*, SC24-6262
- *z/VM: CMS Pipelines User's Guide and Reference*, SC24-6252
- *z/VM: CP Programming Services*, SC24-6272
- *z/VM: CPI Communications User's Guide*, SC24-6273
- *z/VM: ESA/XC Principles of Operation*, SC24-6285
- *z/VM: Language Environment User's Guide*, SC24-6293
- *z/VM: OpenExtensions Advanced Application Programming Tools*, SC24-6295
- *z/VM: OpenExtensions Callable Services Reference*, SC24-6296
- *z/VM: OpenExtensions Commands Reference*, SC24-6297
- *z/VM: OpenExtensions POSIX Conformance Document*, GC24-6298
- *z/VM: OpenExtensions User's Guide*, SC24-6299
- *z/VM: Program Management Binder for CMS*, SC24-6304
- *z/VM: Reusable Server Kernel Programmer's Guide and Reference*, SC24-6313
- *z/VM: REXX/VM Reference*, SC24-6314
- *z/VM: REXX/VM User's Guide*, SC24-6315
- *z/VM: Systems Management Application Programming*, SC24-6327
- *z/VM: z/Architecture Extended Configuration (z/XC) Principles of Operation*, SC27-4940

Diagnosis

- *z/VM: CMS and REXX/VM Messages and Codes*, GC24-6255
- *z/VM: CP Messages and Codes*, GC24-6270
- *z/VM: Diagnosis Guide*, GC24-6280
- *z/VM: Dump Viewing Facility*, GC24-6284
- *z/VM: Other Components Messages and Codes*, GC24-6300
- *z/VM: VM Dump Tool*, GC24-6335

z/VM Facilities and Features

Data Facility Storage Management Subsystem for z/VM

- *z/VM: DFSMS/VM Customization*, SC24-6274
- *z/VM: DFSMS/VM Diagnosis Guide*, GC24-6275
- *z/VM: DFSMS/VM Messages and Codes*, GC24-6276
- *z/VM: DFSMS/VM Planning Guide*, SC24-6277

- *z/VM: DFSMS/VM Removable Media Services*, SC24-6278
- *z/VM: DFSMS/VM Storage Administration*, SC24-6279

Directory Maintenance Facility for z/VM

- *z/VM: Directory Maintenance Facility Commands Reference*, SC24-6281
- *z/VM: Directory Maintenance Facility Messages*, GC24-6282
- *z/VM: Directory Maintenance Facility Tailoring and Administration Guide*, SC24-6283

Open Systems Adapter

- Open Systems Adapter-Express Customer's Guide and Reference (https://www.ibm.com/support/pages/node/6019492), SA22-7935
- Open Systems Adapter-Express Integrated Console Controller User's Guide (https://www.ibm.com/support/pages/node/6019810), SC27-9003
- Open Systems Adapter-Express Integrated Console Controller 3215 Support (https://www.ibm.com/docs/en/SSLTBW_2.1.0/com.ibm.zos.v2r1.ioa/ioa.htm), SA23-2247
- Open Systems Adapter/Support Facility on the Hardware Management Console (https://www.ibm.com/docs/en/SSLTBW_2.1.0/com.ibm.zos.v2r1.ioa/ioa.htm), SC14-7580

Performance Toolkit for z/VM

- *z/VM: Performance Toolkit Guide*, SC24-6302
- *z/VM: Performance Toolkit Reference*, SC24-6303

RACF Security Server for z/VM

- *z/VM: RACF Security Server Auditor's Guide*, SC24-6305
- *z/VM: RACF Security Server Command Language Reference*, SC24-6306
- *z/VM: RACF Security Server Diagnosis Guide*, GC24-6307
- *z/VM: RACF Security Server General User's Guide*, SC24-6308
- *z/VM: RACF Security Server Macros and Interfaces*, SC24-6309
- *z/VM: RACF Security Server Messages and Codes*, GC24-6310
- *z/VM: RACF Security Server Security Administrator's Guide*, SC24-6311
- *z/VM: RACF Security Server System Programmer's Guide*, SC24-6312
- *z/VM: Security Server RACROUTE Macro Reference*, SC24-6324

Remote Spooling Communications Subsystem Networking for z/VM

- *z/VM: RSCS Networking Diagnosis*, GC24-6316
- *z/VM: RSCS Networking Exit Customization*, SC24-6317
- *z/VM: RSCS Networking Messages and Codes*, GC24-6318
- *z/VM: RSCS Networking Operation and Use*, SC24-6319
- *z/VM: RSCS Networking Planning and Configuration*, SC24-6320

TCP/IP for z/VM

- *z/VM: TCP/IP Diagnosis Guide*, GC24-6328
- *z/VM: TCP/IP LDAP Administration Guide*, SC24-6329
- *z/VM: TCP/IP Messages and Codes*, GC24-6330

- *z/VM: TCP/IP Planning and Customization*, SC24-6331
- *z/VM: TCP/IP Programmer's Reference*, SC24-6332
- *z/VM: TCP/IP User's Guide*, SC24-6333

Prerequisite Products

Device Support Facilities

- Device Support Facilities (ICKDSF): User's Guide and Reference (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5gc350033/$file/ickug00_v2r5.pdf), GC35-0033

Environmental Record Editing and Printing Program

- Environmental Record Editing and Printing Program (EREP): Reference (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5gc350152/$file/ifc2000_v2r5.pdf), GC35-0152
- Environmental Record Editing and Printing Program (EREP): User's Guide (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5gc350151/$file/ifc1000_v2r5.pdf), GC35-0151

Related Products

z/OS

- *Common Programming Interface Communications Reference (https://publibfp.dhe.ibm.com/epubs/pdf/c2643999.pdf)*, SC26-4399
- z/OS and z/VM: Hardware Configuration Definition Messages (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sc342668/$file/cbdm100_v2r5.pdf), SC34-2668
- z/OS and z/VM: Hardware Configuration Manager User's Guide (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sc342670/$file/eequ100_v2r5.pdf), SC34-2670
- z/OS: Network Job Entry (NJE) Formats and Protocols (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa320988/$file/hasa600_v2r5.pdf), SA32-0988
- z/OS: IBM Tivoli Directory Server Plug-in Reference for z/OS (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa760169/$file/glpa300_v2r5.pdf), SA76-0169
- z/OS: Language Environment Concepts Guide (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380687/$file/ceea800_v2r5.pdf), SA38-0687
- z/OS: Language Environment Debugging Guide (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5ga320908/$file/ceea100_v2r5.pdf), GA32-0908
- z/OS: Language Environment Programming Guide (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380682/$file/ceea200_v2r5.pdf), SA38-0682
- z/OS: Language Environment Programming Reference (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380683/$file/ceea300_v2r5.pdf), SA38-0683
- z/OS: Language Environment Runtime Messages (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380686/$file/ceea900_v2r5.pdf), SA38-0686
- z/OS: Language Environment Writing Interlanguage Communication Applications (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa380684/$file/ceea400_v2r5.pdf), SA38-0684
- z/OS: MVS Program Management Advanced Facilities (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa231392/$file/ieab200_v2r5.pdf), SA23-1392
- z/OS: MVS Program Management User's Guide and Reference (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosv2r5sa231393/$file/ieab100_v2r5.pdf), SA23-1393

XL C++ for z/VM

- XL C/C++ for z/VM: Runtime Library Reference, SC09-7624
- XL C/C++ for z/VM: User's Guide, SC09-7625

Index

Product Number: 5741-A09

Printed in USA

SC24-6326-73