

z/OS



MVS Recovery and Reconfiguration Guide

z/OS



MVS Recovery and Reconfiguration Guide

Note

Before using this information and the product it supports, be sure to read the general information under "Notices" on page C-1.

Third Edition, October 2003

This is a major revision of SA22-7623-01.

This edition applies to Version 1 Release 4 of z/OS (5694-A01), to Version 1 Release 4 of z/OS.e (5655-G52), and to all subsequent releases and modifications until otherwise indicated in new editions.

IBM welcomes your comments. A form for readers' comments may be provided at the back of this document, or you may address your comments to the following address:

International Business Machines Corporation
Department 55JA, Mail Station P384
2455 South Road
Poughkeepsie, NY 12601-5400
United States of America

FAX (United States & Canada): 1+845+432-9405

FAX (Other Countries):

Your International Access Code +1+845+432-9405

IBMLink™ (United States customers only): IBMUSM10(MHVRCFS)

Internet e-mail: mhvrfs@us.ibm.com

World Wide Web: <http://www.ibm.com/servers/eserver/zseries/zos/webqs.html>

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

Make sure to include the following in your comment or note:

- Title and order number of this document
- Page number or topic related to your comment

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

© Copyright International Business Machines Corporation 1988, 2003. All rights reserved.

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

Contents

Figures	ix
Tables	xi
About this document	xiii
Who should use this document	xiii
Where to find more information	xiii
Information updates on the web	xiii
Accessing z/OS licensed documents on the Internet	xiii
Using LookAt to look up message explanations	xiv
Summary of Changes	xv

Part 1. Recovery

Chapter 1. Introduction to Recovery	1-1
Approach to Recovery Procedures	1-1
Problem Indications	1-2
Problem Source Identification	1-2
Recovery	1-2
Recovery Actions in IOSP Concurrent Maintenance Mode	1-4
Chapter 2. Recovery from Waits or Hangs	2-1
Disabled Wait with Wait State Code	2-1
Restartable Waits	2-1
Wait in a System in a Sysplex	2-1
Recovery Actions for a Restartable Disabled Wait	2-2
Recovery Actions for a Non-restartable Disabled Wait	2-2
Enabled Wait	2-2
Causes	2-2
Recovery Actions for an Enabled Wait	2-3
Hang of a Subsystem, Address Space, or Job	2-3
Recovery Actions for a Hung Subsystem	2-3
Recovery Actions for a Hung Address Space or Job	2-4
Chapter 3. Recovery from Loops	3-1
Loop in a System in a Sysplex	3-1
Enabled Loops	3-1
Stopping Enabled Loops	3-1
Disabled Loops	3-3
Identifying Disabled Loops	3-3
Address Instruction Trace	3-3
Operator Actions for a Disabled Loop	3-3
Spin Loops	3-4
Recovery Actions for Spin Loops	3-5
Chapter 4. Recovery from JES2 Problems	4-1
Problems during JES2 Initialization	4-1
Recovery Actions for a JES2 Initialization Problem	4-1
Errors in Spool Data Set or Checkpoint I/O	4-1
Recovery Actions for a JES2 Spool or Checkpoint I/O Problem	4-2
Errors in Remote Job Entry (RJE)	4-2
Recovery Actions for an RJE Problem	4-3

Errors in Network Job Entry (NJE)	4-3
Recovery Actions for an NJE Problem	4-3
Chapter 5. Recovery from JES3 Problems	5-1
Restarting JES3 After a Failure	5-1
Restarting the Global Processor	5-1
Restarting a Local Processor	5-2
Recovering JES3 Checkpoint Data Sets	5-2
Recovering from Permanent Errors on a Checkpoint Data Set	5-2
Recovering when a Checkpoint Data Set is Out of Space	5-3
Using Dynamic System Interchange in Recovery	5-3
Disabling the Old Global Processor	5-3
Starting a Local Processor as a Global Processor.	5-4
Writing Dynamic System Interchange Procedures	5-4
Recovering from CTC Failures	5-5
Operator Recovery Actions when Alternate CTC is Defined	5-6
Operator Recovery Actions when Only Primary CTC is Defined.	5-6
Recovering from BSC RJP Problems	5-6
Recovering from Output Writer FSS Failures.	5-7
Recovering a 3480 Tape for a Stand-Alone Dump.	5-7
Operator Recovery Actions for a 3480	5-8
Chapter 6. Recovery from Sysplex Problems	6-1
Severe Errors	6-1
Initialization Problems	6-1
Duplicate System in the Sysplex	6-1
Taking Over Sysplex Initialization	6-2
Inactive Systems in the Sysplex	6-2
No Primary Sysplex Couple Data Set	6-3
Problems With Signalling Path Definitions.	6-3
Incorrect Sysplex Timer	6-4
Spin Loop During Nucleus Initialization.	6-4
Too Many Systems in the Sysplex	6-5
Problems with Coupling Facility Initialization	6-5
Loss of a System in the Sysplex	6-6
Automatic Recovery.	6-7
Recovery Actions for Loss of a System in the Sysplex	6-8
Signalling Path Problems.	6-8
Loss of Some Signalling Paths Between Systems.	6-9
Loss of All Signalling Paths Between Some Systems	6-9
Loss of All Signalling Paths Between All Systems	6-10
Sysplex Timer Problems.	6-10
Sysplex Timer Problem in Sysplex of One System	6-10
Sysplex Timer Problem in Sysplex of Two or More Systems	6-10
Couple Data Set Problems	6-11
Loss of the Primary When an Alternate Exists	6-11
Loss of the Primary When No Alternate Exists.	6-11
DASD Failures	6-12
Coupling Facility Problems.	6-12
Failure of a Coupling Facility	6-12
Failure of a Structure in a Coupling Facility.	6-12
Connectivity Failure to a Coupling Facility or a Structure.	6-12
Problems Causing an Abend or Wait State Code.	6-13
Chapter 7. Recovery for Global Resource Serialization	7-1
Recovery when Some Ring Complex Systems Are in Sysplex	7-2

Recovery when Sysplex and Ring Complex Contain Same Systems	7-2
Global Resource Serialization Ring Disruption	7-3
Needed Recovery Actions	7-3
Example Complex	7-4
Purge a System from the Ring Complex Before RelIPL	7-4
Actions to Purge a System before relIPL	7-5
Ring Disruption in Progress during IPL	7-5
Recovery Actions for a Ring Disruption during IPL	7-5
Primary Link Failure	7-5
Recovery Actions for a Primary Link Failure	7-7
Alternate Link Failure	7-7
Recovery Actions for an Alternate Link Failure	7-8
System Failure	7-8
Rebuilding the Ring	7-8
Bringing the Failed System Back into the Complex	7-8
Recovery Planning for Different Size Complexes	7-9
Automatic Restart Example	7-9
Manual Restart	7-11
Restart of Two-System Complex	7-13
Reactivating a Quiesced System	7-15
Recovery Actions for Reactivating Quiesced Systems	7-16
Chapter 8. Recovery for Consoles	8-1
System Problems	8-1
Console Hardware Errors	8-1
Other Procedures	8-1
System Programming Problems	8-2
Recovery Actions for an Error Message in the Message Area	8-2
Recovery Actions for a Lack of Console Activity	8-2
Console Message Backups	8-2
Recovery Actions for a WTO Buffer Shortage	8-4
Recovery Actions for a WTOR Buffer Shortage	8-5
Recovery Actions for Command Flooding Indications	8-5
Console Hardware Errors	8-6
Recovery Actions for Error Messages Centered on the Screen	8-6
Recovery Actions for a Blank Screen	8-6
Recovery Actions for a Locked Keyboard	8-6
Recovery Actions for Failure of the Master Console	8-7
Recovery Actions for a No-Consoles Condition	8-7
Recovery Actions if a No-Consoles Recovery Fails	8-8
Recovery Actions for Lost Messages	8-9
Actions to See System Messages at the System Console	8-9
Chapter 9. Recovery from Hardware Problems	9-1
Hardware Problems	9-1
Hardware to Operating System Recovery Actions	9-1
Information Provided with Machine Checks	9-1
CPU Errors	9-2
Soft CPU Errors	9-2
Hard CPU Errors	9-3
Terminating Errors on CPUs	9-4
Vector Facility Errors	9-4
Alternate CPU Recovery (ACR)	9-5
Service Processor Damage	9-6
Permanent Failure	9-6
Temporary Failure	9-6

Storage Errors	9-6
Soft Storage Errors	9-7
Hard Storage Errors	9-7
Effects of Storage Errors	9-8
High Speed Buffer (Cache)	9-8
Channel Subsystem Errors	9-8
Channel Report Words (CRWs)	9-8
Channel Path Recovery	9-8
Channel Path Alert Conditions	9-11
Subchannel Recovery	9-11
Monitoring Facility Recovery	9-12
I/O Device Errors	9-12
I/O Errors	9-12
Missing Interrupts	9-12
Hot I/O	9-14
Recovery for Hung Devices	9-15
Recovery for Failing Devices	9-16
Shared Device Recovery	9-17
3880/3380 Considerations	9-19
DASD Maintenance and Recovery	9-20
Boxed Device - Operator Actions	9-21
Recovery for a Failing Alias Unit Control Block (UCB)	9-21
Boxed Tapes under Tape Management System Control	9-22
Tape Boxed Due to Lost Assign	9-22
Printer Boxed Due to Lost Assign	9-22
Additional Recovery Actions	9-24
Recovery by CPU Restart	9-24
Continuing a Vector Job If a Vector Facility is Offline	9-25
 Chapter 10. Recovery from APPC Problems	 10-1
Recovery for the APPC Address Space	10-1
APPC Initialization Problems	10-1
Abnormal Ending of APPC with Restart	10-1
Abnormal Ending of APPC without Restart	10-2
Recovery for the APPC/MVS Transaction Scheduler	10-2
Abnormal Ending of the APPC/MVS Transaction Scheduler with Restart	10-2
Abnormal Ending of the APPC/MVS Transaction Scheduler without Restart	10-2

Part 2. Reconfiguration

 Chapter 11. Introduction to Reconfiguration	 11-1
Reconfiguration	11-1
Partitions	11-1
RSU Parameter for Partitioning	11-2
 Chapter 12. Input/Output Configuration	 12-1
Channel Subsystem Configurations	12-1
Channel Designations on Non-partitionable 3090 Models	12-2
Channel Designations on Partitionable 3090 Models	12-2
Configuration Guidelines	12-2
Channel Subsystem Considerations for an ES/4381 Processor Model 92E	12-2
I/O Configuration for Non-Partitionable Processor Complex	12-3
I/O Configuration for Partitionable Processor Complex in Single-Image Mode	12-7
Configuration for Master Console	12-8
Processing Messages at the System Console	12-10
IOCP Considerations	12-10

MVS Hardware Configuration Definition Program Considerations	12-10
Chapter 13. Reconfiguration	13-1
Dynamic I/O Configuration	13-1
Logical and Physical Reconfiguration	13-2
Reconfiguration Support According to Processor Types	13-2
Uniprocessor (UP)	13-2
Multiprocessor (MP)	13-2
Reconfiguring a Central Processor	13-3
Actions to Reconfigure a Central Processor Offline	13-3
Reconfiguring a Central Processor with an ICRF.	13-3
Actions to Bring Online a Central Processor and its ICRF	13-4
Removing the Last ICRF	13-4
Actions to Take Offline a Central Processor and its ICRF.	13-4
Reconfiguring Vector Facilities	13-4
Actions to Reconfigure a Vector Facility	13-5
Removing the Last Vector Facility	13-5
Actions to Bring Online a Central Processor and its Vector Facility	13-6
Actions to Bring Online a Central Processor but Not its Vector Facility	13-6
Actions to Take Offline a Central Processor and its Vector Facility	13-7
Actions to Bring Online a Vector Facility for an Online Central Processor	13-7
Actions to Take Offline a Vector Facility When its Central Processor Remains Online	13-8
Reconfiguring Central and Expanded Storage	13-8
Physical View of Central Storage	13-8
Storage Increments and Sub-increment Sizes	13-9
Specifying the RSU Parameter	13-11
Reconfiguring Channel Paths	13-14
Actions to Reconfigure Channel Paths	13-15
Reconfiguring I/O Devices	13-15
Reconfiguring a Coupling Facility	13-15
Chapter 14. Dynamic Partitioning and Merging	14-1
Sequence for Partitioning and Merging	14-1
Reconfiguration of JES3 Systems	14-2
RSU Parameter	14-2
Central Storage in a Partitionable Processor Complex.	14-3
Characteristics of a Partitionable Processor	14-4
Configuration in Single-Image Mode	14-4
Example of Partitioning	14-6
DISPLAY M=SIDE Information	14-10
DISPLAY M Information	14-10
Example of Merging Sides	14-12
DISPLAY M=SIDE Information	14-14
DISPLAY M Information	14-15
Chapter 15. Reconfiguration under PR/SM	15-1
Uses for Logical Partitions	15-1
Uses in a Sysplex	15-1
Actions to Use Logical Partitions.	15-1
Defining Logical Partitions for PR/SM Dynamic Storage Reconfiguration	15-2
Logical Storage Elements	15-2
Defining Channel Paths for Reconfiguration	15-3
Defining the RSU parameter	15-3
Initializing a System into a Logical Partition.	15-4
Stand-Alone Dump IPL in Logical Partitions	15-4

Using the CONFIG Command to Configure Storage 15-4

 Configuring Storage Online to the Logical Partition 15-5

 Configuring Storage Offline to the Logical Partition 15-5

 When Only Some of the Reserved Central Storage is Available 15-5

 When Only Some of the Reserved Expanded Storage is Available 15-5

 Automating Dynamic Storage Reconfiguration. 15-5

 Dynamically Configuring Central Storage on an ES/9021 Processor. 15-5

 Examples of Dynamic Storage Reconfiguration 15-6

Using the CONFIG Command to Move Channel Paths 15-9

 Example of Moving a Channel Path from One Logical Partition to Another 15-10

Appendix A. Accessibility A-1

 Using assistive technologies A-1

 Keyboard navigation of the user interface. A-1

Appendix B. Creating Path Group Identifiers (PGIDs) B-1

Notices C-1

 Trademarks. C-2

Glossary D-1

Index X-1

I

Figures

9-1.	DASD Devices Shared Between Two Systems	9-10
9-2.	DASD Device Shared Between Three Systems.	9-19
9-3.	Boxed Device Recovery Procedure	9-23
12-1.	ES/3090 Processor Complex Channel Subsystem Configuration	12-1
12-2.	4381 Dual-Processor Channel Subsystem Configuration	12-3
12-3.	DASD Configuration for Maximum Availability with a 3090 Complex	12-4
12-4.	Tape Configuration for Maximum Availability (3090 Complex).	12-5
12-5.	Configuration of a 3725 Communication Controller for Maximum Availability (3090 Complex)	12-6
12-6.	Unit Record or Local Display Device Configuration for Maximum Availability (3090 Complex)	12-7
12-7.	DASD Configuration for Maximum Availability (3090 in Single-Image Mode)	12-8

Tables

7-1.	Conditions for Rebuilding the Lock Structure	7-1
9-1.	Correcting Boxed Conditions	9-23
12-1.	Channel Paths and Channel Elements on Partitionable 3090 Processor Complexes	12-2

About this document

This document gives system programmers information needed to create the following for system operators or for automation programs:

- Recovery actions to correct hardware and operating system problems so that the system can again do productive work. The document emphasizes maintaining system availability after an abnormal event. Recovery information appears in Part 1, "Recovery."
- Reconfiguration actions so that the system can be physically and logically changed. Reconfiguration information appears in Part 2, "Reconfiguration."

The document covers the MVS system and the hardware on which it runs.

The document does not contain ready-to-use procedures; rather, it contains information and general procedures that an installation can use to develop its own procedures. An installation can use the information and procedures:

- To write an operator's manual, which is often called a **run book**
- To develop automation programs to resolve problems
- To create a mixture, so that the operators resolve some problems and automation programs resolve others

Before developing recovery procedures, system programmers should read *z/OS MVS Diagnosis: Procedures* to plan the installation's problem management and to plan for recovery.

Who should use this document

This document is for system programmers who must plan recovery and reconfiguration procedures for the installation. These programmers can be experienced MVS users or experienced system programmers setting up their first MVS system.

Where to find more information

Where necessary, this document references information in other documents, using shortened versions of the document title. For complete titles and order numbers of the documents for all products that are part of z/OS™ and z/OS.e, see *z/OS Information Roadmap*.

Information updates on the web

For the latest information updates that have been provided in PTF cover letters and Documentation APARs for z/OS and z/OS.e, see the online document at:

<http://www.s390.ibm.com:80/bookmgr-cgi/bookmgr.cmd/BOOKS/ZIDOCMST/CCONTENTS>

This document is updated weekly and lists documentation changes before they are incorporated into z/OS publications.

Accessing z/OS licensed documents on the Internet

z/OS licensed documentation is available on the Internet in PDF format at the IBM Resource Link™ Web site at:

<http://www.ibm.com/servers/resourceLink>

Licensed documents are available only to customers with a z/OS license. Access to these documents requires an IBM Resource Link user ID and password, and a key code. With your z/OS order you received a Memo to Licensees, (GI10-0671), that includes this key code.¹

To obtain your IBM Resource Link user ID and password, log on to:

<http://www.ibm.com/servers/resourceLink>

To register for access to the z/OS licensed documents:

1. Sign in to Resource Link using your Resource Link user ID and password.
2. Select **User Profiles** located on the left-hand navigation bar.

Note: You cannot access the z/OS licensed documents unless you have registered for access to them and received an e-mail confirmation informing you that your request has been processed.

Printed licensed documents are not available from IBM.

To print licensed documents, you can use the PDF format on either **z/OS Licensed Product Library CD-ROM** or IBM Resource Link .

Using LookAt to look up message explanations

LookAt is an online facility that lets you look up explanations for most messages you encounter, as well as for some system abends and codes. Using LookAt to find information is faster than a conventional search because in most cases LookAt goes directly to the message explanation.

You can access LookAt from the Internet at:

<http://www.ibm.com/eserver/zseries/zos/bkserv/lookat/> or from anywhere in z/OS or z/OS.e where you can access a TSO/E command line (for example, TSO/E prompt, ISPF, z/OS UNIX System Services running OMVS).

The LookAt Web site also features a mobile edition of LookAt for devices such as Pocket PCs, Palm OS, or Linux-based handhelds. So, if you have a handheld device with wireless access and an Internet browser, you can now access LookAt message information from almost anywhere.

To use LookAt as a TSO/E command, you must have LookAt installed on your host system. You can obtain the LookAt code for TSO/E from a disk on your z/OS *Collection* (SK3T-4269) or from the LookAt Web site's **Download** link.

1. z/OS.e™ customers received a Memo to Licensees, (GI10-0684) that includes this key code.

Summary of Changes

Summary of Changes for SA22-7623-02 z/OS Version 1 Release 4 as updated October 2003

The book contains information previously presented in *z/OS MVS Recovery and Reconfiguration*, SA22-7623-01, which supports z/OS Version 1 Release 2.

New information:

- MVS Path group identifiers (PGIDs) now come in two formats, 0 and 1. Format 0 is for systems on pre-z990 processors, or systems on z990 processors at the z/OS z990 compatibility support level. Format 1 is for systems on the z990 processor at the z/OS exploitation support level. See Appendix B, “Creating Path Group Identifiers (PGIDs),” on page B-1. Any operating system that shares devices with MVS needs to use the same method as MVS to create path group identifiers (PGIDs).
- Information has been added to indicate this document supports z/OS.e.

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

Starting with z/OS V1R2, you may notice changes in the style and structure of some content in this document—for example, headings that use uppercase for the first letter of initial words only, and procedures that have a different look and format. The changes are ongoing improvements to the consistency and retrievability of information in our documents.

Summary of Changes for SA22-7623-01 z/OS Version 1 Release 2

The book contains information previously presented in *z/OS MVS Recovery and Reconfiguration*, SA22-7623-00, which supports z/OS Version 1 Release 1.

New Information

- A section has been added for command flooding prevention.

This document contains terminology, maintenance, and editorial changes, including changes to improve consistency and retrievability.

Summary of Changes for SA22-7623-00 z/OS Version 1 Release 1

The book contains information also presented in *OS/390 MVS Recovery and Reconfiguration*.

Changed information: Information is added to describe SNA MCS consoles.

This document contains terminology, maintenance, and editorial changes, including changes to improve consistency and retrievability.

Part 1. Recovery

Chapter 1. Introduction to Recovery

This part of the book covers recovery from hardware and software problems.

Recovery is the attempt by the hardware, the operating system, the operator, automation programs, or any combination of these, to correct software or hardware problems and allow the system to do productive work again.

The hardware itself recovers from many temporary hardware errors. These errors do not require operating system or operator intervention.

The hardware communicates most hardware failures to the operating system. In an attempt to recover, the system may retry a failed operation. The system may also determine that retry is not possible or may find that a retry failed; then the system may configure a failed hardware unit out of the system, or *offline*.

If recovery requires action from outside the system, the operating system issues one or more messages. To recover, the operator or an automation program responds to the messages. For example, the system may issue a message to request that the operator or automation program select a recovery option.

This book gives generalized recovery actions for the following types of problems:

- Waits and hangs, consisting of disabled waits, enabled waits, and hangs of a subsystem, an address space, or a job
- Loops, consisting of enabled loops or disabled loops, which include spin loops
- JES2 problems, including initialization problems and errors in a spool data set, checkpoint I/O, remote job entry (RJE), and network job entry (NJE)
- JES3 problems, including a JES3 failure, loss of a JES3 checkpoint data set, channel-to-channel (CTC) failure, and output writer failure
- Sysplex problems, including loss of a system and external time reference (ETR) problems
- Loss of connectivity to the global resource serialization lock structure on the coupling facility for a global resource serialization star complex
- Global resource serialization problems that cause ring disruption
- Console problems caused by the operating system or by hardware
- Hardware problems in central processors (CPU), the service processor, storage, the channel subsystem, and direct access storage devices (DASDs)
- APPC problems, including non-recoverable errors for the APPC address space and for the APPC/MVS Transaction Scheduler

Approach to Recovery Procedures

If you are the system programmer who oversees operations, you need to develop installation-specific recovery actions, written in an operator's manual or performed by automation programs. Base these actions on:

- The recovery actions given in this book. These actions are generalized; the system programmer needs to adapt them for the installation.
- The operator responses described in the abend codes, wait state codes, and messages.

Include in the recovery actions the following types of activities:

- Look for problem indications

- Identify the source of the problem
- Recover from the problem

Problem Indications

To find problems when they occur, be alert to the following:

- Audible alarms on the processor controller element (PCE)
- Messages on the system console
- Messages on the multiple console support (MCS) and SNA multiple console support (SMCS) and job entry subsystem (JES) consoles
- Wait states
- The behavior of the system, such as the system being hung with no other indicators of problems
- Problems indicated in Resource Measurement Facility (RMF) Monitor III reports

A problem may be permanent or intermittent

Problem Source Identification

The next step is to identify the cause of the problem. The problem source identifications are:

- Hardware and, if possible, the unit
- Software and, if possible, the component, product, or application
- Environment, such as the electrical distribution to the building

Recovery

Recovery Goals

The problem, its impact on the system users and work load, and the installation's recovery policies affect the goal of the recovery. The system programmer should design the recovery actions to produce the needed recovery. The recovery goals can include:

- **The fastest recovery possible:** Recovery actions can make the maximum number of components operational as soon as possible. The actions may completely disrupt some aspects of the operation, but would benefit the most users or the most important applications.
- **The maximum computing power:** Recovery actions can make resources available for a critical application that requires many resources.
- **The least disruptive recovery possible:** Recovery actions can keep from disrupting the overall operation. Individual applications or interactive users could be impacted.
- **Degraded operation:** Recovery actions can keep the system running for a time, even though performance would be slower. This approach may be the best solution if it meets the response times needed for priority applications or if users agree to a slower response during a problem period. Later, the system can be stopped for repair.

Use this approach if the failure occurs at a critical time when operations must not be stopped if the system is still up, or if hardware support and repair parts are not on site.

Using one side of a partitionable processor complex to carry the work load of the entire complex is degraded operation.

Note: A processor complex is the set of hardware units that support one or more operating systems.

Recovery Approaches

The recovery approaches depend on the model of the processor complex, the type of failure, and installation recovery policies. In the same installation, the approach chosen for a particular time of day, work load, or affected application may be different when an identical failure occurs for a different time, work load, or application.

Recovery actions by the hardware, the software, or both can perform recovery for some problems. These actions are often automatic. Usually, the recovery actions can be customized through parmlib members or through the operator replies to some messages.

For other failures, the operator or an automation program must perform actions to do one of the following:

- **Switch to a back-up system:** The operator can move the work load to another system. If available, the move can be performed using the extended recovery facility (XRF).
- **Continue running with a degraded system:** If major required elements are working, the processor complex can operate without the failing elements. The system can continue running until hardware support can repair the failure. This degraded operation occurs when:
 - Automatic recovery places a failing element offline.
 - An outage occurs, but the operator reIPLs the system without the failing element.

Degraded operation can also be obtained through dynamic partitioning. On partitionable models, the operator can partition the system if the installation prepared the system for dynamic partitioning, if the critical work load can run on half of the system, and if the time is convenient for half the processor complex to be stopped for repairs.

- **Stop the system and fix the problem:** If needed, the system can be stopped until it is repaired.

To achieve some operation on partitionable models, the operator can statically partition the system after the complex stops because of a hardware or software failure and one side has an error in a required major element and cannot continue to run. After partitioning, power-on reset (POR) the working side.

If a processor complex that cannot be partitioned fails in a major element, all operations may have to be stopped until the processor is repaired. One of the advantages of a partitionable processor is that hardware support can perform maintenance on one side while the other side continues to perform productive work.

Recovery Actions

The commonly used recovery actions follow.

- **Restart:** The operator restarts a CPU by the RESTART option on a system console frame or by the RESTART service language command. Restart ends the current unit of work that was being processed on the restarted CPU.
- **Configuration:** The operator configures hardware units offline and online both physically to the hardware and logically to the software using CONFIG operator commands. See Chapter 13, “Reconfiguration.”
- **Dynamically Partition the Processor Complex:** The operator should dynamically partition a partitionable complex if using a side will permit operation until the hardware can be repaired.

- **Interface Reset:** The operator enters an IFRST service language command on the system console to reset the interface on one or all of the channel paths on the system. Any outstanding reserves from this system are lost.
- **System Initialization:** The operator performs initial program load (IPL) to:
 - Reset all hardware units
 - Clear storage, if desired
 - Load the stand-alone dump program into storage and obtain a dump, if needed
 - Reload the operating system control program (SCP) into storage
 - Initialize the operating system
- **System Initial Microprogram Load:** The operator enters a SYSIML service language command on the system console to reset and initialize the processor, but not the processor controller. A SYSIML does the following:
 - Initializes all processors that are configured online
 - Initializes the channel subsystem
 - Validates and clears central storage
 - Initializes the hardware system area (HSA)
- **Statically Partition the Processor Complex:** The operator should statically partition a partitionable complex if using a side will permit operation until the hardware can be repaired.
- **Power on Reset:** The operator performs a power on reset (POR) to do all the SYSIML actions plus:
 - Power on the processor unit
 - Perform additional processor complex resets
 - Clear expanded storage
 - Reset the time-of-day (TOD) clocks
- **Power Off/On Processor Complex:** On rare occasions, the only appropriate recovery action will be for the operator to power off the processor complex, then to power it on. A power off/on sequence is the only way to reset the processor complex completely.

Restart or IPL the PCE is another recovery action. A warm start (IPL) of the PCE is usually done automatically by the system and rarely requires operator intervention. This action normally does not disrupt the processor complex because the only actions involve the PCE.

Reference Books

See the following:

- *Processor Recovery Guide* for these recovery actions and for the service language commands
- *Principles of Operation* for RESTART

Recovery Actions in IOSP Concurrent Maintenance Mode

If a service processor fails and is in I/O Support Processor (IOSP) concurrent maintenance mode, the system continues operating but cannot perform certain functions. Consequently, certain functions of the operating system will not work or will work incompletely.

A PCE entering IOSP concurrent maintenance mode may issue an operator message with the prefix ARRP to an MVS console.

In writing procedures for the operator's manual, do not ask the operator to do the following in IOSP concurrent maintenance mode:

- Enter a CONFIG STOR operator command.
- Restart the operating system from a restartable wait state.
- RelPL the operating system.
- Obtain a stand-alone dump.
- Use frames and displays on the system console.

The following functions work incompletely or not at all:

- A CONFIG CPU operator command can reconfigure the processor logically, but not physically. The command will not be able to write restart text.
- Alternate CPU recovery (ACR) cannot configure a processor physically offline.
- The system cannot obtain channel measurement data.
- The input/output configuration program (IOCP) cannot update the input/output configuration data set (IOCDS).
- The system will not be able to communicate with the operator on the system console.
- The PCE alarm will not be sounded to indicate the MVS no-consoles condition or as a precursor to MVS communicating to the operator on the system console.

Chapter 2. Recovery from Waits or Hangs

The operating system can enter one of the following:

- Disabled wait with a wait state code
- Enabled wait
- Hang of a subsystem, an address space, or a job

Disabled Wait with Wait State Code

The system loads a wait state program status word (PSW) and stops processing. A valid wait state PSW has one of the following formats:

```
X'000A0000  
rrrrrwww'  
X'000A0000 8rrrrwww'
```

In the format:

000A0000

The A is bits 12 through 15 (the CMWP bits).

rrrrr

8rrrr

A supplement code for the wait state code appears in bits 32 through 51.

The wait state determines the size and position of the supplement code. If the supplement code was loaded with the LOADWAIT macro, this field starts with an 8.

Usually the supplement code is a reason code. Some wait states do not provide a supplement code in the PSW. See the description of the individual wait state code for more information.

www

The wait state code. The IBM-supplied wait state codes appear in *z/OS MVS System Codes*. Installation-supplied programs can use wait state codes X'FF0' through X'FFE'; see installation-supplied information for these codes.

Disabled waits can be restartable and non-restartable.

Restartable Waits

A restartable wait is one of the following:

- An attempt by the operating system to communicate with the operator. When the system cannot send a message to a console, the system may use a restartable wait state to contact the operator and obtain a response.
- A way to preempt processing. For a SLIP trap with an action of wait, the system will issue a message, then enter a restartable wait.
- A symptom of a problem.

Wait in a System in a Sysplex

If a system in a sysplex enters a wait, it does not update its status in the couple data set within a specified time. Another system in the sysplex will detect this delay and take actions. See "Loss of a System in the Sysplex" on page 6-6.

Recovery Actions for a Restartable Disabled Wait

1. Perform the actions in the operator response for the wait state code.
Responding may mean altering the contents of storage using the alter/display function on the system console. The operator response for the wait state code in the *z/OS MVS System Codes* book tells the operator to use alter/display, when necessary.
2. Restart the system by using a system console frame or by entering a RESTART service language command.

If the restart is successful, the system will begin operation. If not all central processors (CPU) are restarted, the system will issue a message. Recover the CPUs that were not restarted by configuring them *offline* and then back *online*.

Recovery Actions for a Non-restartable Disabled Wait

1. Perform the actions in the operator response for the wait state code.
2. If the wait state code is not in *z/OS MVS System Codes* or is not installation-defined, system data was overlaid before the system loaded the wait state PSW. The system programmer needs a dump to diagnose this problem. Do the following:
 - Store system status.
 - Request a stand-alone dump.
3. Then relPL the system.

Enabled Wait

The system stops processing without giving any indication of an error. The system apparently has nothing to do. The problem is an **enabled wait**. This wait is also called a **no work wait** or a **dummy wait**. Indications of an enabled wait are:

- The program status word (PSW) contains X'070E0000 00000000'.
- The system activity display (SAD) frame shows no activity on any CPU or the partition.
- The job entry subsystem does not respond to any commands. For example, in a JES2 system, enter a \$DI1 command. JES2 does not respond.

Causes

The most common causes of an enabled wait are that the system is waiting for:

- Work. The system has no active jobs to process or all active jobs are swapped out.
- An operator reply or other action.
- Missing interrupts. The system is waiting for a critical device, which is busy, not ready, reserved by another system, or has a mount pending. If the system residence (SYSRES) or paging (PAGE) volumes have missing interrupts, the operator may not get a message.
- System resource. Work is waiting for a resource, which can be a lock, queue, input/output (I/O) device, page, or device allocation.

Recovery Actions for an Enabled Wait

Before recovering the system, do the following to look for the reason for the wait and, if possible, to correct the problem:

1. Enter a DISPLAY R,LIST command to display outstanding messages requiring operator action. Perform any outstanding actions.
If the system does not answer the command, you cannot communicate with the system. Check the console for I/O errors by looking in the LOGREC error records. Obtain hardware support to repair the device.
2. Press the VIEWLOG key on the system console to look at the processor controller element (PCE) system console log. For the messages that indicate errors, perform the actions in the operator response for the message.
3. Enter a DISPLAY A,LIST command to show the active jobs and started tasks. If the display shows no active jobs or all jobs are swapped out, then enter JES commands to determine if jobs are waiting to be processed. For JES2, enter:
 - \$D i to determine if initiators are active
 - \$D Q to determine if jobs are waiting in each queue
4. Enter a DISPLAY U command to look for any critical device with a status of BSY, MTP, or NRD. This device is probably causing the wait.
5. In a system with JES2, look for message \$HASP050, which indicates a resource shortage. Correct the shortage.
6. Dump the master scheduler address space with the following commands:
DUMP COMM=(text)
id IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
REPLY id,ASID=1,SDATA=(option,option,...)
7. Use RESTART reason 1 to restart the CPU. Restart must be done with the SYSCTL frame, not with RESTART from the system console, key, or OPRCTL frame.
Upon restart, the system may issue message IEA501I to describe the WTO buffer status and the dispatchability indicators.
8. If nothing has worked, obtain a stand-alone dump.
9. RelPL the system.

Hang of a Subsystem, Address Space, or Job

A subsystem, address space, or job is hung if it does no processing. In a hang, the operator notices that a part of the system, for example, the teleprocessing (TP) network, is not functioning. Or the operator notices that a job has been in the system for a long time and is not doing any processing.

Recovery Actions for a Hung Subsystem

1. If processing can continue without the hung part, let the system continue. Just before the next scheduled relPL, request a stand-alone dump.
2. If processing cannot continue, use the DUMP operator command to request a dump. Then cancel the subsystem and start it again.

Recovery Actions for a Hung Address Space or Job

1. Enter a DISPLAY JOBS,ALL command to obtain the address space identifier (ASID) for the hung address space or the address space containing the hung job.
2. Enter DUMP commands to request two SVC dumps. Before entering the DUMP command for the second dump, make sure that the first dump has completed.
 - Request the first dump. Reply with the ASID of the master scheduler address space.
 - After the first dump completes, request the second dump. Reply with the ASID of the hung address space.

The first dump shows the dispatchability indicators for the hung address space. The second dump shows the hung address space itself, but with changed dispatchability indicators.

Chapter 3. Recovery from Loops

A loop is the endless repeating of a sequence of instructions. Loops tie up system resources and may impact other work. The types of loops are:

- Enabled loops, which can be interrupted.
- Disabled loops, which cannot be interrupted.

A *spin loop* is a disabled loop used by the operating system. A spin loop is defined as a central processor (CPU) spinning disabled to wait for a resource held by another CPU in the same system.

Loop in a System in a Sysplex

If a system in a sysplex enters a loop, it may not update its status in the couple data set within a specified time. Another system in the sysplex will detect this delay and take actions. See “Loss of a System in the Sysplex” on page 6-6.

Enabled Loops

Enabled loops allow some or all interrupts. The loops are usually caused by an error in an application program. All or most of the loop will be in code running in problem state, but the loop can include system code if any instructions in the loop request system services. An enabled loop can run on more than one central processor.

The loop will uselessly consume resources and may take over all system operation.

Stopping Enabled Loops

Usually the system ends a looping job when it exceeds the job or step time limitations defined in:

- The TIME parameter on the JCL EXEC statement, for a job step
- The TIME parameter on the JCL JOB statement, for a job
- The system default time limit, which is specified during JES2 or JES3 initialization

Plan these limitations so that they do not stop long jobs that are running correctly, but do stop looping jobs.

Identifying Enabled Loops

Enabled loops are difficult for the operator to identify. Indications can be:

- Performance slows as the looping program uses large amounts of resources.
- System status line on the system console shows operation that is not normal. Specifically, if the wait indicator never goes on for a processor, this might indicate a loop.
- A job does not change status for a long time.

Preparing for Loop Identification: The system activity display (SAD) frame does not indicate to the operator that an enabled loop is running; therefore, an operator must look at the storage keys of currently running work.

Most installations use six storage keys; a SAD frame can show four storage keys per CPU definition. To show all keys used, define multiple sets of SAD frames. For consistency, make the keys the same for all CPUs in a set.

Actions to Identify a Loop in Native Mode

1. Use SAD frames to monitor all CPUs and all storage keys in the installation. If the first set of frames does not show a loop, stop it and start another set of frames.
2. When a loop is suspected, activate all the loop diagnosis SAD frames together with a time interval. If a CPU shows 100% utilization in one key, a loop in the programs that use that key is likely.

The following are the storage keys and the programs that run in the keys:

Storage Key	Programming
0	Supervisor/operating system
1	JES2 or JES3
2	
3	Availability manager
4	
5	Data Facility Product (DFP)
6	Telecommunications access methods (VTAM or TCAM)
7	Information Management System (IMS) and some Customer Information Control System (CICS) functions
8	Applications (V=V)
9	Applications (V=R)
.	.
.	.
15	Applications (V=R)

Recovery Actions for an Enabled Loop

1. Use RMFMON to identify the looping task. The ARD option in delta mode may show the task that consumes most of the processor time.
2. End the job using the MVS operator command:

```
CANCEL jobname,DUMP
```

If the system programmer asks for problem data, do the following, as requested:

1. In a system with JES2, if JES2 is looping, enable JES2 trace identifiers 1 and 2 for all processor controller elements (PCEs) or 11 and 12 for specific PCEs involved in the problem.
2. In a system with JES2, enable JES2 trace identifiers 18 and 19, if the loop appears to be in an application program.
3. If the loop occurs after system initialization, recreate the problem. Just before the problem recurs, ask the system to trace branches with the following command:

```
TRACE ST,BR=ON
```

Use a DUMP command to obtain an SVC dump. Then restart the system, using reason 0. If restart does not work, obtain a stand-alone dump and reIPL the system. The dumps contain the system trace data.

Disabled Loops

Disabled loops allow no I/O or external interrupts. Only authorized programs can be disabled for interrupts; therefore, disabled loops are usually caused by a problem in the system or in installation-provided modifications to the system. A disabled loop runs on one central processor and, in a multiprocessor, quickly ties up the processing on the other processors.

The loop harms system operation.

Identifying Disabled Loops

Disabled loops are easier than enabled loops for the operator to identify. Indications can be:

- In native mode, every time the SAD frame is updated, it shows that a central processor is in 100% utilization in supervisor state.
- Console communications are locked out. To check for communication with the console, an operator can enter a simple command, such as DISPLAY T. The system will not respond.

Address Instruction Trace

On an ES/3090, the tracing is called an *address instruction trace*. All configured CPUs are placed in ISTEP (instruction step) and the manual (stopped) state. Tracing occurs for all CPUs that are configured online, starting with the lowest numbered CPU. All other configured CPUs are left in the manual state while the target CPU is being traced. When all CPUs have been traced, or if the operator stops the trace operation via the CNCL key, the CPUs are taken out of ISTEP and the manual state and are placed in the running state.

On an ES/4381 Processor Complex, the tracing is called a *trace stop*. The trace records a preset number of instructions. The processor is stopped after the trace.

The data that the trace recorded can be obtained from a dump written after completion of the trace.

Operator Actions for a Disabled Loop

Any time a disabled loop is suspected, the operator should initiate an instruction address trace to identify the instructions in the loop.

1. Activate instruction address tracing from the SYSCTL frame on the system console. In response, the processor saves the PSW, registers, and instruction addresses in a buffer. The trace occurs in all online CPUs, in turn, starting with the target CPU.
2. Do one of the following:
 - Restart the CPU that seems to be in the loop, then request an SVC dump by a DUMP command. Initiate restart, specifying restart reason 0.
 - Request a stand-alone dump, then relPL the system.

Note: Trace data taken in a physically-partitioned configuration might be lost if the operator does not request a dump that includes the trace data before a merge to a single-image configuration.

Spin Loops

A spin loop occurs when one processor in a multiprocessor environment is unable to communicate with another processor or requires a resource currently held by another processor. The processor that has attempted communication is the *detecting* or *spinning* processor. The processor that has failed to respond is the *failing* processor.

The detecting processor continuously attempts its communication with the failing processor until either:

- It is successful.
- A specified time interval has passed.

When the communication is not successful within this interval, an *excessive spin loop time out* exists. The detecting processor then initiates recovery processing for the condition.

MVS processing for excessive spin-loop conditions can provide recovery without any actions or decisions from the operator. The major features of excessive spin loop recovery are:

- There are four recovery actions to choose from to resolve an excessive spin loop condition. These actions are defaults or specified in the EXSPATxx member of Parmlib. They are:
 - **SPIN**: Continue spinning for another interval to allow the event to complete
 - **ABEND**: End the current unit of work on the failing CPU but allow the recovery routines to retry
 - **TERM**: End the current unit of work on the failing CPU and do not allow the recovery routines to retry
 - **ACR**: Invoke ACR to take the failing CPU offline
- The system chooses the appropriate action without requiring any decision or action from the operator.

If an action taken in response to an occurrence of an excessive spin loop does not resolve the condition, the system takes the next action when the next excessive spin loop time out occurs. The default order in which the system takes the actions is SPIN, ABEND, TERM, and ACR.

- An installation can change the order of the actions, except the first one, that the system takes.
- For hardware-related errors that formerly caused message IEA490A, the system immediately initiates ACR processing without working through the sequence of actions and without requiring any operator intervention.
- The default spin loop time-out interval is 10 seconds. The installation can change this interval through the combination of a parameter in the EXSPATxx parmlib member and entering the SET command.
- To avoid unnecessary recovery actions, system functions that can validly exceed the interval are exempt from excessive spin-loop processing, so that they will not cause any recovery actions. If they exceed the time-out interval, these system functions do cause an excessive spin loop record to be written to the logrec data set.
- The installation can still control excessive spin loop recovery through operator actions.

Recovery Actions for Spin Loops

The system default sequence for taking the recovery actions applies increasingly severe recovery actions to a condition that persists for several excessive spin-loop intervals. Each time the system detects again an excessive spin loop for the same problem, it takes the next action in the sequence. Whenever the system takes an action, it issues a message to notify the operator. The operator, however, needs to take no action in response to the message.

SPIN Action

The SPIN action is taken on the first detection of most excessive spin loops. This action causes the system to return control to the routine that detected the spin-loop condition without initiating any recovery action. This routine will then again try to communicate with the failing CPU until either it is successful or another spin-loop interval passes.

On the initial detection, part of the system SPIN processing is to write an error record to the logrec data set to provide data about the program that is causing the spin loop, that is, the program running on the failing CPU.

For a condition that the system can immediately identify as being caused by a hardware failure, SPIN is not the first action. In this case, the initial action is ACR. No record is written to the logrec data set.

ABEND Action

The ABEND action causes the system to initiate recovery processing for the program running on the failing CPU. If the recovery routine(s) for this program specify a retry into the mainline program, the operating system allows the retry. If, however, the recovery routine does not specify retry, the unit of work on the failing CPU is ended.

TERM Action

The TERM action also causes the system to initiate recovery processing for the program running on the failing CPU. However, the operating system does not allow the recovery routines to retry; thus, the unit of work on the failing CPU is always ended.

The processing of the recovery routines, the ending of the unit of work, or both frees resources held on the failing CPU and usually resolves the cause of the excessive spin loop.

ACR Action

The ACR action causes the system to end the current unit of work on the failing CPU with an abend X'0F3', and take the failing CPU offline. Unless there is some other indication of a CPU problem, the operator should try to configure the CPU back online by entering a CONFIG CPU(x),ONLINE command after ACR processing is complete.

Specifying the Sequence of Actions

The default for the sequence of recovery actions is SPIN, ABEND, TERM, and ACR. The best approach is to allow the operating system to perform the default actions; they are the least disruptive and require the least operator intervention.

An installation can change the sequence of the recovery actions, after the first one, for a persistent excessive spin loop in the EXSPATxx member of Parmlib. When the operator enters a SET command specifying this member, the system starts the designated sequence of actions on the second detection of a persistent excessive spin loop. SPIN is always the action taken on the first detection.

If EXSPATxx specifies SPIN as an action in the sequence, the system does not create a logrec data set record for these additional SPIN actions.

If the cause of a persistent excessive spin loop is not resolved by the sequence of recovery actions, the system initiates processing to put itself into non-restartable wait state X'0A1'. To avoid this possibility, specify both TERM and ACR as two of the actions in the sequence.

When the system initiates any of the defaulted or specified recovery actions, it issues message IEE178I to inform the operator. This is strictly for information. The operator does not need to take any action for the excessive spin loop.

Operator Control

If an installation wants the recovery actions to be controlled by the operator, it can specify OPER as an action in the EXSPATxx parmlib member. After a SET command is entered to this parmlib member, when the OPER action is reached in the recovery sequence for a persistent problem, the system stops automatic recovery processing and issues message IEE331A for that detection, and for every subsequent detection of that excessive spin loop. When that particular excessive spin loop has been resolved, any subsequent excessive spin loop causes the sequence of automatic recovery actions to start with SPIN and proceed through the sequence again.

If the system cannot issue message IEE331A to the master console, it sends the message to the system console.

Message IEE331A includes a brief description of the event causing the problem.

```
IEE331A PROCESSOR (y) IS IN AN EXCESSIVE DISABLED SPIN LOOP
      WAITING
      FOR event REPLAY U OR SPIN TO CONTINUE SPIN, REPLY
      ABEND TO
      TERMINATE WORK ON PROCESSOR (x) WITH RETRY, REPLY TERM TO
      TERMINATE WORK ON PROCESSOR (x) WITHOUT RETRY, OR STOP
      PROCESSOR (x) AND REPLY ACR. (AFTER STOPPING THE
      PROCESSOR, DO NOT START IT) PREVIOUS REPLY WAS INVALID,
      ENTER A VALID REPLY
```

event is one of the following:

```
CPU IN STOPPED STATE

OPERATOR INTERVENING STATE

RISGNL RESPONSE

LOCK RELEASE

RESTART RESOURCE

ADDRESS SPACE TO QUIESCE

INTERSECT RELEASE

SUCCESSFUL BIND BREAK RELEASE
```

Reply U on the console displaying the message to continue in the spin loop. If the message recurs, reply ABEND.

If the system cannot access the master and system consoles, it enters restartable wait state X'09x'.

The following table indicates the recommended operator recovery actions for each event and lists the associated wait state X'09x'. To continue, the operator can restart the processor that is in wait state X'09x'.

Associated Wait State 09x	Event in the IEE331A Message Text	Primary Action	Secondary Action
091	RISGNL RESPONSE	Continue	ABEND
092	LOCK RELEASE	Continue	ABEND
N/A	RESTART RESOURCE	Continue	ABEND
095	ADDRESS SPACE TO QUIESCE	Continue	ABEND
097	INTERSECT RELEASE	Continue	ABEND
09E	SUCCESSFUL BIND BREAK RELEASE	Continue	ABEND

Reference Books

See the following:

- Wait state X'09x' in the *z/OS MVS System Codes* for the full operator recovery actions
- *z/OS MVS Diagnosis: Procedures* for loop diagnosis, including spin loop diagnosis

Chapter 4. Recovery from JES2 Problems

For problems in JES2 that require recovery, prepare recovery actions for the following:

- Problems during JES2 initialization. See the next topic.
- Errors in the spool data set or in checkpoint input/output (I/O). See “Errors in Spool Data Set or Checkpoint I/O.”
- Errors in Remote Job Entry (RJE). See “Errors in Remote Job Entry (RJE)” on page 4-2.
- Errors in Network Job Entry (NJE). See “Errors in Network Job Entry (NJE)” on page 4-3.

Reference Books

See the following:

- *z/OS JES2 Commands* for the commands
- *z/OS JES2 Diagnosis* for the DEBUG facility

Problems during JES2 Initialization

Problems can occur during JES2 initialization. The symptoms are:

- JES2 abnormally ends with a 4-digit decimal user completion code of \$ddd.
- JES2 fails to initialize.

Recovery Actions for a JES2 Initialization Problem

1. If you manually entered the START JES2 command, check it. If the command is in error, reenter it correctly.
2. Enter a DISPLAY JOBS,LIST command to look for tasks on the started task queue. In the display message IEE104I or IEE105I, look for STARTING in place of a job or step name. (Note that STARTING can also appear if the system cannot locate the procedure specified on a START command.) If STARTING appears, reenter the START JES2 command.
3. If JES2 still fails to initialize, re-IPL the system. Disable exits 0 and 19, if present, and any exits changed since the last JES2 start. The error causing the problem is probably in a changed exit.

If the system programmer asks for problem data, do the following, as requested:

1. Restart JES2 with the LIST option. This option writes the JES2 initialization stream to the device defined on the JES2 procedure.
2. Restart JES2 with the following trace identifiers enabled: 1, 2, 3, 6, 7, 11, 12, 13, 17, 18, 19, and 20.
3. Turn on the DEBUG facility.

Errors in Spool Data Set or Checkpoint I/O

Errors on a spool data set cause problems in applications running on the system and for users. Checkpoint I/O errors will impact the system, unless the system runs in backup checkpoint mode. Symptoms of these errors are:

- I/O error messages indicate the spool or checkpoint data set.
- A printer rejects selected work and marks it non-selectable.

- JES2 displays the checkpoint reconfiguration dialog.
- JES2 issues \$HASP095 message.
- JES2 issues \$HASP096 message followed by \$HASP185.
- JES2 issues \$HASP289 message during JES2 initialization.
- JES2 marks jobs as non-selectable because of I/O errors when a JES2 processor selects work.

Recovery Actions for a JES2 Spool or Checkpoint I/O Problem

1. If the problem occurred during JES2 initialization and message \$HASP289 was issued, reply TERM. Then restart JES2 with the RECONFIG start option.
2. If the problem is in the checkpoint data set, do the following:
 - a. Make sure that the checkpoint data set was allocated with DSORG=PSU.
 - b. Make sure that defragmentation programs were not run against volumes containing the checkpoint data set. Defragmenting a checkpoint volume is a prime cause of spool problems.
 - c. If JES2 displays the reconfiguration dialog, the problem is in a backup checkpoint data set. Replace the damaged data set as soon as is convenient.
3. If the problem is in the spool volume, enter a \$ZSPL command for each volume that is in error. This will do the following:
 - Prevent JES2 from trying to allocate any more space from the volume.
 - Prevent all jobs that currently have space allocated from the volume from being selected for work (conversion, execution, printing).

If a system-wide warm start is done eventually without the volumes that had errors, JES2 issues the following messages:

HASP424 *volume* NOT MOUNTED
 HASP853 REPLY GO, QUIT, OR PURGE.

Do one of the following:

- If the volume is thought to be recoverable, reply GO. This will cause the volume to come up in an inactive state. The jobs that have space allocated from this volume will not be available for work (same as above after issuing \$ZSPL). JES2 will not attempt to use any space from the volume until a \$SSPL command is entered.
- If the volume is permanently damaged, reply PURGE to this message. Reply PURGE only if you are certain that the volume cannot be recovered. Replying PURGE will cause all jobs with any space allocated from the volume to be purged. The system issues no messages or warnings to indicate that this has happened.

Errors in Remote Job Entry (RJE)

Common problems in installations that can receive jobs submitted from remote work stations and send the output to the work stations are:

- A remote work station cannot establish contact with the host.
- An I/O error occurs on the RJE lines.
- Output for a remote work station remains in the spool data set, so that output cannot reach the remote work station.
- A remote work station receives incorrect or garbled output.
- A remote work station incorrectly floods the host with input that is destined for another node.

Recovery Actions for an RJE Problem

1. Remove the affected line or lines from use.
2. Restart the RJE on a different line, if available.
3. Check the host and remote modems and communication controllers for the affected line or lines. Contact hardware support for any problems.
4. If a remote work station incorrectly sends input to the host, check the destination for the input. Ensure that the destination is operative.
5. Contact the long-line carrier, if the problem is not in equipment at either end of the line.

Errors in Network Job Entry (NJE)

Common problems in installations with network job entry nodes are:

- A remote node cannot establish contact with another node.
- An I/O error occurs on the NJE lines.
- Output for a node remains in the spool data set.
- A node receives incorrect or garbled output.
- One node incorrectly floods another node with input that is destined for a third node.
- Output cannot reach a node.

Recovery Actions for an NJE Problem

1. Remove the affected node or nodes and the affected line or lines from use.
2. Restart the node on a different line, if available.
3. Check the Virtual Telecommunications Access Method (VTAM) definition of the nodes. Correct it, if needed.
4. Check the modems and communication controllers for the affected line or lines. Contact hardware support for any problems.
5. If a node incorrectly sends input to another node, check the destination for the input. Ensure that the destination is operative.
6. Contact the long-line carrier, if the problem is not in equipment at either end of the line.

Chapter 5. Recovery from JES3 Problems

For problems in JES3 that require recovery, prepare recovery actions for the following:

- Restarting JES3 after a failure
- Recovering JES3 checkpoint data sets
- Using dynamic system interchange in recovery
- Recovering from CTC failures
- Recovering from BSC RJP problems
- Recovering from output writer functional subsystem failures
- Recovering a 3480 tape for a stand-alone dump

Reference Book

See *z/OS JES3 Commands* for the commands.

Restarting JES3 After a Failure

JES3 failsoft provides recovery to avoid JES3 restarts whenever possible. JES3 failsoft in the converter/interpreter (C/I) functional subsystem (FSS) address spaces provides the same type of recovery to avoid C/I FSS abnormal end whenever possible.

- For jobs, restart processing is determined by installation- and programmer-supplied restart parameters.
- For failing JES3 functions or dynamic support programs (DSP), recovery facilities of the operating system, JES3, and failsoft allow the function or DSP to recover, if possible, or be ended. If ended, the system writes a logrec data set error record and writes a dump, depending on the DUMP and WANTDUMP parameters on the OPTIONS initialization statement.

If a function critical to JES3 processing fails, JES3 or the C/I FSS address space ends abnormally.

The failsoft logout messages (IAT3713) are not saved across a JES3 restart. Part of JES3 initialization is to DOM this message. If AMRF is active and the installation would like to use the message for diagnostic purposes, it must be captured prior to the restart.

After the operating system or JES3 fails, JES3 has to be restarted. When restarting JES3, use the type of JES3 start that disrupts the system least. If the operating system failed, relPL the system before restarting JES3.

Restarting the Global Processor

For a JES3 failure, including abnormal end of the JES3 address space, restart JES3 on the global processor as follows:

- **Hot start:** If the job queue is not involved in the problem.
- **Hot start with analysis:** If the job queue is probably in error.
- **Warm start:** If either type of hot start fails.
- **Warm start with analysis:**
 - After a hardware failure causes JES3 to end abnormally.

If a permanently damaged spool data set or spool device caused the JES3 failure, reallocate the spool data set to the same or a different device. Then restart JES3 to replace the spool data set.

- To replace a spool data set, if the job queue is in error
- **ReIPL the system, then a cold start:** If the hot and warm starts fail.

Reference Book

See *z/OS JES3 Commands* for the sequence of commands for each type of restart.

If you cannot restart JES3 on the global processor, assign the functions of the global processor to a local processor. This local processor then becomes the global processor. Use dynamic system interchange to reassign the processors; see “Using Dynamic System Interchange in Recovery” on page 5-3.

Restarting a Local Processor

Restart JES3 on a local processor, as follows:

- After the operating system fails on a local processor, reIPL the operating system, then do a local start of JES3.
- After JES3 fails on a local processor, do a local start of JES3.
- After doing a cold start or any type of warm start on the global processor, reIPL the operating system on all local processors, then do a local start of JES3 on all local processors.

After an IPL on a local processor, JES3 processes jobs that were previously running on the local processor according to their failure options.

Operator Recovery Action if JES3 Restart on Local Processor Fails

If JES3 cannot be restarted on a local processor, logically remove the processor from the complex.

1. Enter the JES3 command *S,main,FLUSH

Recovering JES3 Checkpoint Data Sets

The JES3 checkpoint data set or sets, which are allocated using the JES3 catalogued procedure, provide the information needed for a warm start or hot start of JES3 with minimum or no loss of system information.

JES3 writes job-related information to the JES3 checkpoint data set when the information changes in the system. The JES3 checkpoint data set contains the information required to initialize JES3 on either a global or local processor. During a warm or hot start, JES3 restores this information to the system.

Recovering from Permanent Errors on a Checkpoint Data Set

If a permanent I/O error occurs on one of the JES3 checkpoint data sets, recovery actions depend on whether or not the installation allocated one or both checkpoint data sets.

If the only checkpoint data set develops a permanent I/O error:

1. Perform a cold start.

If one of the two checkpoint data sets develops the error:

1. Enter a DISPLAY command for the data set that is not in error. In the display, look for messages indicating problems in the data set.
2. If the display shows problems, perform a cold start.
3. If the display shows no problems, perform a hot or warm start to replace the data set with the error.

In this start, JES3 copies the checkpoint records it finds on the old checkpoint data set onto the new checkpoint data set. When the hot or warm start is finished, JES3 has two complete checkpoint data sets once again.

Recovering when a Checkpoint Data Set is Out of Space

If either checkpoint data set runs out of space, replace the data set. Recalculate the amount of space the checkpoint data set needs. Then allocate a new checkpoint data set that is larger than the old one.

Reference Book

See *z/OS JES3 Initialization and Tuning Guide* to calculate checkpoint data set size.

Using Dynamic System Interchange in Recovery

Use dynamic system interchange (DSI) to reassign the JES3 global function to a JES3 local processor, which then becomes the new JES3 global processor. Use DSI when:

- The global processor is not active.
- The installation wants a local processor assigned as the global processor.

If the global processor is inactive, the operator can invoke DSI to keep the complex running. Once DSI is complete and the old global processor can be reinitialized, the operator can then reinitialize the old global processor as a JES3 local processor without an intervening IPL.

If the global processor is active but the installation needs to assign another processor as the global processor, the operator can invoke DSI. DSI can be used to allow maintenance or to allow use of a different processor as the global.

The basic distinction in the two uses for DSI is the way the global processor is disabled.

Disabling the Old Global Processor

If the global processor is inactive, disable the global processor by performing a system reset. A system reset causes the operating system and JES3 to end. All jobs running on the global processor are ended; JES3 will reschedule them.

All FSS address spaces on the global are also lost. The operator must restart all FSS address spaces that were running before the system reset.

Reference Book

See *z/OS JES3 Commands* for instructions on how to restart FSS address spaces.

If the global processor is active, disable the global processor by:

1. Completing all reconfiguration tasks that require JES3, such as stopping RJP to disable communication lines.
2. Entering a *CALL,DSI command and a *START,DSI command on the console with master authority for the global processor.

If you use a *CALL,DSI command, any output writer FSSs active on the old global processor remain active when the new global connects to the old global. However, if the old global fails as a result of an IPL or system reset, all output writer FSSs active on the old global end abnormally.

After completing the DSI, reinitialize the old global as a local processor without doing a relPL.

Starting a Local Processor as a Global Processor

Enter *CALL,DSI on the console with master authority for a local processor to make it the new global processor. All FSSs running on local processors at the time of the DSI, including the local that is to become the new global, continue processing during and after the DSI.

If a failure occurs during DSI, perform a warm start.

Writing Dynamic System Interchange Procedures

When writing DSI procedures, do the following:

- Tell the operator how to disable the old global processor. If it is not disabled, the job spool may be damaged so that a cold start would be required.
- Tell the operator to cancel DSPs if all of the following apply:
 - The DSPs were called from a console on the old global processor.
 - The DSPs issue input commands to that calling console.
 - The calling console is not valid on the new global processor.
- Tell the operator how to switch to the new global processor global devices that are defined by a JUNIT parameter on the DEVICE statement, if the devices are not already shared.
- Specify as not transferable JES3 functions that use devices that cannot be shared with all processors or that cannot be switched to the new global processor. If the old global processor and the new global processor are different processor models, they may not support the same set of devices.
- Tell the operator where messages will appear on the new global processor. For a DSI, JES3 rebuilds the MSGROUTE and CONSOLE routing tables. This means that, unless the procedures are symmetrical, messages from called DSPs may appear on different MCS consoles on the new global.
- Tell the operator how to connect each operative local processor to the new global processor before a subsequent DSI is performed. Note that, if the local processor

is not connected to the new global processor before another DSI is attempted, the operator must reIPL the local processor before connecting it to the next new global processor.

- When System Network Architecture (SNA) Remote Job Processing (RJP) is active on the old global processor, tell the operator to perform the following Virtual Telecommunications Access Method (VTAM) operations before starting SNA RJP on the new global:
 1. Start VTAM.
 2. Vary the application definition, which contains the JES3 application, online to VTAM.
 3. Vary the required network online to VTAM.
 4. Enter a *CALL,SNARJP command.
- After DSI completes, tell the operator to determine the status of the writer output multitasking facility by entering the command *INQUIRY,MT. Then:
 - If the new global processor is a multiprocessor and the multitasking facility is off, turn it on by entering a *MODIFY,MT=ON command.
 - If the new global processor is a uniprocessor and the multitasking facility is on, turn it off by entering a *MODIFY,MT=OFF command.

When writing the procedures, consider the following:

- After a DSI, FSS address spaces continue operating on the same processor as before. FSS address spaces defined to operate on a specific processor depending on which processor is the global processor (that is, specifying paired system names on the SYSTEM parameter of the FSSDEF statement) change location, if necessary, the next time the FSS is restarted.

Changes to the definition of an FSS address space caused by a *MODIFY command before the DSI remain in effect across the DSI.
- Jobs queued as a result of the SYSTEM=JGLOBAL or SYSTEM=JLOCAL parameters on //*MAIN statements prior to DSI are not requeued to the new global or local processor after the DSI. The jobs remain queued on the processor on which they were previously queued. If that processor becomes available, the jobs can run.

Recovering from CTC Failures

Recover from a channel-to-channel (CTC) failure depending on the number of CTC addresses defined in the DEVICE initialization statement.

Operator Recovery Actions when Alternate CTC is Defined

If an alternate CTC address was defined and a CTC fails, JES3 issues the following messages:

- IAT2070 to identify the failing CTC address.
- IAT2071 to identify the alternate CTC address.
- IAT2075 to remind the operator to switch the hardware to make the logical CTC paths available.
- IAT2076 to request operator intervention.

Reply to IAT2076 as follows:

1. REDRIVE: To try to reestablish communication using the primary address. Use this reply for an intermittent or temporary failure.
2. SWITCH: To switch to the alternate address. Use this reply if the failure seems permanent or if a REDRIVE attempt fails.
3. NO: To stop using the CTC. Use this reply if the error persists on both the primary and alternate addresses. After you reply NO, JES3 can no longer communicate with the local processor that is attached to the CTC adapter.

Operator Recovery Actions when Only Primary CTC is Defined

If only a primary CTC address was defined and the CTC fails, JES3 issues the following messages:

- IAT2070 to identify the failing CTC address.
- IAT2072 to request operator intervention.

Reply to IAT2072 as follows:

1. REDRIVE: To try to reestablish communication. Use this reply for an intermittent or temporary failure.
2. NO: To stop using the CTC. Use this reply if the error persists. After you reply NO, JES3 can no longer communicate with the local processor that is attached to the CTC adapter.

Recovering from BSC RJP Problems

The binary synchronous communication (BSC) remote job processing (RJP) facility tries to recover automatically from errors and suspended operations that would otherwise end BSC RJP processing and require restart of JES3.

If a failure occurs while the BSC RJP facility is processing, BSC RJP issues a message, then does the following to recover:

- Cancels the line, if the line had an error in a line I/O event, in a timer event, or in Remote Terminal Access Method (RTAM) processing.

If an error occurs during line canceling, the line is lost to the JES3 system; BSC RJP continues to service the rest of the lines.

- Varies the line offline, if the error occurred during line starting. The line cannot be restarted until the operator varies the line online again.
- Ignores the error, if it occurs while BSC RJP is processing a message.

Recovering from Output Writer FSS Failures

JES3 tracks each data set sent to an output writer functional subsystem (FSS) until the FSS notifies JES3 that it has printed the data set. If an FSS fails, JES3 reschedules all data sets that were printing at the time of the failure.

An output writer FSS can fail under any of the following conditions:

- The operator relPLs the system.
- Operator commands cancel the FSS.
- The last active writer in the JES3 global address space fails. In this case, the output writer FSS associated with it also ends.

If an FSS fails, JES3 also fails all of the writers that are in the JES3 global address space and that sent work to the FSS.

In all of these cases, JES3 recovers the work the FSS was processing and reschedules it.

To restart the FSS, use operator commands to start a writer dynamic support program (DSP) for a page mode device. The writer DSP restarts the output writer FSS.

If you define more than one device to run under the control of an output writer FSS, and one of those devices fail, the FSS remains active and the remaining devices continue processing work. However, if the failed device is the last active device running under control of an output writer FSS, then JES3 cancels the FSS.

After a hot start, output writer FSSs continue running in their own address spaces unless:

- The FSSDEF initialization statement specified TERM=YES.
- The operator IPLs the processor on which the FSSs were operating.

If an output writer FSS runs out of work before the JES3 global address space restarts, the output writer FSS remains idle until the restart. After the hot start, JES3 restarts the writer DSPs, both hot writers and dynamic writers, associated with output writer FSSs that were active before the hot start. The writer DSPs reestablish contact with the output writer FSSs and work continues as before the hot start.

Recovering a 3480 Tape for a Stand-Alone Dump

To use an IBM 3480 Magnetic Tape Subsystem for a stand-alone dump, the operator has to recover a 3480 if the JES3 global processor failed.

JES3-managed 3480s are always assigned to more than one processor, that is, they are assigned so that they belong to one or more local processors and the global processor.

Operator Recovery Actions for a 3480

Do the following to recover a 3480 for dump output from a JES3 global processor:

1. Determine if a 3480 is assigned to a local processor by entering a DISPLAY U,,,devnum command, where *devnum* is the device number of the 3480. Enter this command from an MCS console attached to a local processor.
2. Enter VARYL commands to unassign a 3480 from each local processor. For example, to unassign a 3480 that is multi-system assigned at address 560, enter:

```
*X,VARYL  
*S,VARYL,560,OFF  
*C,VARYL
```

Do the following to recover a 3480 for dump output from a JES3 local processor:

1. Enter a *VARY,devnum,OFFLINE command, where *devnum* is the device number of the 3480. Enter this command from an MCS console attached to the JES3 global processor.

Chapter 6. Recovery from Sysplex Problems

The problems that might occur in the initialization and running of a sysplex by the cross-system coupling facility (XCF and XES) are described in this chapter. For sysplex recovery, prepare recovery actions for the following:

- Initialization problems. See the next topic.
- Loss of a system in the sysplex. See “Loss of a System in the Sysplex” on page 6-6.
- Signalling path problems. See “Signalling Path Problems” on page 6-8.
- Sysplex Timer problems. See “Sysplex Timer Problems” on page 6-10.
- Couple data set problems. See “Couple Data Set Problems” on page 6-11.
- Coupling facility problems. See “Coupling Facility Problems” on page 6-12.
- Problems causing an abend or wait state. See “Problems Causing an Abend or Wait State Code” on page 6-13.

Severe Errors

For severe errors that occur during XCF or XES initialization or processing, the system puts the system in non-restartable wait state X'0A2'. (See “Problems Causing an Abend or Wait State Code” on page 6-13). In this case, the operator must relPL the system.

Initialization Problems

The following topics describe problems that can occur during initialization of a system in a sysplex.

- “Duplicate System in the Sysplex”
- “Taking Over Sysplex Initialization” on page 6-2
- “Inactive Systems in the Sysplex” on page 6-2
- “No Primary Sysplex Couple Data Set” on page 6-3
- “Problems With Signalling Path Definitions” on page 6-3
- “Incorrect Sysplex Timer” on page 6-4
- “Spin Loop During Nucleus Initialization” on page 6-4
- “Too Many Systems in the Sysplex” on page 6-5
- “Problems with Coupling Facility Initialization” on page 6-5

Duplicate System in the Sysplex

When the operator IPLs a system that has the same system name in the SYSNAME keyword of the IEASYSxx parmlib member as an existing system in the sysplex, XCF issues messages IXC203I and IXC207A.

Message IXC207A requests the operator to specify another COUPLExx parmlib member.

This condition can occur when one of the systems is removed from the sysplex; but, before XCF completes removal of the system, the same system or another system with the same name is initialized into the sysplex.

Recovery Actions for a Duplicate System in the Sysplex

1. Enter the following command on an active system in the sysplex to determine if there are any outstanding WTOR messages that require operator intervention.

`DISPLAY R,LIST`

2. Enter the following command to obtain the status of the systems in the sysplex:

`DISPLAY XCF,SYSPLEX,ALL`

3. Check for XCF messages that might help identify the problem.

If you need to change the system name in the SYSNAME parameter of the IEASYSxx parmlib member and reIPL the system, be very careful because the system name in the SYSNAME parameter is used by many MVS components.

Taking Over Sysplex Initialization

When the operator IPLs a system in multisystem mode, XCF checks to see if this is the first system in the sysplex. If it is the first, XCF starts the sysplex initialization and initializes the couple data sets.

If it is not the first system and XCF detects that the sysplex initialization has been started by another system but is not finished, XCF issues message IXC403D.

Recovery Actions for Taking Over Sysplex Initialization

Message IXC403D requests the operator to decide whether to wait for the other system to finish sysplex initialization or to have this system take over initialization. Do the following:

1. Check messages issued by the other system to see why it had not finished initialization.
2. Reply to IXC403D as follows:

W If the other system is operative, this system should wait until the other system finishes initialization.

I If the other system is down, this system should take over initialization.

Inactive Systems in the Sysplex

When the operator IPLs a system in multisystem mode and all of the systems already in the sysplex are active, this system joins the sysplex.

If one or more systems are considered active, that is they are defined to the sysplex, but none of the systems is actually running, XCF issues message IXC404I, which lists the systems that are active or are initializing in the sysplex, and then IXC405A.

Recovery Actions for Inactive Systems in the Sysplex

Message IXC405A requests the operator to reply:

- | | |
|----------|--|
| I | Take over the sysplex initialization. Any other systems that are active or initializing are put in a wait state and must be reinitialized. This system becomes the only system in the sysplex.

Reply I when all other systems have failed during initialization and you need to start sysplex initialization again. |
| J | Join the sysplex and continue initialization of this system.

Reply J when you want this system to join the existing sysplex. |
| R | Reinitialize XCF on this system.

Reply R when you want this system to stop initialization. You will be prompted for a new COUPLExx parmlib member.

It is important to reply R rather than do a relPL of the system in order to allow XCF to reinitialize properly. |

No Primary Sysplex Couple Data Set

When the operator IPLs a system with PLEXCFG=MONOPLEX or MULTISYSTEM, XCF requires a primary sysplex couple data set. If the COUPLExx parmlib member does not specify a sysplex couple data set, the operator is then prompted for a new COUPLExx parmlib member.

The system is not allowed to be in a sysplex without a defined primary sysplex couple data set that is the same data set shared by all systems in the sysplex and that is formatted for the specific sysplex.

Problems With Signalling Path Definitions

The following problems can occur with signalling path definitions at IPL time:

- Incorrect signalling path definitions
- Not enough signalling path definitions
- Unable to establish connectivity
- Incorrect coupling facility structure definitions

Incorrect Signalling Path Definitions

When the operator IPLs a system into a multisystem sysplex, XCF on each of the systems in the sysplex checks the signalling path definitions in its COUPLExx parmlib member.

If one of the following reasons are found, XCF on the initializing system or on an existing system issues message IXC451I. The reasons detected by XCF on an existing system are:

- **NOT SAME SYSPLEX:** A signalling path on the initializing system is connected to a signalling path on this (an existing) system that is not using the same sysplex name as the initializing system. All paths must connect systems within the same sysplex.
- **COUPLING DATASETS DIFFER:** A signalling path on the initializing system is connected to a signalling path on this (an existing) system that is not using the same sysplex couple data set as the initializing system. All paths must connect systems that use the same couple data set.

The reason detected by XCF on the initializing system is:

- **CIRCULAR PATH:** An outbound signalling path on the initializing system is connected to an inbound signalling path that is also on the initializing system. This is not allowed.

Also message IXC458I may indicate the device address of a signalling path that has been stopped or has failed.

Not Enough Signalling Path Definitions

If there are not sufficient signalling paths defined to provide at least one inbound and one outbound path between each of the systems in the sysplex, XCF on the initializing system issues message IXC453I to indicate that signalling path connectivity could not be established.

The operator is then prompted for a different COUPLExx parmlib member.

Check also for message IXC451I on other systems in the sysplex. This message indicates incorrect signalling path definitions that can result in message IXC453I.

Unable to Establish Connectivity

If signalling paths cannot be established to provide at least one inbound and one outbound path between each of the systems in the sysplex, XCF on the initializing system issues message IXC454I, which indicates that signalling connectivity cannot be established. XCF then issues message IXC455D and requests the operator to respond with:

R To be prompted for a different COUPLExx parmlib member.

INTERVAL=NNN

To specify the number of seconds that XCF is to continue checking for signalling connectivity to other systems in the sysplex. Use INTERVAL when one or more other systems, shown in message IXC454I, are slow to respond to the requests of the initializing system to establish connectivity.

Message IXC451I may indicate incorrect signalling path definitions that can result in message IXC454I.

Incorrect Coupling Facility Structure Definitions

When a coupling facility structure is used for signalling, errors in either definition or size-determination can occur. See *z/OS MVS Setting Up a Sysplex* for a description of using coupling facility structures for signalling.

Incorrect Sysplex Timer

When the operator IPLs a system in multisystem mode, XCF requires the same Sysplex Timer for all MVS systems in the sysplex. If the Sysplex Timer for the system being IPLed is not the same as the Sysplex Timer for the other systems in the sysplex, XCF issues message IXC406I.

Under Virtual Machine (VM) or in a processor complex with the Processor Resource/Systems Manager (PR/SM) feature, if the same simulated Sysplex Timer is specified but the central processor identifiers are not the same, XCF issues message IXC416I.

Spin Loop During Nucleus Initialization

During nucleus initialization, the system may issue a WTOR message and wait in a spin loop for the operator to respond. In this case, XCF can encounter a status update missing condition and cause global resource serialization to disrupt ring processing.

The operator should respond to the WTOR in a timely manner and ensure that global resource serialization has restarted ring processing.

Too Many Systems in the Sysplex

When the operator IPLs a system and the sysplex already contains the maximum number of systems, XCF ends initialization and issues messages IXC202I and IXC207A. These messages also apply to the situation where the sysplex couple data set has not been formatted to support the additional system(s) joining the sysplex.

Message IXC207A requests the operator to specify another COUPLExx parmlib member. The operator might respond with COUPLE=00 to indicate XCF-local mode or with COUPLE=xx to specify the correct COUPLExx parmlib member.

This condition can occur if a sysplex has the maximum number of systems and one of the systems is removed from the sysplex; but, before XCF completes removal of the system, the same system or another system with a different system name is initialized into the sysplex.

Recovery Actions for Too Many Systems in the Sysplex

1. Enter the following command on an active system in the sysplex to determine if there are any outstanding WTOR messages issued by XCF that require operator intervention.

```
DISPLAY R,LIST
```

2. Check for IXC402D and other XCF messages that might help identify the problem.
3. Enter the following to obtain the status of the systems in the sysplex.

```
DISPLAY XCF,SYSPLEX,ALL
```

XCF issues message IXC335I to display system status.

4. Enter the following to determine how many systems the couple data set is formatted to support.

```
DISPLAY XCF,COUPLE
```

If necessary, you can format another sysplex couple data set to support an increased number of systems. Make the newly formatted couple data set active as an alternate using the SETXCF COUPLE,ACOUPLE command, and then use the SETXCF COUPLE,PSWITCH command to make it the primary sysplex couple data set.

Problems with Coupling Facility Initialization

When a sysplex contains a coupling facility, the following types of problems can occur during the initialization process:

- Possible incorrect coupling facility specification
- Connectivity errors
- Inconsistent information in coupling facility and CFRM active policy.

The process of bringing a coupling facility into a sysplex requires not only that the coupling facility and the appropriate CFRM couple data sets are properly configured in the sysplex, but also that data is reconciled between the coupling facility and the CFRM active policy. The reconciliation is required to determine the resolution of any data that might still remain in the coupling facility from a former owner. The system

notifies the operator at different points in the initialization process, and occasionally requests that the operator make a decision about the sysplex configuration or data requirements.

z/OS MVS Setting Up a Sysplex describes the coupling facility initialization process.

Possible Incorrect Coupling Facility Specification

A coupling facility may be used by only one sysplex at a time. Message IXC500I specifies that a system is attempting to gain ownership of a coupling facility for the sysplex, but that this sysplex was not the last owner of the coupling facility. The operator is prompted with message IXC501A to reply either 'Yes' or 'No' to use the coupling facility.

Message IXC523I specifies that structures from another sysplex are in the coupling facility. The operator is prompted with message IXC508A to either 'K' (keep) or 'D' (delete) the structures.

Coupling Facility Connectivity

Once the coupling facility is determined to be eligible for use by a system, message IXC517I is issued, identifying both the system and the coupling facility.

If the coupling facility is not eligible to be used by this system, message IXC518I is issued, identifying the system, the coupling facility, and the reason that the system cannot use the coupling facility.

Inconsistent Information in Coupling Facility and CFRM Active Policy

Before a coupling facility can be used in a sysplex, coupling facility resource management examines the current CFRM active policy, as specified in the CFRM couple data set. CFRM then goes through a reconciliation process to ensure that any remaining data in the coupling facility matches what is defined in the active policy. This reconciliation process might require that the operator make decisions about whether or not to permit the reconciliation to occur.

CFRM Couple Data Set Size: Any structures or connections that remain in the coupling facility, which are not in the current CFRM active policy, must be added to the policy. If the CFRM couple data set has not been formatted with a large enough size to contain these additional structures or connections, the system issues message IXC503I (for structures) or IXC502I (for connections). The operator or system programmer must format a new CFRM couple data set allowing for the increased number of structures or connections and bring it into use in the sysplex. Use the SETXCF COUPLE,ACOUPL command to make the new CFRM couple data set the alternate and then issue SETXCF COUPLE,PSWITCH to make the alternate the primary CFRM couple data set.

Loss of a System in the Sysplex

When the failure detection interval specified in the INTERVAL keyword in the COUPLExx member of Parmlib is exceeded, a condition called *status update missing* can occur.

For example, assume a sysplex of two systems, SYSA and SYSB. The failure detection interval on SYSA specifies the amount of elapsed time at which XCF on SYSB is to initiate system status notification processing for SYSA because XCF on SYSA has not updated its status on the couple data set within the specified time.

Possible causes of a status update missing condition include:

- The system has failed and entered a non-restartable wait state.
- An SVC dump is being obtained on the system and is taking longer than the failure detection interval.
- The system is going through reconfiguration.
- A spin loop is occurring.
- The operator stopped the system.
- The system is in a restartable wait state.
- The system lost access to the sysplex couple data set and is in a non-restartable wait state.
- The system is communicating with the operator by means of a branch entry synchronous WTOR macro.

The first action that XCF on SYSB takes when the failure detection interval is exceeded on SYSA is to notify all members that are running on SYSB of each group that also has members on SYSA, via their group exits, that SYSA appears inoperative, that is, its system status has not been updated. The group members on SYSB can then initiate processing for any of their group members on the SYSA system.

The next action that XCF on SYSB takes is to issue message IXC402D, after the amount of elapsed time on the operator notification interval, which is specified in the OPNOTIFY keyword in the COUPLExx parmlib member. The operator must reset SYSA and then reply to message IXC402D.

Automatic Recovery

In a sysplex on a processor complex with the Processor Resource/Systems Manager (PR/SM) feature, the XCF PR/SM policy can provide automatic recovery by allowing a subsystem to move processing from a failing system in a logical partition to a backup system in another logical partition.

Likewise, in a sysplex using sysplex failure management, the SFM policy allows you to specify the recovery actions to be taken, thus avoiding operator intervention. The SFM policy takes precedence over XCF PR/SM policy.

Reference Book

See *z/OS MVS Setting Up a Sysplex* for the XCF PR/SM policy and the SFM policy.

In a sysplex that has enabled automatic restart management, MVS can restart batch jobs and started tasks, running on the failing system, that are registered as elements of the automatic restart manager.

Reference Book

See *z/OS MVS Setting Up a Sysplex* for more information about automatic restart management.

Recovery Actions for Loss of a System in the Sysplex

Message IXC402D requests the operator to respond DOWN or to specify a reprompt interval. Note that if SYSA updates its status before the operator responds, the message is deleted.

1. If MVS on SYSA is down and the operator resets the system and replies DOWN, then XCF on SYSB removes SYSA from the sysplex. SYSA must then be reinitialized.

XCF on SYSB notifies via their group exits all group members in groups with members on SYSA; the notification tells the group members that SYSA is down and taken offline. The group members on SYSB can then continue processing without the members on the removed SYSA. Members on SYSB can access the resources previously held by members on SYSA.

2. If MVS on SYSA is not down, for example, the system had been temporarily stopped, and the operator responds with a time interval, XCF on SYSB then monitors the status of SYSA at the end of the time interval.

If SYSA has not updated its status at the end of the time interval, XCF on SYSB issues message IXC402D again.

Note that if SYSA is taking a stand-alone dump, the operator should respond DOWN because SYSA may be unavailable for a considerable time.

Operations Note: When a status update missing condition occurs, XCF may not be able to communicate with the failed system. MVS component exploiters, such as global resource serialization, and other multisystem applications might be forced to stop work in the sysplex. Therefore, plan to respond to status update missing problems in a timely manner.

SFM Note: See *z/OS MVS Setting Up a Sysplex* for a description of recovery processing using SFM. Depending on the SFM policy definitions specified and on the coupling facility configuration, the system might or might not issue message IXC402D.

Signalling Path Problems

Use the DISPLAY XCF command and Resource Measurement Facility (RMF) reports to obtain information about paths.

In response to the DISPLAY XCF,PATHIN or PATHOUT command, XCF issues message IXC356I, which indicates information about the paths, including the status of signalling paths as:

- STARTING: The path is being verified.
- RESTARTING: The path is being restarted; for example, after a failure.
- LINKING: The path is waiting for a connection to another system.
- WORKING: The path is available for message transfer.
- QUIESCING: The path is being quiesced.
- STOPPING: The path is being removed from service.
- STOPFAILED: The path requires intervention.
- INOPERATIVE: The path is unusable but remains defined to XCF.
- REBUILDING: Rebuild has been initiated for this signalling path list structure.

When a coupling facility structure is being used for signalling, the operator can issue other DISPLAY commands for additional information about the status of the signalling structures:

- DISPLAY CF provides information about space utilization within a coupling facility.

- DISPLAY XCF,CF identifies structures allocated within a coupling facility and identifies each system with connectivity to the coupling facility.
- DISPLAY XCF,STRUCTURE provides information about the attributes of a structure, such as its size, maximum number of supported connections, or disposition. DISPLAY XCF,STRUCTURE also indicates the status of a structure, including whether it is in the rebuilding process.

Also, the operator can enter the DISPLAY U,CTC command to obtain information about the CTC devices.

The following topics describe signalling path problems.

- “Loss of Some Signalling Paths Between Systems”
- “Loss of All Signalling Paths Between Some Systems”
- “Loss of All Signalling Paths Between All Systems” on page 6-10

Operations Note: When signalling paths are down, XCF cannot communicate between systems. MVS component exploiters, such as global resource serialization, and other multisystem applications might be forced to stop work in the sysplex. Therefore, plan to respond to signalling path problems in a timely manner.

Loss of Some Signalling Paths Between Systems

If one or more paths are lost between systems and there is at least one outbound and one inbound path between all systems, XCF uses the remaining paths for message traffic. XCF issues message IXC458I or IXC459I for most problems involving signalling paths.

Some reasons why XCF stops a path include:

- The operator has stopped the path.
- The retry limit of the path is reached.
- The system is removed from the sysplex.
- A subchannel command has failed.
- XCF does not have exclusive use of the path.

Note that a loss of paths can cause performance problems with message traffic because more group members are competing for the remaining paths. Additional outbound and inbound paths can be assigned to the systems via the SETXCF START,PATHIN and PATHOUT commands. If a path is listed as INOPERATIVE, in some cases XCF can automatically restart the signalling path. Thus, the operator does not need to redefine the path on the SETXCF START command. See *z/OS MVS Setting Up a Sysplex*.

Loss of All Signalling Paths Between Some Systems

If one or more paths are lost between systems and there is not at least one outbound and one inbound path between two of the systems in the sysplex, XCF issues message IXC409D. Message IXC409D requests the operator to either retry the path(s) or to remove the system from the sysplex.

If a retry of the paths is not successful, the operator responds with the name of the system to be removed from the sysplex. To avoid removing a system from the sysplex, the operator may be able to specify additional outbound and inbound paths via the SETXCF START,PATHIN and PATHOUT commands.

A loss of paths between two systems can be caused by a channel failure on one of the systems when there is not a second channel providing signalling paths between the systems.

To rejoin the sysplex, a system must be reinitialized with sufficient signalling paths between the systems.

Loss of All Signalling Paths Between All Systems

If all paths are lost between all systems, XCF issues message IXC409D. Message IXC409D requests the operator to either retry the path or paths or to remove the system from the sysplex.

If a retry of the paths is not successful, the operator responds with the name of the system to be removed from the sysplex. This is repeated until there is only one system left in the sysplex. To avoid removing a system from the sysplex, the operator may be able to specify additional outbound and inbound paths via the SETXCF START,PATHIN and PATHOUT commands.

A loss of all signalling paths between systems could be caused by a failure of either the 3088 Multisystem Channel Communication Unit or the Enterprise System Connection (ESCON) Director that connects all systems when there is not a second 3088 or ESCON Director providing additional paths between systems. When coupling facility structures are being used for signalling, path loss can occur because of a coupling facility failure when there are no backup paths available, either on another coupling facility, a 3088, or ESCON Director. Path loss also can occur because of a coupling facility failure when it is not possible to rebuild the structure in another coupling facility.

To rejoin the sysplex, a system must be reinitialized with sufficient signalling paths between the systems.

Sysplex Timer Problems

Systems that specify PLEXCFG=XCFLOCAL and PLEXCFG=MONOPLEX (in the IEASYSxx parameter of Parmlib) are the only sysplex configurations that will survive the loss of the Sysplex Timer.

Sysplex Timer Problem in Sysplex of One System

When XCF determines that the only system in a sysplex has lost access to the Sysplex Timer, XCF continues processing by using the clock on this system.

When the Sysplex Timer becomes available and usable, the system returns to using the Sysplex Timer and other systems can IPL and join the sysplex.

Sysplex Timer Problem in Sysplex of Two or More Systems

One System Loses the Sysplex Timer

When XCF determines that a system in a sysplex of two or more systems has lost access to the Sysplex Timer, XCF issues message IXC462W and puts the system in a non-restartable wait state X'0A2', with reason code X'114'.

If there are one or more other active systems in the sysplex that did not lose Sysplex Timer connectivity, then one of the systems will recognize that the system has become inoperative, based on the failure detection interval for the system. The other system will remove the system from the sysplex. See "Loss of a System in the Sysplex" on page 6-6.

All Systems Lose the Sysplex Timer

If the Sysplex Timer fails (and there is no connected backup Sysplex Timer) so that all systems lose access to the Sysplex Timer, XCF on all systems issues message IXC462W and puts all systems in a non-restartable wait state X'0A2', wait state X'114'

If the Sysplex Timer cannot be made available and usable, then the operator can reIPL one system. XCF on the reIPLed system removes the other systems from the sysplex.

When the Sysplex Timer is available and usable, the reIPLed system returns to using the Sysplex Timer and XCF issues message IXC411I to indicate that the Sysplex Timer is available and usable. Other systems can now rejoin the sysplex via a reIPL.

Couple Data Set Problems

The following topics describe primary and alternate couple data set problems and apply to all types of couple data sets.

- “Loss of the Primary When an Alternate Exists”
- “Loss of the Primary When No Alternate Exists”
- “DASD Failures” on page 6-12

Loss of the Primary When an Alternate Exists

If access to a primary couple data set is lost to any system in the sysplex, XCF automatically switches the alternate couple data set to become the primary couple data set. XCF on each system then continues sysplex operation using the new primary couple data set.

Caution

Any time that the alternate sysplex couple data set becomes the primary sysplex couple data set, ensure that the next system reinitialized into the sysplex specifies in its COUPLExx parmlib member the names of the correct couple data sets.

Loss of the Primary When No Alternate Exists

If access to the primary sysplex couple data set is lost to one or more systems in the sysplex when there is no alternate sysplex couple data set, each of the systems that lost access to the data set is put into non-restartable wait state X'0A2' reason code X'10'.

Before a system can be reinitialized into the sysplex, the operator needs either:

- To correct the problem with the existing primary sysplex couple data set
- To format a new primary sysplex couple data set for use by the sysplex.

For the other types of couple data sets, loss of access to the primary couple data set when no secondary couple data set exists results in the following:

- CFRM — The system loads a non-restartable wait state X'0A2' reason code X'9C' on the affected system; other systems in the sysplex are notified.
- SFM — When the loss of access is detected by SFM, the SFM policy will no longer be active.

- WLM — If in goal mode, the loss of access has no effect unless the operator attempts to change the active policy.

DASD Failures

If recoverable I/O errors are occurring on the primary couple data set, consider asking the operator to enter the SETXCF COUPLE,PSWITCH command.

Analyze the logrec data set error records to determine the severity of the errors with the couple data set.

Coupling Facility Problems

If access to a coupling facility is lost by any system in the sysplex, the primary considerations must be to preserve the data stored in the coupling facility and ensure the data's integrity.

The following topics describe coupling facility problems.

- “Failure of a Coupling Facility”
- “Failure of a Structure in a Coupling Facility”
- “Connectivity Failure to a Coupling Facility or a Structure”

Failure of a Coupling Facility

The application using the coupling facility must be aware that if a coupling facility fails, loss of data in the coupling facility can occur. When the coupling facility is reIPLed, the data stored there is lost, and it is the responsibility of the application to have documented recovery mode processing available. For example, when users are notified of a coupling facility failure, the users can attempt recovery actions such as rebuilding the coupling facility structures in another coupling facility, if one is available.

Failure of a Structure in a Coupling Facility

The failure of a structure in a coupling facility may be the result of a single structure failure or the failure of the entire coupling facility. For each structure failure, MVS invalidates the connect token of all affected connectors and notifies these connectors about the structure failure through the Structure Failure event presented to the event exit.

The connectors to the failed structure can either disconnect from the failed structure or can synchronize cleanup of the failed connection and rebuild the structure into a new location. MVS deallocates the failed structure when there are either no active connections to the structure or when the rebuild process for the structure is complete.

Connectivity Failure to a Coupling Facility or a Structure

When a system loses connectivity to a coupling facility or to a structure within a coupling facility, MVS issues a message indicating the connectivity loss. MVS invalidates the connect token of all affected connectors and notifies these connectors about the loss of connectivity through the Loss of Structure Connectivity event in the event exit. To avoid the loss of data in each structure to which connectivity has been lost, connectors on other systems can initiate the rebuilding of the structure in another coupling facility.

If a data path to the coupling facility fails, MVS uses the redundant data path to ensure continued operations. If a coupling facility request fails, MVS retries the

operation and determines the status of the retried request. MVS logs hardware errors and performs associated recovery actions. When MVS determines that an error is permanent, that information is returned to the coupling facility user.

Problems Causing an Abend or Wait State Code

For certain problems, XCF issues abend X'00C' or wait state X'0A2'. The wait state can be restartable or non-restartable, as indicated by the reason code.

XES issues abend X'026' or wait state X'0A2' in certain error conditions.

Reference Book

See *z/OS MVS System Codes* for the description of these codes.

Chapter 7. Recovery for Global Resource Serialization

Recovery for a Star Complex: Recovery from processing problems in global resource serialization depends on the version of the MVS/System Product or OS/390. Starting with OS/390 Release 2, serializing global resources can be accomplished using the coupling facility in a global resource serialization star complex. The coupling facility eliminates the concept of a “ring disruption”, that is, the time period that global resource serialization is unable to process requests due to cleanup activity required when a system leaves the ring complex. Processing done by global resource serialization for a system leaving the star complex amounts to processing a DEQ for each resource held by a requestor on that system. No rebuild of the topology (that is, disruption of the ring) is required.

Star Rebuild Processing: Requests for global resources in a global resource serialization star complex are handled by the ISGLOCK lock structure on the coupling facility. A rebuild of the lock structure can be initiated by one of the following events:

- Loss of connectivity to the coupling facility containing the global resource serialization lock structure
- Failure of the global resource serialization lock structure
- Operator command, **SETXCF START,REBUILD=ISGLOCK**
- Failure of the coupling facility that contains the global resource serialization lock structure

In all of the above instances, global resource serialization quiesces global resource activity and begins processing the rebuild. The particular activity that caused the rebuild (reflected in the state of the structure before the rebuild) determines the action taken by global resource serialization. If a rebuild stop event is received, either from the sysplex failure management (SFM) policy or operator command (**SETXCF STOP,REBUILD,STRNAME=ISGLOCK**), global resource serialization terminates the current rebuild.

Table 7-1. Conditions for Rebuilding the Lock Structure

State of the ISGLOCK Structure	Action Taken
Lock structure or coupling facility containing the lock structure failed.	Global resource serialization starts new rebuild of lock structure.
Loss of connectivity to the coupling facility containing the lock structure.	• Systems that lost connectivity are partitioned from the complex. • The remaining systems process requests through the original lock structure.
Rebuild initiated by the operator.	Global resource serialization resumes processing requests through the original structure.

In the event of a loss of connectivity to the coupling facility, coupling facility structure failure, or operator initiated rebuild command, the following message is issued on each system notified by global resource serialization of a rebuild in progress.

ISG323A GLOBAL RESOURCE SERIALIZATION STOPPED ON *sysname*.

LOCK STRUCTURE (ISGLOCK) REBUILD IS DUE TO *reason*.

As global resource serialization on each system is notified global resource serialization requestors will be suspended due to a rebuild in progress. Each system will then proceed to perform its part of the rebuild process.

While the global resource serialization lock structure is being rebuilt, no systems are allowed to join the star complex. If a system attempts to join the complex during the rebuild, the following message is issued:

```
ISG318I GRS INITIALIZATION IS SUSPENDED UNTIL LOCK  
STRUCTURE (ISGLOCK) REBUILD IS COMPLETE.
```

In the case of a structure or connectivity error, the first system in the complex to be notified will respond by automatically initiating a rebuild of the lock structure. XES will determine if a rebuild is warranted under the current policy. If so, global resource serialization will continue with the rebuild process. If not, the rebuild will be stopped and global resource serialization will attempt to return to and use the original lock structure. Any systems without connectivity to the original lock structure will be terminated with a X'0A3' wait state with reason code X'C8' or X'D0'. For either of the reason codes, the following message is issued:

```
ISG309W GRS PROCESSING TERMINATED. UNRECOVERABLE FAILURE  
DURING LOCK STRUCTURE, REBUILD PROCESSING.
```

When a lock structure rebuild successfully completes, the following message is issued:

```
ISG325I GRS LOCK STRUCTURE (ISGLOCK) REBUILD HAS  
COMPLETED ON sysname.
```

Recovery for a Ring Complex: The remainder of this chapter provides recovery actions when the systems in a global resource serialization ring complex are not all using the coupling services of the cross-system coupling facility (XCF). Specifically, at least one of the systems in the global resource serialization complex is one of the following:

- MVS/ESA SP Version 4 system in XCF-local mode
- MVS/ESA SP Version 4 system in single-system sysplex mode
- MVS/SP Version 3 or earlier system

The examples in this chapter assume that global resource serialization uses channel-to-channel (CTC) links.

Recovery when Some Ring Complex Systems Are in Sysplex

In a global resource serialization complex that has more than one system in the same sysplex, perform the sysplex recovery first; see Chapter 6, “Recovery from Sysplex Problems.” Performing the sysplex recovery actions may remove the need for performing actions for global resource serialization recovery. For example, when a system in the sysplex fails, another system in the sysplex will automatically do the needed global resource serialization purge when the system is removed from the sysplex.

Recovery when Sysplex and Ring Complex Contain Same Systems

When all the systems in a global resource serialization ring complex become part of the same sysplex, recovery becomes automatic. For global resource serialization recovery in a sysplex, see Chapter 6, “Recovery from Sysplex Problems.”

Reference Books

See the following:

- *z/OS MVS Planning: Global Resource Serialization* for information about global resource serialization
- *z/OS MVS Setting Up a Sysplex* for information about sysplexes

Global Resource Serialization Ring Disruption

Systems in a global resource serialization ring complex pass a ring system authority message (RSA-message) as the primary means of communication. Problems that prevent the RSA-message from passing around the ring disrupt the ring. During a ring disruption, none of the operating systems in the ring processes any global resource requests; however, some or all of the operating systems might still be running.

A ring disruption can be caused by:

- **Failure in a global resource serialization link:** Global resource serialization detects a link failure almost immediately.
Successful recovery from a link failure results in a ring of all systems in the complex. This recovery requires a fully-connected complex with alternate links.
To plan for recovery from a link failure, determine if a backup 3088 Multisystem Channel Communication Unit should be provided. A backup 3088 would eliminate the 3088 as a single point of failure.
- **Failure of a system in the complex:** Global resource serialization detects a system failure as fast as the RSA-message overdue tolerance interval permits. Failure includes any system problem, including slow performance and writing of a dump.

A failure can also include removal of a system by XCF. For more information, see Chapter 6, “Recovery from Sysplex Problems.”

Successful recovery from a system failure results in a ring of all systems in the complex except the failed system.

Needed Recovery Actions

For problems in a global resource serialization ring complex with some systems not in a sysplex, prepare recovery actions for the following:

- Purge a system from the complex before reIPL. See the next topic.
- Ring disruption in progress during IPL. See “Ring Disruption in Progress during IPL” on page 7-5.
- Primary link failure. See “Primary Link Failure” on page 7-5.
- Alternate link failure. See “Alternate Link Failure” on page 7-7.
- System failures. See “System Failure” on page 7-8.
- Reactivating a quiesced system. See “Reactivating a Quiesced System” on page 7-15.

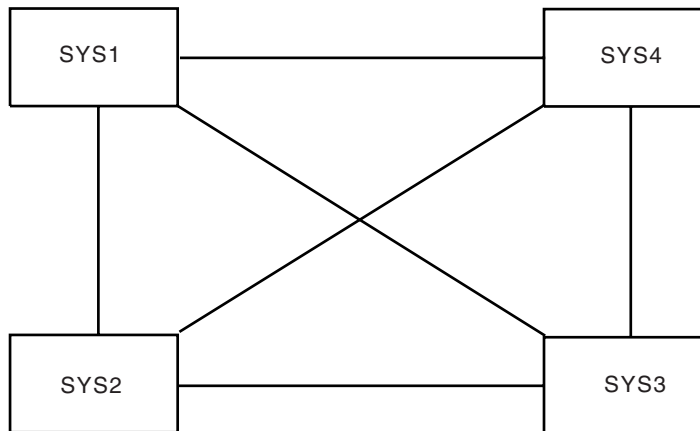
An installation can control the amount of operator intervention needed for complex recovery. As a general rule, minimize operator intervention as much as possible; specify RESTART(YES) and REJOIN(YES) for as many systems as possible in the GRSCNFxx parmlib member.

Reference Book

See *z/OS MVS Initialization and Tuning Reference* for the GRSCNFxx member.

Example Complex

The recovery actions assume a four-system complex, like the following. All systems are in either XCF-local mode or single-system sysplex mode.



Purge a System from the Ring Complex Before ReIPL

Before reinitializing a failed system, the operator must purge the system from an active global resource serialization ring complex.

If the operator does not first purge the system, the following messages appear during reIPL of the failed system (SYS4):

```
ISG006I GRS JOIN OPTION INVALID-SYSTEM SYS4 ALREADY A GRS SYSTEM
```

This message says that the global resource serialization ring complex already knows the system name: SYS4. It means that, because the operator did not purge a failed system from the complex before starting reIPL, global resource serialization finds two systems in the complex with the same name.

ISG006I also appears if the operator types the wrong system name during IPL or specifies the wrong IEASYSxx parmlib member.

```
ISG007I fc-rc ERROR PROCESSING GRS JOIN OPTION.
```

```
ISG009D RELOAD THE SYSTEM OR REPLY JOIN OR NONE
```

For an obvious typing error, the operator should restart the IPL and specify the system name correctly.

Actions to Purge a System before relPL

1. On an active system, enter DISPLAY GRS to check the state of the systems in the complex.
2. On an active system, enter a VARY command to remove SYS4 from the complex:
 - VARY GRS(SYS4),PURGE
3. On SYS4, after the purge command completes successfully, continue to relPL the system. Reply JOIN to message ISG009D.

The IPL continues. Message ISG011I appears on all systems to indicate that SYS4 has successfully joined the complex.

Ring Disruption in Progress during IPL

If a system failed and the operator purged it from the complex, the operator should wait until the disruption clears up and one system has restarted. Then the operator should relPL the purged system for it to resume processing and rejoin the complex.

Another type of failure can occur during that relPL. During relPL, when the system tries to rejoin the complex, the complex could be experiencing a ring disruption. In this case, the following messages appear on the failed system (SYS4):

```
ISG006I GRS JOIN OPTION INVALID - GRS DISRUPTION MAY BE IN PROGRESS
ISG009D RELOAD THE SYSTEM OR REPLY START OR JOIN
```

On other systems, message ISG023E indicates that a ring disruption occurred.

Recovery Actions for a Ring Disruption during IPL

Assuming that GRSCNFxx specifies RESTART(YES) for all systems, after the failure is corrected so that at least one system has restarted, then do the following:

1. On SYS4, continue with the IPL. Reply JOIN to message ISG009D.

Primary Link Failure

The primary link is the link global resource serialization uses to send the RSA-message. Use a DISPLAY GRS,LINK operator command to identify the links. The response for DISPLAY GRS,LINK shows IN USE for the primary link and ALTERNATE, QUIET, or DISABLED for all other links.

A primary link failure occurs when a channel-to-channel (CTC) link fails while global resource serialization is using the link to send the RSA-message. A system in the complex detects a primary link failure when the link it is using to send the RSA-message receives unanticipated interrupts, cannot perform an I/O operation, or receives unanticipated messages. The cause of the failure is not necessarily the link itself; the cause might also be a channel or software error.

Global resource serialization detects a primary link failure almost immediately. It does not wait for the tolerance interval to expire. On the system that detected the link failure, global resource serialization reports the problem to the operator as a link failure and a ring disruption. Other systems in the complex report only the ring disruption; they detect a problem but not its cause.

If a primary link fails, global resource serialization automatically selects an alternate link, if one is available. If there is no alternate available, global resource serialization tries to rebuild a valid ring using the connections that are available. A fully-connected complex with alternate links can always recover from the loss of a single link.

To permit this recovery without operator intervention, specify RESTART(YES) for all systems in the GRSCNFxx parmlib member.

To recover the disrupted ring, global resource serialization varies the failed link offline and selects an alternate link to replace it. If the complex has only one alternate and it is being used to send the ring acceleration signal, global resource serialization stops sending the ring acceleration signal between the two affected systems. If the complex has two alternates, global resource serialization uses one as the primary link and the other to send the acceleration signal.

For the following sequence of messages, assume a four-system complex and failure of the primary link between SYS1 and SYS2. SYS1 detects the failure. The device number of the failed link is A01. The installation specified RESTART(YES) for all systems.

On SYS1, the following messages appear:

```
ISG046E CTC A01 DISABLED DUE TO HARDWARE ERROR FAILURE CODE=05
ISG047I CTC A01 DISABLED
ISG022E SYSTEM SYS1 DISRUPTED GLOBAL RESOURCE SERIALIZATION DUE
        TO COMMUNICATION FAILURE - GLOBAL RESOURCE REQUESTORS WILL
        BE SUSPENDED
```

These messages indicate that SYS1 was using link A01 as the primary link, but it failed because of a hardware error. The operator should contact hardware support to fix the problem. Global resource serialization attempts to disable the link.

On other systems, the following message appears:

```
ISG023E GLOBAL RESOURCE SERIALIZATION HAS BEEN DISRUPTED - GLOBAL
        RESOURCE REQUESTORS WILL BE SUSPENDED
```

Whether this message or ISG022E appears on the other systems depends on how each system detected the error. This message appears if the system detected the error because the tolerance interval expired or because SYS1 began recovery from the ring disruption. Message ISG022E appears if the system detected the communication failure. In most cases, message ISG023E appears.

Make sure that the operators understand that these messages simply reflect the different points at which the systems detected the error.

On SYS1, the system coordinating the ring recovery, the following message indicates that automatic restart has begun:

```
ISG024I SYSTEM SYS1 INITIATED AUTO RESTART PROCESSING
        After a ring disruption, all systems defined with RESTART(YES) attempt to
        restart the ring. Only one actually coordinates the process; this message
        identifies that system, which will bring all of the others back into the ring.
```

On other systems (SYS2, SYS3, and SYS4 in this example), the following message appears:

ISG025E SYSTEM sysx UNABLE TO INITIATE AUTO RESTART
PROCESSING - PERMISSION GRANTED TO SYS1

This message, where *sysx* is SYS2, SYS3, or SYS4, indicates that this system has given permission to SYS1 to rebuild the ring. It is a message that the operators should expect to see during automatic restart processing. It normally requires no operator action.

As SYS1 rebuilds the ring, assuming that SYS2, SYS3, and SYS4 rejoin the ring in that order, the following messages appear:

On SYS1:

ISG011I SYSTEM SYS1 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS1 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG011I SYSTEM SYS2 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS2 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG011I SYSTEM SYS3 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS3 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG011I SYSTEM SYS4 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION

On SYS2:

ISG011I SYSTEM SYS2 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS2 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS3 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION

On SYS3:

ISG011I SYSTEM SYS3 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS3 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION

On SYS4:

ISG011I SYSTEM SYS4 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION

Recovery Actions for a Primary Link Failure

1. After ring processing resumes, enter DISPLAY GRS, LINK to see the links that are now IN USE.
2. DISPLAY GRS, LINK does not identify a link used for ring acceleration. For this information, use the RMF Monitor I report on communication equipment activity.
3. Contact hardware support for any failed links.
4. Vary links online for the system to be able to use them again.

Alternate Link Failure

An alternate link problem occurs when a CTC link fails while global resource serialization is using the link to send the ring acceleration signal. An alternate link failure does not always disrupt ring processing.

If global resource serialization detects an alternate link failure, it issues messages ISG046E and ISG047I to describe the error.

If global resource serialization was using the alternate link to send the ring acceleration shoulder-tap signal, it tries to locate another alternate link to use for ring acceleration. If one is available, global resource serialization uses it to send the signal. If one is not available, global resource serialization stops sending the signal between the two systems and issues the following message:

```
ISG083E GRS CANNOT SEND ACK-TAP SIGNALS TO SYSTEM sysname
```

Ring processing continues normally, but throughput might slow down on one of the systems because all requests for global resources must wait for ring processing. If an alternate link becomes available, global resource serialization begins using it to send the ring acceleration signal and deletes message ISG083E.

Recovery Actions for an Alternate Link Failure

Do the following:

1. Contact hardware support to repair the problem.
2. When the link is repaired, enter a VARY devnum,ONLINE command to make the link available again.

System Failure

A system in the complex can fail because of a hardware or software error. The failure can be permanent, such as a wait state, or temporary, such as for a spin loop. Global resource serialization on a system waiting to receive the RSA-message detects the system failure when the tolerance interval expires. Global resource serialization recovery is the same regardless of whether hardware or software caused the failure and regardless of the length of the failure. Recovery consists of:

- Rebuilding the complex
- Bringing the failed system back into the complex

Rebuilding the Ring

As a first step, global resource serialization tries to rebuild the ring without the failed system. The recovery requires a fully-connected complex; otherwise, global resource serialization cannot always rebuild a ring of all but the failed system.

Installation parameters control the amount of operator intervention required to rebuild the ring:

- Specify RESTART(YES) in GRSCNFxx to allow the system to restart the complex automatically. IBM recommends that you specify RESTART(YES) and, for a system in a sysplex, RESTART(YES) is required.
- Specify RESTART(NO) to prevent the system from automatically restarting the complex. RESTART(NO) means an operator must intervene to restart the complex manually.

Bringing the Failed System Back into the Complex

The second step in recovery is to bring the failed system back into the complex. The specific actions depend on how the system stopped:

- **The system failed and requires relPL.** Before the relPL, the operator must intervene to purge the system or to remove it from the sysplex.
- **The system temporarily stopped.** The GRSCNFxx parameters control whether or not operator intervention is required, as follows:

- REJOIN(YES) - Global resource serialization will try to bring the system back into the complex automatically. IBM recommends that you specify REJOIN(YES) and, for a system in a sysplex, RESTART(YES) is required.
- REJOIN(NO) - The operator must intervene to bring the system back into the complex.

Recovery Planning for Different Size Complexes

For complexes of three or more systems, specify RESTART(YES) to allow all systems to restart the complex automatically and REJOIN(YES) to allow all systems to rejoin the complex automatically. If there is a reason why you do not want all systems to restart the complex automatically, it is still good practice to specify REJOIN(YES) on all systems.

Recovery planning for a two-system complex is slightly different.

- To increase the probability of ring rebuild, specify RESTART(YES) and REJOIN(YES) on one system. Choose the more important system, or choose one arbitrarily.
- Specify RESTART(NO) and REJOIN(YES) on the other system.

With these parameters, recovery actions depend on which system fails. See “Restart of Two-System Complex” on page 7-13.

Automatic Restart Example

For this example, the operator actions assume a fully-connected four-system complex. Assume that SYS1 failed and that SYS2 detected the failure. Assume also that GRSCNFxx specified RESTART(YES) and REJOIN(YES) for all systems.

On all other systems, the following message appears:

```
ISG023E GLOBAL RESOURCE SERIALIZATION HAS BEEN DISRUPTED - GLOBAL
        RESOURCE REQUESTORS WILL BE SUSPENDED
```

This message indicates that an error has disrupted ring processing, though it does not pinpoint the source of the error. All active systems become inactive; any task that tries to obtain or free a global resource is suspended. All systems then attempt to restart the complex, but only one coordinates the process. In this example, assume that SYS2 is the system that will recover the complex.

On SYS2, the system coordinating the complex recovery, the following message indicates that automatic restart has begun:

```
ISG024I SYSTEM SYS2 INITIATED AUTO RESTART PROCESSING
```

On SYS3 and SYS4, the following message appears:

```
ISG025E SYSTEM sysx UNABLE TO INITIATE AUTO RESTART
        PROCESSING - PERMISSION GRANTED TO SYS2
```

This message, where sysx is SYS3 or SYS4, indicates that this system has given permission to SYS2 to rebuild the complex. It is a message that the operators should expect to see during automatic restart processing. It normally requires no operator action.

As SYS2 rebuilds the complex, assuming that SYS3 and SYS4 rejoin the complex in that order, the following messages appear as each system rejoins the complex.

On SYS2:

```

ISG011I SYSTEM SYS2 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS2 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG011I SYSTEM SYS3 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS3 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG011I SYSTEM SYS4 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION

```

On SYS3:

```

ISG011I SYSTEM SYS3 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS3 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION

```

On SYS4:

```

ISG011I SYSTEM SYS4 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION

```

Global resource serialization has recovered the complex automatically without operator intervention. The recovered complex consists of SYS2, SYS3, and SYS4.

Recovery Actions for a System Failure with Automatic Restart

From an active system, the operator must determine if the failed system, SYS1, stopped temporarily or if the failure requires relPL.

- If SYS1 stopped temporarily, do nothing.

When SYS1 resumes processing, it automatically rejoins the complex because REJOIN(YES) was specified. The following messages appear on all active systems:

```

ISG011I SYSTEM SYS1 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS1 - RESTARTED GLOBAL RESOURCE SERIALIZATION

```

- If SYS1 requires relPL, do the following:

1. On an active system, enter VARY GRS(SYS1),PURGE to remove SYS1 from the complex. The following message appears:

```

ISG017D CONFIRM PURGE REQUEST FOR SYSTEM SYS1 - REPLY YES OR NO
Reply YES. The following message appears:

```

```

ISG011I SYSTEM SYS1 - BEING PURGED FROM GRS COMPLEX

```

2. If the following appears before ISG013I, the message indicates a possible data integrity exposure.

```

ISG016I SYSTEM SYS1 OWNS OR IS WAITING FOR GLOBAL RESOURCES
Notify the system programmer. Give the system log to the system programmer. The
system log normally contains messages about the problem. The following message
describes any resources that might have been damaged by purging the system:

```

```

ISG018I REQUESTORS FROM SYSTEM SYS1 HAVE BEEN PURGED FROM
RESOURCES NAMED xxxx,yyyy

```

3. On SYS1, relPL the system with GRS=JOIN.
4. On the active systems, the following messages appear when SYS1 rejoins the complex:

```

ISG011I SYSTEM SYS1 - JOINING GRS COMPLEX
ISG004I GRS COMPLEX JOINED BY SYSTEM SYS1

```

Manual Restart

The operator actions assume a fully-connected four-system complex. Assume that SYS1 failed and that SYS2 detected the failure. Assume also that GRSCNFxx specified RESTART(NO) and REJOIN(NO) for all systems.

On SYS2, the following messages appear:

```
ISG023E GLOBAL RESOURCE SERIALIZATION HAS BEEN DISRUPTED - GLOBAL  
        RESOURCE REQUESTORS WILL BE SUSPENDED  
ISG025E SYSTEM SYS2 UNABLE TO INITIATE AUTORESTART PROCESSING - THIS  
        SYSTEM IS NOT AUTHORIZED
```

These messages indicate that an error has disrupted ring processing, and that the system is not authorized to restart the complex automatically. All active systems become inactive; any task that tries to obtain or free a global resource is suspended. Message ISG023E and message ISG025E appear on all of the other systems in the complex.

Tasks suspended while waiting for resources will quickly affect throughput on all systems; it is therefore imperative that an operator start recovery immediately.

Recovery Actions for a System Failure with Manual Restart

The recovery actions assume the operator on SYS2 is to initiate complex recovery.

1. On SYS2, enter VARY GRS(ALL),RESTART. Enter this command rather than trying to restart specific systems. Enter the VARY command only once. Otherwise, a split ring may be created.

As SYS2 rebuilds the complex, assuming that SYS3 and SYS4 rejoin the complex in that order, the following messages appear as each system rejoins the complex.

On SYS2:

```
ISG011I SYSTEM SYS2 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS2 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG011I SYSTEM SYS3 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS3 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG011I SYSTEM SYS4 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION
```

On SYS3:

```
ISG011I SYSTEM SYS3 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS3 - RESTARTED GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION
```

On SYS4:

```
ISG011I SYSTEM SYS4 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS4 - RESTARTED GLOBAL RESOURCE SERIALIZATION
```

2. On SYS2, after the restart is complete, enter DISPLAY GRS to verify that the systems are active. Determine if the failure on SYS1 requires relPL or if the operator action stopped SYS1 temporarily.
3. If the failure is temporary, do the following to recover the system:
 - a. On SYS1, resume system processing.
 - b. On SYS2, after determining that SYS1 has resumed processing, enter VARY GRS(SYS1),RESTART.
 - c. On the active systems, the following messages appear when SYS1 rejoins the complex:

```
ISG011I SYSTEM SYS1 - RESTARTING GLOBAL RESOURCE SERIALIZATION
ISG013I SYSTEM SYS1 - RESTARTED GLOBAL RESOURCE SERIALIZATION
```

1. If the failure requires relPL of SYS1, do the following:
 - a. On an active system, enter VARY GRS(SYS1),PURGE to remove SYS1 from the complex. The following message appears:

ISG017D CONFIRM PURGE REQUEST FOR SYSTEM SYS1 - REPLY YES OR NO
 - b. If the following appears before ISG013I, the message indicates a possible data integrity exposure.

ISG016I SYSTEM SYS1 OWNS OR IS WAITING FOR GLOBAL RESOURCES
Notify the system programmer. Give the system log to the system programmer. The system log normally contains messages about the problem. The following message describes any resources that might have been damaged by purging the system:

ISG018I REQUESTORS FROM SYSTEM SYS1 HAVE BEEN PURGED FROM
RESOURCES NAMED xxxx,yyyy
 - c. On SYS1, relPL the system with GRS=JOIN.
 - d. On the active systems, the following messages appear when SYS1 rejoins the complex:

ISG011I SYSTEM SYS1 - JOINING GRS COMPLEX
ISG004I GRS COMPLEX JOINED BY SYSTEM SYS1

Restart of Two-System Complex

For a two-system complex, specify in the GRSCNFxx parmlib member RESTART(YES) and REJOIN(YES) for SYS1 and RESTART(NO) and REJOIN(YES) for SYS2.

If SYS2 fails, SYS1 automatically restarts the complex. If SYS2 was temporarily stopped, it can automatically rejoin the complex when it resumes processing. In this case, the complex recovers without operator intervention.

If SYS1 fails, however, recovery requires operator intervention.

Recovery Actions for a Two-System Complex if SYS1 Fails

1. On SYS2, the following messages appear:

ISG023E GLOBAL RESOURCE SERIALIZATION HAS BEEN DISRUPTED - GLOBAL
RESOURCE REQUESTORS WILL BE SUSPENDED

ISG025E SYSTEM SYS2 UNABLE TO INITIATE AUTO RESTART
PROCESSING - THIS SYSTEM IS NOT AUTHORIZED

These messages indicate that an error disrupted ring processing, and that SYS2 is not authorized to restart the complex automatically. SYS2 becomes inactive; any task that tries to obtain or free a global resource is suspended. Suspended tasks will quickly degrade performance on both systems.

2. On SYS2, enter VARY GRS(ALL),RESTART.

3. On SYS2, the following messages appear:

ISG026I SYSTEM SYS2 MAY CREATE A SPLIT RING IF ANY OTHER GRS
SYSTEM IS ACTIVE. VERIFY THAT NO GRS SYSTEM IS ACTIVE
BEFORE CONFIRMING RESTART

ISG027D CONFIRM RESTART RING FOR SYSTEM SYS2 - REPLY NO OR YES

4. Before replying to message ISG027D, enter DISPLAY GRS or check with the other operator to verify that SYS1 is not active.

- If SYS1 is inactive, reply YES to continue the restart.
- If SYS1 is active, reply NO to avoid a split ring.

On SYS2, reply YES to message ISG027D (assuming that SYS1 is inactive). The following messages appear:

ISG011I SYSTEM SYS2 - RESTARTING GLOBAL RESOURCE SERIALIZATION

ISG013I SYSTEM SYS2 - RESTARTED GLOBAL RESOURCE SERIALIZATION

5. On SYS2, determine if the failure on SYS1 requires relPL of SYS1 or if operator action stopped SYS1 temporarily.
6. If the failure is temporary, no further action is needed. When SYS1 resumes processing, it automatically rejoins the complex because REJOIN(YES) was specified on both systems. The following message appears:

ISG012I RESTART REQUEST PASSED TO SYS2

The following messages appear on both systems when SYS1 rejoins the complex:

ISG011I SYSTEM SYS1 - RESTARTING GLOBAL RESOURCE SERIALIZATION

ISG013I SYSTEM SYS1 - RESTARTED GLOBAL RESOURCE SERIALIZATION

1. If the failure requires relPL of SYS1, do the following:
 - a. On SYS2, enter VARY GRS(SYS1),PURGE to remove SYS1 from the complex. The following messages might appear:

```
ISG017D CONFIRM PURGE REQUEST FOR SYSTEM SYS1 - REPLY
        YES OR NO
```

Reply YES. The following message appears:

```
ISG011I SYSTEM SYS1 - BEING PURGED FROM GRS COMPLEX
```

- b. If the following appears before ISG013I, the message indicates a possible data integrity exposure.

```
ISG016I SYSTEM SYS1 OWNS OR IS WAITING FOR GLOBAL RESOURCES
```

Notify the system programmer. Give the system log to the system programmer. The system log normally contains messages about the problem. The following message describes any resources that might have been damaged by purging the system:

```
ISG018I REQUESTORS FROM SYSTEM SYS1 HAVE BEEN PURGED FROM
        RESOURCES NAMED xxxx,yyyy
```

- c. On SYS1, relPL the system with GRS=JOIN.
 - d. On both systems, the following messages appear when SYS1 rejoins the complex:

```
ISG011I SYSTEM SYS1 - JOINING GRS COMPLEX
ISG004I GRS COMPLEX JOINED BY SYSTEM SYS1
```

Reactivating a Quiesced System

Restarting a quiesced system requires help from an active system — a system that is currently actively participating in the global resource serialization complex. The active system provides current information about global resource requests. A quiesced system does not have this current information. For this reason, a quiesced system cannot make itself an active system without introducing a data integrity exposure.

In contrast, an inactive system has current information about the requests and can make itself an active system without introducing a data integrity exposure.

Under normal conditions, including most ring disruptions, at least one system is active and can restart a quiesced system, or an inactive system can make itself active and can restart a quiesced system.

Under unusual conditions, all systems in the complex might be either quiesced or failed. This condition can occur when the operators quiesce all but one system and then the only active system fails. When all systems are quiesced or have failed, an installation can recover the complex in two ways:

1. Perform a complex-wide IPL. This process is cumbersome and time-consuming, particularly when MVS is still running on the quiesced systems.

2. Allow global resource serialization to reactivate a system. **Reactivating a system** means turning a quiesced system into an active system. It avoids the complex-wide IPL but can introduce data integrity exposures or other problems related to global resources.

Reactivating a system is less disruptive than a complex-wide IPL. Careful planning for the process can reduce the impact of the data integrity exposure.

The recovery procedure shows how to reactivate a four-system complex. The procedure assumes a four-system complex, SYS1, SYS2, SYS3, and SYS4. The following events occurred:

1. At 10:00, the operator entered VARY GRS(*),QUIESCE on SYS4.
2. At 10:01, the operator entered VARY GRS(*),QUIESCE on SYS3.
3. At 10:02, the operator entered VARY GRS(*),QUIESCE on SYS2.
4. At 10:03, the only remaining active system SYS1, failed.

Recovery Actions for Reactivating Quiesced Systems

The operator on SYS4 begins the reactivation.

1. On SYS4, enter DISPLAY GRS. The response does not show SYS1, but does show that all other systems are QUIESCED, and that the links between the quiesced systems are ALTERNATE.
2. On SYS4, enter any one of the following:
 - VARY GRS(ALL),RESTART
 - VARY GRS(*),RESTART
 - VARY GRS(SYS4),RESTART

On SYS4, the following message appears:

ISG121I VARY REQUEST REJECTED - ENTER COMMAND ON SYSTEM SYS2

This message indicates that global resource serialization has determined that another system, SYS2 in this example, has more current resource information. The message tells the operator to enter the restart command on SYS2, which minimizes but does not entirely eliminate the data integrity exposure in reactivating a quiesced system.

3. On SYS2, enter VARY GRS(SYS2),RESTART. The following messages appear:

ISG110E NO ACTIVE SYSTEMS FOUND IN THE GRS COMPLEX. REACTIVATE SHOULD BE ATTEMPTED IF THERE ARE NO ACTIVE GRS SYSTEMS IN OPERATION. REACTIVATE MUST NOT BE ATTEMPTED IF THERE ARE ACTIVE SYSTEMS.

ISG111D CONFIRM REACTIVATE SHOULD BE ATTEMPTED - REPLY NO OR YES

These messages indicate that SYS2 has the most current resource information of all the quiesced systems and has thus accepted the command.

4. Before replying to ISG111D, verify the following:
 - a. The complex contains no ACTIVE or INACTIVE system.
 - b. All required links are operational.
 - c. No system has been temporarily stopped since SYS2 was quiesced.

Also, contact the system programmer, describe the problem, and ask if a complex-wide IPL should be done. If the complex-wide IPL is preferable, reply NO to message ISG111D.

1. *The following steps assume that reactivation is to continue.*

On SYS2, reply YES to message ISG111D. Some of the following messages appear:

ISG112I SYSTEM SYS3 RESPONDED WITH GLOBAL RESOURCE STATUS -- IT
WILL BE PERMITTED TO RESTART AFTER REACTIVATE COMPLETES

This message indicates that the system is responding and does not have the most current information about global resources.

ISG114I RESOURCE STATUS NOT RECEIVED FROM SYSTEM SYS4 -- IT MUST
BE STOPPED BEFORE CONFIRMING REACTIVATE COMPLETION

This message indicates that the system is responding but did not supply information about global resources. The system might have more current information, but it cannot reactivate the complex. (Only systems that include MVS/SP Version 2.2 with the fix for APAR OY13087 can reactivate the complex.) The operator must make sure the system is stopped before proceeding.

ISG113I NO RESPONSE RECEIVED FROM SYSTEM SYS1 - IT MUST BE
STOPPED BEFORE CONFIRMING REACTIVATE COMPLETION

This message indicates that the named system did not respond. In this case, this message confirms that SYS1 has failed. In other cases, the operator must investigate the state of the named system and make sure the system is stopped before proceeding.

ISG115I IF ANY SYSTEMS WERE NOT LISTED IN THE PREVIOUS LIST,
THEY ARE UNKNOWN TO THIS SYSTEM - THEY MUST BE STOPPED
BEFORE CONFIRMING REACTIVATE COMPLETION

ISG116E STOP INDICATED SYSTEMS BEFORE CONFIRMING REACTIVE
COMPLETION. REPLY NO TO CANCEL REACTIVATE

ISG117D CONFIRM REACTIVATE SHOULD BE COMPLETED - REPLY NO OR YES

These messages warn the operator to make a final check before allowing the reactivate process to complete. At this point, you could verify, for example, that no system had joined the complex after SYS2 was quiesced but before SYS1 failed.

Message ISG112I, ISG113I, or ISG114I must appear for every system in the complex. If there is no message for a system, investigate the reason before proceeding. If you find any problems, reply NO to message ISG117D to stop reactivation.

1. *The following steps assume that reactivation is to complete.*

2. On SYS2, reply YES to message ISG117D. The following message indicates that SYS2 is now an active system:

ISG118I REACTIVATE FUNCTION IS COMPLETE. SYSTEM SYS2 HAS
RESTARTED AS A RING OF ONE SYSTEM.

On SYS2, enter the following to complete the recovery:

- a. VARY GRS(SYS1),PURGE to purge SYS1 from the complex
- b. VARY GRS(SYS4),PURGE to purge SYS4 from the complex
- c. VARY GRS(SYS3),RESTART to bring SYS3 back into the complex

3. On SYS1, determine the problem, fix it, then relPL the system with GRS=JOIN.

4. On SYS4, relPL the system with GRS=JOIN.

At this point, the recovery process is complete.

Chapter 8. Recovery for Consoles

Several errors can directly affect operation of the display consoles used by operators. Symptoms may be obvious, such as the screen suddenly fails, messages describe the error, or the keyboard locks. In other cases, the error might not be immediately apparent. Prepare recovery actions for the following.

System Problems

When a system error occurs, one or more of the following can happen:

- The screen blanks out, and then an error message appears in the message area. See “Recovery Actions for an Error Message in the Message Area” on page 8-2.
- There is an abnormal lack of console activity. See “Recovery Actions for a Lack of Console Activity” on page 8-2.
- Messages IEA405E and IEA404A appear, to indicate that console messages are backed up. See “Console Message Backups” on page 8-2.

Console Hardware Errors

When a console hardware error occurs, one or more of the following can happen:

- Error messages are centered on the screen. The remainder of the screen is blank. See “Recovery Actions for Error Messages Centered on the Screen” on page 8-6.
- The screen blanks out, but no error message appears. See “Recovery Actions for a Blank Screen” on page 8-6.
- The screen appears normal, but the keyboard is locked and the operator cannot enter commands. See “Recovery Actions for a Locked Keyboard” on page 8-6.

Other hardware-based problems are:

- Failure of the master console. See “Recovery Actions for Failure of the Master Console” on page 8-7.
- No-consoles condition, when there are no active full capability consoles available. See “Recovery Actions for a No-Consoles Condition” on page 8-7.

Other Procedures

This chapter also tells how to:

- Recover messages lost because of a console failure. See “Recovery Actions for Lost Messages” on page 8-9.
- Process operating system messages at the system console. See “Actions to See System Messages at the System Console” on page 8-9.

Reference Book

See *z/OS MVS Planning: Operations* for information about consoles.

System Programming Problems

Recovery Actions for an Error Message in the Message Area

An error message in the message area at the top of the screen indicates that a recoverable system error occurred.

1. Perform the action specified by the error message.
2. Perform a cancel action. This should restore the screen.
3. Review the messages to make sure that no messages were lost during error recovery. To recover lost messages, see “Recovery Actions for Lost Messages” on page 8-9.

If the message in the message area is accompanied by a wait state, see Chapter 2, “Recovery from Waits or Hangs” for recovery.

Recovery Actions for a Lack of Console Activity

A lack of console activity can be no messages appearing on the console or no response by the system to commands. Causes can be simply a low level of system activity or a problem in message processing in the operating system.

System message processing checks for the end line of a status display or other multiple-line messages. If, for some reason, the system fails to detect an end line, the console would become inactive while it waits for completion of the display or message. This situation would occur because the system presents all lines of a status display or multiple-line message, once it is begun, before presenting any other message on the console.

1. If a full-capability console appears inactive, check the system response by requesting a display of the time. Enter a DISPLAY T command.
The system should respond within a few seconds with the time and date.
2. If it does not, enter a CONTROL C,D command to cancel any status displays being presented on the inactive console.
The console should return to normal activity.
3. If it does not, assume the console has some other problem. Do the following:
 - a. To avoid losing the console's messages, enter one of the following:
 - The CONTROL Q command to reroute the messages to another console or to the hard-copy log.
 - The SWITCH command to reroute the messages to another specific console or to any console in an alternate console group.
 - b. Check for a console hardware error.
 - c. If possible, bring up another console.
 - d. Notify the system programmer.
 - e. If necessary, relPL the system, following normal procedures.

Console Message Backups

The operating system places WTO and WTOR messages in buffers in virtual storage. The WTO buffers hold messages the system has not yet displayed at eligible consoles. Each WTOR buffer holds one WTOR message that the system has displayed but that an operator has not replied to. The MLIM and RLIM parameters on the INIT statement in the CONSOLxx parmlib member specify the maximum numbers of buffers. The RMAX parameter on the default statement in CONSOLxx also affects the number of buffers, because the system cannot have

more buffers than the largest reply identifier value. If the installation does not code these parameters, the system defaults are:

- For WTO messages, 1500 buffers
- For WTOR messages, 10 buffers

In a sysplex, the first system that joins determines the RMAX value. If no RLIM value is set, RLIM is set to RMAX for all systems in the sysplex.

Messages back up when the system cannot free buffers for new messages because the buffers contain old messages.

WTO Buffer Backup

When WTO message buffer use reaches 80% of the limit specified at IPL, the system issues the following message. The system also takes out of hold mode any consoles in hold mode with messages queued.

IEA405E WTO BUFFER SHORTAGE - 80% FULL

If the problem continues and WTO buffer use reaches its limit, the system issues the following message:

IEA404A SEVERE WTO BUFFER SHORTAGE - 100% FULL

Also, the system puts any task requesting a WTO buffer into a wait state or, if the WTO macro specifies MCSFLAG=BUSYEXIT, the system returns to the task with a return code of X'20' in register 15 and a reason code in register 0 that specifies the number of active WTO buffers for the issuer's address space.

WTOR Buffer Backup

When WTOR message buffer use reaches 80% of the limit specified at IPL, the system issues the following message:

IEA230E WTOR BUFFER SHORTAGE - 80% FULL

If the problem continues and WTOR buffer use reaches its limit, the system issues the following action message:

IEA231A WTOR BUFFER SHORTAGE CRITICAL - 100% FULL

Notes about Console Message Backups

- All lines of an out-of-line multiple-line status display that have not been presented occupy message buffers. Therefore, the operator should erase these displays when they are no longer needed.
- The current buffer count can be larger than the specified limit. Even though the buffer count is greater than or equal to the limit, the system always gives an authorized task a buffer unless the storage available for buffers is exhausted. However, until the buffer count goes below the limit, the system puts any non-authorized task requesting a WTOR buffer into a wait state.
- During system initialization, the system does not use the MLIM and RLIM parameter values in the CONSOLxx parmlib member until either the system log or a console becomes active or processing by the nucleus initialization program (NIP) completes. After NIP processing, consoles other than the initialization console become active and buffer space becomes important.
- When activating an extended MCS console, specify the optional alert percentage and an event control block (ECB) address on the MCSOPER macro. Then, when the message data space reaches the specified percentage, the system will post the ECB.

Recovery Actions for a WTO Buffer Shortage

1. Determine why the buffers are full and correct the problem. Possible reasons are:
 - A console is not ready because:
 - A multiple-line WTO message is hanging.
 - An I/O error occurred.
 - One or more consoles have a slow roll time.
 - The console is in roll deletable (RD) mode but the screen is filled with action messages.
 - The buffer limit is too low to handle the message traffic in the system. Either the MLIM value in the CONSOLxx parmlib member is too low, or the system default is not sufficient.
 - If the installation uses extended MCS consoles, the message data space may be full. If it is full, the system cannot move any more messages from the WTO buffers to the data space.
2. If the installation is not using extended MCS consoles, do the following:
 - a. Enter a DISPLAY CONSOLES,BACKLOG command. The display lists all MCS and SMCS consoles that have any outstanding WTO messages.
 - b. To determine the extent of the problem, look in the display at the following line:
MSG: CURR=aaaa LIM=bbbb RPLY: CURR=c LIM=dd SYS=ssssssss PFK=ee

CURR=aaaa
The number of WTO buffers in use

LIM=bbbb
The limit of WTO buffers
 - c. To determine the console responsible for the problem, look in the display at the NBUF values, which give the number of buffers queued to each console. A high value identifies a console that is backed up. If the limit seems adequate, assume that the backed up console is failing and causing undisplayed messages to fill the message buffers.
 - d. If the buffer limit is not adequate, issue a CONTROL M,MLIM command to increase the WTO buffer limit for the duration of the IPL.
3. Notify the system programmer of the buffer problem, if it was not caused by hardware. The system programmer may want to increase the buffer limit before the next IPL or change the way the extended MCS console is activated.

When the number of buffers in use drops below 60% of the limit specified at IPL time, the system issues the following message:

```
IEA406I WTO BUFFER SHORTAGE RELIEVED
```

Recovery Actions for a WTOR Buffer Shortage

1. Reply to WTOR messages that are outstanding.
If any WTORs have rolled off the screen (console roll mode is DEL=R), enter a DISPLAY R,R command to retrieve their texts.
2. To determine the extent of the problem, issue a DISPLAY CONSOLES,BACKLOG command and look in the display at the following line:
MSG: CURR=aaaa LIM=bbbb RPLY: CURR=c LIM=dd SYS=ssssssss PFK=ee
CURR=c
The number of WTOR messages that have been displayed and are waiting for an operator reply
LIM=dd
The limit of WTOR buffers
3. To determine the console responsible for the problem, look in the display at the NBUF values, which give the number of buffers queued to each console. A high value identifies the console that is backed up. If the limit seems adequate, assume that the backed up console is failing and causing undisplayed messages to fill the message buffers.
To raise the limit of WTOR buffers for the duration of the IPL, issue the CONTROL M,RLIM command.
4. If WTOR buffer use often reaches 80% of the limit, the limit might be too low to handle the WTOR message traffic in the sysplex. Notify the system programmer. The system programmer may want to increase the buffer limit before the next IPL.

Recovery Actions for Command Flooding Indications

Command flooding occurs when too many MVS commands are issued at one time, possibly because a program has issued too many MGCRE macros. The first indication of command flooding is message IEE822E COMMANDS ARE AT 80% OF LIMIT IN COMMAND CLASS cc, followed by message IEE806A COMMANDS EXCEED LIMIT IN COMMAND CLASS cc. The CMDS command can be used to correct this situation.

1. Determine why the command flooding occurred.
Issue the CMDS SHOW command. This displays all of the executing commands as well as the commands that are waiting for execution with the time that the command started execution and the job that issued the commands.
2. If most of the commands in the CMDS SHOW output appear to be from the same job or automation program, the job may be in a loop issuing commands, or the job may have legitimately issued a large number of commands. For example, a program may have issued a large number of VARY commands for many devices. If the commands were issued legitimately, and it appears that the commands are being processed, the commands may be allowed to complete execution.
If the commands appear to have been issued in error, take one of the following actions:
 - Use the CANCEL command to cancel the job that is issuing the commands.
 - Use the CMDS REMOVE command to remove the commands that are waiting for execution. CMDS REMOVE cannot remove commands that are already executing.
3. If it appears that an executing command has been running for a long time, it may be hung due to a resource deadlock or other required action. The CMDS ABEND command can be used to ABEND the command that is hung. CMDS ABEND should be used with extreme caution and should be used only as a last resort, as the system could be left in an inconsistent state.

Console Hardware Errors

Recovery Actions for Error Messages Centered on the Screen

If a console hardware error occurs, one of the following pairs of messages can appear centered on the screen:

```
IEE170E RETRYABLE ERROR. RECENT ACTION MAY NEED TO BE REPEATED
IEE170E PRESS THE CANCEL KEY TO RESTORE THE SCREEN
IEE171E CONDITIONAL ERROR. RECENT ACTION MAY NEED TO BE REPEATED
IEE171E PRESS CANCEL TO CONTINUE OR SWITCH CONSOLES
```

1. Perform a cancel action, which may be pressing the PA2 key.

The cancel action should restore most of the screen, including messages displayed in line in the message area, the instruction line, and the warning line. However, the system blanks out the entry area and the PFK line, erases any out-of-line displays, positions the cursor to the first data entry position, and ends message numbering, if active.

Note: If you do not perform a cancel action, the system rewrites the screen (same effect as cancel) after about 30 seconds.

If the keyboard input for the cancel action results in a console hardware error, the system sees this error as a permanent I/O error and automatically switches consoles. The system moves all messages, except status displays, to the new console.

Recovery Actions for a Blank Screen

If the console screen goes blank, the system is probably switching to another console. When the switch is done, the following message should appear on the new console:

```
IEE870I CONSOLE SWITCH, OLD=(consname,system)
      NEW=(consname,sysname), REASON=rrrrr
```

Note: It is normal for the screen of a 3277 Display Station to go blank for a few seconds if the back-tab key is pressed when the cursor is not in the entry area.

1. If the system has switched to a new console, operate the system from the new console.
2. Contact hardware support to fix the error in the old console.

Recovery Actions for a Locked Keyboard

Sometimes the system is unable to blank out the screen. You may find that you cannot enter commands through a console that otherwise appears normal.

Note: Inhibited input, with or without keyboard locking, can also occur for the following system programming problems. See “System Programming Problems” on page 8-2.

- The system enters a wait state.
- The system is in a disabled loop.
- A problem occurs in message processing in the operating system.

1. Look for software problems first, based on messages and logrec data set error records. Many software problems can inhibit console input.
2. If no software problems are found, try to restore the screen by performing a cancel action.
3. If the system has switched to a new console, operate the system from the new console.
4. Contact hardware support to fix the error in the old console.

Recovery Actions for Failure of the Master Console

Normally, the operating system detects failure of the master console. The system automatically switches to another console. The system sends a message to the new console to tell the operator that the console is now the master console. The message also gives the device number of the old console so that it can be repaired.

Occasionally, the system does not detect a failure.

1. If the system does not detect the failure, press the external interrupt key on any system to switch to another console.
2. If no other consoles are available and if an attempt is made to switch the master console, the system sends a message to all active secondary consoles asking the operator to enter a VARY MSTCONS command. In response, one operator should enter a VARY MSTCONS command. Alternatively, if the master console was lost due to a system partition, then bringing the failed system back into the sysplex will relieve the condition.
3. The system accepts and processes the first VARY MSTCONS command entered with the correct syntax from any secondary console. The system sends a message to the new master console to indicate the change and give the device number of the old console.
4. Contact hardware support to fix the error in the old console.

Recovery Actions for a No-Consoles Condition

A no-consoles condition occurs when no full-capability consoles (those capable of input and output functions) are available to the system or sysplex. This condition can happen if no full-capability consoles are active in the sysplex when the master console fails.

In a no-consoles condition, if the system cannot deliver the following messages, the system sends the messages to consoles with the UD attribute:

- Action messages (descriptor code of 1, 2, 3, or 11)
- WTOR messages
- Important informational messages (descriptor code of 12)

If no consoles have a UD attribute, including the hardcopy log device, the system sends undeliverable messages to the system console. The system console, however, receives only messages from the system to which it is attached.

The system tries to notify the operator of a no-consoles condition by doing one or more of the following:

- Sounding an alarm on a device on one or more systems.
- Sounding the service processor alarm on one or more systems.
- Issuing message IEA546E to any console.

```
IEA546E NO FULL CAPABILITY CONSOLES, REASON=reason,
      TO RELIEVE THIS CONDITION:
      PRESS THE ENTER KEY ON ANY AVAILABLE FULL
      CAPABILITY CONSOLE AND PRESS THE EXTERNAL INTERRUPT KEY
OR
      ISSUE VARY CONSOLE OR CONTROL V COMMAND TO
      RESTORE A FULL CAPABILITY CONSOLE
OR
      LOG ON TO A FULL CAPABILITY SMCS CONSOLE.
```

The operator should respond to message IEA546E.

Notes:

1. The operator can run a system or sysplex from extended MCS consoles.
2. If the operator varies the master console offline in a no-consoles condition, the system unallocates all the devices that the system had allocated as potential candidates to become the master console.

Do one of the following to acquire a master console again:

1. Press the attention key on any device that can be allocated as a console on the system or, in a sysplex, on any system. Then issue an external interrupt on the system or, in a sysplex, on any system.
2. If the sysplex has no consoles, initialize a new system that has one or more active full capability consoles with master level authority. The system will select one of the consoles as the master console and issue another message IEE870I.
3. Enter a VARY CN command to activate a full capability console. Enter the command on an MCS extended console.
4. Enter a CONTROL V command to transform a message stream console, which has I/O capability, into the master console.
5. Logon on to an SMCS console.

Recovery Actions if a No-Consoles Recovery Fails

If you cannot recover the master console using the general recovery procedure, do one of the following:

- Use an extended MCS console with master level authority to duplicate the function of the master console.
- Take away from a subsystem, such as JES3, one of its full-capability consoles. For example, if JES3 is active and has two full-capability JES3 consoles (CN1 and CN2), try the following:
 1. Using JES3 commands, direct JES3 to give up control of a full-capability console:


```
*SWITCH,CN2,CN1
*DISABLE,CN2
```

The console is now available.
 2. Place the system console in problem determination mode using the VARY CN,ACTIVATE command.
 3. Press ENTER, REQUEST, or END on the former JES3 console.
 4. Press the external interrupt key on the processor controller. The former JES3 console becomes the new console with master authority.

If the system has only one JES3 console, if JES3 is not active, or if the general recovery procedure does not work, relPL the system.

Recovery Actions for Lost Messages

If a console fails, you might lose some messages. You can recover lost messages, except for the following:

- Inline status displays, which are deleted
- Out-of-line status displays, which remain on the failing console

Do one of the following:

1. Enter a CONTROL Q command to reroute the failing console's message queue to another console or to the hard-copy log. If the failing console is a full-capability console, enter this command from a console with master authority.
CONTROL Q requeues action messages. However, if messages are being queued for display on only one console, the hard-copy log could be the only place they will subsequently appear.
2. Enter a VARY command. This command does not, however, recover multiple-line messages in progress. If the failing console is a full-capability console, do the following to avoid losing the waiting messages:
 - a. Enter a VARY command to make the current console with master authority the new alternate of the failing console.
 - b. Enter a VARY command to make the failing console the console with master authority.
 - c. Press the interrupt key on the processor. This action causes the failing console's functions and waiting messages to be automatically switched to the new console.

Actions to See System Messages at the System Console

During system recovery, the operating system might try to communicate with the operator through a multiple console support (MCS) console defined to a console group and attached to the system. (You define the console group on the SYNCHDEST keyword in CONSOLxx. For information, see *z/OS MVS Planning: Operations*.) If a console group is not defined to SYNCHDEST, the system tries to communicate through the master console, if the master console is not an SMCS console and is attached to the system issuing the message. If the system cannot use the master console, the system tries to use the system console for messages requiring an operator response.

Note: In a sysplex, only the system console attached to the system can receive these messages.

If the system issues a message to the system console, the alarm on the processor controller rings, and a message replaces the screen image on the system console. The message tells you that an operating system message is pending.

1. To display this operating system message, enter the following command on the system console:

F OPRMSG (on an ES/9000 processor)

or

F SCPMSF (on an ES/3090 processor)

or

F SYSMSG (on an ES/308x processor)

2. A response line appears below the operating system message. Do not precede the response on the system console with the characters:

R 00,

Enter the response requested by the operating system message.

3. The operating system message on the system console does not time out. The message remains on the screen until you enter a reply.

To help you debug problems and perform recovery actions when none of your other consoles are available, you can enter VARY CN,ACTIVATE on the system console to place it in problem determination mode. See *z/OS MVS Planning: Operations*.

Chapter 9. Recovery from Hardware Problems

Recovery is the attempt by the hardware, operating system, operator, automation, or any combination of these, to correct system malfunctions and return the system to a state in which it can do productive work. Recovery from some hardware errors is automatic; that is, the hardware recovers without any actions from the operating system or intervention by the operator or automation. Recovery from other hardware errors requires overt actions from the operating system, operator, and/or automation. For example, to keep the system in operation, the operator or the system can configure offline a failing unit, such as a storage element, a processor, or a channel path. The system continues processing, possibly with some degradation.

The process of recovery includes the following:

- Hardware to operating system communication and corrective actions
- Operator to operating system communication and recovery actions

Hardware Problems

This chapter describes the following categories of hardware malfunctions:

- Central processor (CPU) errors
- Service processor damage
- Storage errors
- Channel subsystem errors
- I/O device errors

For each of these categories, the discussion includes the effect on system operation and the recovery actions taken, if any. This chapter also presents some additional recovery actions.

Hardware to Operating System Recovery Actions

When CPU errors, service processor damage, storage errors, or channel subsystem errors occur, except for some I/O errors, the hardware notifies the operating system with a machine check interruption. Machine check interruptions fall into one of three classes depending on the severity of the error. The classes are:

- **Soft (or repressible) errors:** Least severe type. Generally these errors do not affect the operation of the task currently in control. Soft errors can be disabled (repressed) so that they do not cause a machine check interruption.
- **Hard errors:** Malfunctions that affect the processing of the current instruction or make incorrect the contents of hardware areas, such as registers.
- **Terminating error**
- **s:** Malfunctions that affect the operation of a CPU.

Hard and terminating errors are also referred to as *exigent* errors.

Information Provided with Machine Checks

When the hardware detects a failure, it stores the following information about the failure:

- The machine check interrupt code (MCIC), which contains:
 - Information about the severity of the error
 - The time of the error, in relation to the current instruction stream

- An indication of whether the processor has successfully stored additional information about the error

The MCIC is the major interface between the hardware and the operating system, which uses the MCIC to determine what action to take.

- The save areas that contain the values of the general, floating point, control, and access registers, the CPU timer, and the clock comparator.
- The machine check old program status word (PSW), which contains the PSW at the time of error.
- The fixed logout area, which is implemented on only some processor complex models.

Reference Book

See the *Principles of Operation* for the format and content of the MCIC, register and timer save areas, extended interrupt information, machine check old PSW, and fixed logout area.

CPU Errors

CPU errors result from a malfunction of a hardware element, such as a timing facility, instruction-processing hardware, or microcode. When a CPU error occurs, the recovery processing has, in general, two stages depending on the severity and type of error:

1. When possible, the hardware retries the failing operation a certain number of times. If the retry works, the hardware may issue a recovery machine check interruption, which is repressible, so that the operating system can record the error in the logrec data set. After recording, the operating system returns control to the interrupted task.
2. If the error is too severe for hardware retry or the retries fail, the hardware issues either a hard or ending machine check interruption. The system determines the severity of the error and takes the appropriate action, which may range from ending the interrupted task to ending the entire system.

The next topics describe the following CPU errors:

- Soft CPU errors
- Hard CPU errors
- Ending CPU errors
- Vector Facility errors

Then the recovery actions of alternate CPU recovery (ACR) are described.

Soft CPU Errors

The CPU errors that can result in a soft machine check are:

- **System Recovery (SR):** A malfunction has occurred, but the hardware has successfully corrected or circumvented it.
- **Degradation (DG):** A continuous degradation of system performance has been detected.

The operating system does not inform the operator about the occurrence of soft machine checks until the threshold for a given type is reached. The default

threshold set for an SR machine check is 50, and for a DG machine check it is 1. When a threshold for a type of machine check is reached, the system issues message IGF931E.

The MODE command allows the operator to change the threshold value for either SR or DG machine checks, and to specify what processing should be done when the threshold is reached.

- The operator can specify that at the threshold the CPU be disabled for that type of machine check, that is, be put in *quiet* mode.
- If the MODE command specifies RECORD=ALL for a particular type of machine check, the system does not enter quiet mode; it records all instances of the specified type of machine check in the logrec data set. The operating system issues message IGF931E when the number of machine checks reaches a multiple of the threshold. For example, if REPORT=3 is specified, message IGF931E appears after the third, sixth, ninth, twelfth machine checks, and so on.

Numerous IGF931E messages appearing on the console might indicate a performance degradation. In this case, the installation might want to configure offline the processor that is experiencing the errors. Hardware support personnel can repair the offline processor.

Hard CPU Errors

A hard machine check indicates that the current instruction could not complete. The system records the error in the logrec data set. Then the system either abnormally ends the interrupted task or retries the interrupted task at a predefined retry point. Even though the task may be ended, the system usually continues to run.

The CPU errors that cause hard machine checks are:

- **System Damage (SD):** A malfunction has caused the processor to lose control over the operation it was performing to the extent that the cause of the error cannot be determined.
- **Instruction Processing Damage (PD):** A malfunction has occurred in the processing of an instruction.
- **Invalid PSW or Registers (IV):** The hardware was unable to store the PSW or registers at the time of error, as indicated by validity bits in the MCIC. Any error, even a soft machine check, associated with these validity bits is treated as a hard machine check because the operating system does not have a valid address to use to resume operation. The error goes through recovery processing.
- **Timing Facility Damage:** Damage to the following has been detected:
 - TOD clock (TC)
 - Processor timer (PT)
 - Clock comparator (CC)
 - External Time Reference (ETR)

The four types of ETR-related machine checks are: primary synchronization damage, ETR attachment damage, switch to local, and ETR synchronization check.

To overcome the effects of numerous hard machine checks, the MODE command allows the operator to define machine check thresholds for each type. When reached, the thresholds cause the failing processor to be configured offline by alternate CPU recovery (ACR). Thus, the operator can control whether, and to what extent, the system monitors the frequency of hard machine checks, and can define a separate threshold and time interval for each.

The default threshold value for most hard machine checks is 5. The default for PD machine checks is 16. The default for ETR machine checks is 5 in 300 seconds.

Terminating Errors on CPUs

A terminating machine check occurs when the operating system or the hardware considers a failure severe enough that a processor cannot continue operation.

In a uniprocessor (UP), the operating system enters a disabled non-restartable wait state, such as X'A01' or X'A26', and issues the following message:

```
IGF910W UNRECOVERABLE MACHINE FAILURE, RE-IPL SYSTEM
```

In a multiprocessor (MP), the action taken is as follows:

- If the hardware determines that a processor cannot continue operation, it places the processor in a check-stop state and attempts to signal the other processor(s) by issuing a malfunction alert (MFA) external interruption. The hardware issues an MFA when:
 - It cannot store the machine check logout data about the error.
 - It cannot load the machine check new PSW.
 - It is disabled for hard machine checks when a hard error is detected.
- If the operating system determines that a processor cannot continue operation, it attempts to signal the other processor(s) by issuing a Signal Processor (SIGP) instruction to cause an emergency-signal (EMS) external interruption. The operating system issues an SIGP instruction when:
 - The system is processing one machine check when another machine check occurs that cannot be handled.
 - A hard-machine-check threshold, which is an installation option established by entering the MODE command, has been reached.
 - Channel subsystem damage is detected.
 - The content of the MCIC is incorrect.

When a processor receives either an MFA or EMS external interruption for these conditions, the system receives control. The system, in turn, invokes ACR processing, which takes the malfunctioning processor offline and initiates recovery processing for that processor.

In a multiprocessor environment, an MFA or EMS is received by all the other online processors. On the first processor to receive the signal, the system tests and sets a flag before starting to process the error. When the other processors receive the interruption, the system sees that the error is already being processed and returns to the interrupted task.

Terminating Errors on Multiprocessors

In a multiprocessor, failure of some hardware elements may cause a terminating error on more than one CPU. It is possible that a terminating error may occur on a CPU while alternate CPU recovery (ACR) is still processing a terminating error on another CPU. In either case, the system puts the system into non-restartable wait state X'050'.

Vector Facility Errors

If the Vector Facility has a malfunction, it will present one of the following machine checks:

- Vector Facility source error machine check
- Vector Facility failure machine check

These machine checks are represented by two bits in the machine check interrupt code (MCIC).

Vector Facility Source Error Machine Check

A Vector Facility source error is a hard machine check presented with the PD bit and the Vector Facility source bit (VS) set in the MCIC. Vector Facility source errors are not counted as processor damage machine checks for threshold purposes against the CPU, but are counted toward a separate vector source (VS) threshold count. The operator can set the VS threshold in the VS parameter on the MODE command.

For a Vector Facility source error, the system performs these steps:

- Tries to save the vector environment for possible retries during processing of an abend X'0F3'.
- Routes the current work with the abend X'0F3' for possible recovery or abend processing.
- If the threshold of vector source machine checks has not been reached, the system takes no further action. The default threshold is 5 in 5 minutes.
- If, however, the vector source threshold has been reached, the system indicates that a Vector Facility is logically offline and issues this message:

IGF970I VFn NOW OFFLINE. UNRECOVERABLE ERROR DETECTED.

Vector Facility Failure Machine Check

A Vector Facility failure is a soft machine check presented with the Vector Facility failure bit (VF) set in the MCIC. In this case, if the interrupted task is a vector task, its vector status, such as vector registers and clock, are invalidated, and the Vector Facility, but not the CPU, is taken offline. The interrupted unit of work is ended only if it attempts to issue another vector operation. In this case, the work is ended because, even if there are other operational Vector Facilities, the user's vector status at the time of failure cannot be guaranteed.

Recovery Actions for Vector Facility Failures

With either the VF or VS Vector Facility failure, the operator should enter a CONFIG VF ONLINE command. This command may possibly recover the Vector Facility.

Alternate CPU Recovery (ACR)

ACR is a function that is initiated on an operative CPU when that CPU receives a signal that another CPU has had an ending error. ACR has two major functions:

- To configure offline the malfunctioning CPU
- To initiate the release of system resources held on the malfunctioning CPU

If the failing CPU has a Vector Facility or Integrated Cryptographic Feature (ICRF), the Vector Facility or ICRF is also taken offline.

ACR initiates the release of any resources held on the failing CPU by causing control to pass to the recovery routines for the work on the failing CPU. ACR allows the operating system to continue its normal operation on the remaining CPU(s), although the task that was interrupted by the error on the failing CPU might be ended.

When ACR is complete, it issues message IEA858E stating that ACR is complete and identifying the CPU that was configured offline. At this point, the operator can try to configure the failing CPU back online using a CONFIG CPU(x),ONLINE command. The configuration online might, or might not, be successful depending on the error that caused the CPU to be configured offline.

Some hardware malfunctions might cause a subsequent CONFIG CPU(x),ONLINE command to that CPU to fail, or might cause the problem to recur when the CPU is brought back online. In these cases, hardware support personnel need to service the CPU before it can be successfully brought back into the system.

However, if a CPU was configured offline because a threshold was reached or because of an operating system problem, a subsequent request to configure the CPU back online might work.

Service Processor Damage

Permanent Failure

When the system detects that the service processor is permanently, completely failing, the system receives a service processor damage machine check. The system also notifies subsystems about the damage.

For a permanent failure, the system issues the following message:

```
ISN000E THE PROCESSOR CONTROLLER HAS FAILED. SOME CRITICAL SYSTEM
        FUNCTIONS HAVE BEEN DISABLED. AN ORDERLY SHUTDOWN OF THE
        ENTIRE SYSTEM SHOULD IMMEDIATELY BE ATTEMPTED IN ORDER TO
        MINIMIZE THE IMPACT OF THIS FAILURE
```

After this message, the operator can optionally perform an orderly shutdown of the system. Processing can continue, but when a function of the service processor is required, the system may become inoperative. To recover, the operator then performs an initial microprogram load (IML).

Temporary Failure

If a service processor fails temporarily or partially and is in I/O Support Processor (IOSP) concurrent maintenance mode, the system continues operating but cannot perform certain functions.

For a temporary failure, a message with the prefix ARRP is issued to the operator.

In IOSP concurrent maintenance mode, certain functions of the operating system will not work or will work incompletely. These are listed in “Recovery Actions in IOSP Concurrent Maintenance Mode” on page 1-4.

Storage Errors

The hardware detects and corrects storage errors where possible. The system is informed of the error by a machine check interrupt. The system invokes recovery routines.

If the storage error is detected during an I/O operation, however, the operation is ended with either a channel data check or a channel control check, depending on whether the error was encountered during data transfer or fetching of the channel control word (CCW) and indirect data address word (IDAW). No machine check interrupt is generated in this case. Error recovery procedures (ERPs) recover from this type of error.

Soft Storage Errors

The soft storage errors are system recovery (SR) errors with the **storage error corrected** flag set in the MCIC to indicate that the storage controller was able to repair the error.

When a **storage error corrected** (SC) condition occurs, along with **storage degradation** (DS), the system attempts to stop using the affected frame. This action eliminates performance degradation that would result from hardware correction of later occurrences of the same error. It also minimizes the chance that the same problem will later occur as a **storage error uncorrected**.

If the frame contains pageable data, the system moves that data to another frame, and the original frame is marked offline. If the data in the frame cannot be moved, the frame is marked **pending offline**, and is subsequently taken offline if the frame is released or if its contents are made pageable. Note that, before the system takes a frame offline, it tests the frame; if it has no errors, the frame is returned to available status.

The threshold for SR machine checks affects the ability of the system to deal with **storage error corrected** conditions. The default threshold is 50 SR machine checks. The operator can change the SR threshold with the MODE operator command. When the threshold is reached, the system disables SR machine checks. This action prevents a subsequent **storage error corrected** from being presented. The system then does not take any action to remove the affected frame.

Hard Storage Errors

This section deals with these types of hard storage errors:

- **Storage error uncorrected**: Indicates that the hardware could not repair a storage error.
- **Key in storage error uncorrected**: Indicates that the hardware could not repair a storage key that was in error.

When a hard storage error occurs, the operating system attempts recovery. For a storage key problem in a frame containing a virtual page, the operating system tries to reset the key. If the reset fails and the page is not fixed, the operating system moves the page to a new frame, setting the key in the new frame as required.

If recovery cannot repair the error, the operating system either takes the storage frame offline or marks it pending offline. *Pending offline* means that the operating system will take the frame offline when the frame becomes free.

A **storage error uncorrected** condition represents the potential loss of critical data. When this condition occurs with a PD machine check, the system in most cases ends the affected unit of work. If the recovery routines complete successfully so that the affected storage frame is freed, the frame is marked offline and system processing continues. The recovery processing, however, could try to refer to the storage that originally caused the machine check, thus causing further errors. Such action could result in the PD threshold for machine checks being reached, thus taking a CPU offline.

The default threshold for PD machine checks is 16 in 5 minutes. The operator can change this threshold by means of the MODE operator command.

Effects of Storage Errors

Errors in critical areas of storage may cause the hardware system or the operating system to become inoperative. Those areas of storage and the effect of an error are as follows:

- **Hardware storage area (HSA):** An uncorrectable storage error in the HSA causes the system to enter a check-stop state. The system can be recovered by two actions:
 1. Power-on reset (POR) or a SYSIML CLEAR service language command
 2. IPL
- **Nucleus:** A storage error in nucleus pages requires an IPL for recovery. If the IPL fails, recovery requires either a power-on reset or a SYSIML CLEAR, followed by an IPL.
- **Link pack area (LPA), system queue area (SQA), and local SQA (LSQA):** A storage error in SQA could have the same effects as a nucleus storage error. For a storage error in LPA, the operating system handles recovery. Normally, only the associated job is ended with the remainder of the system unaffected.

High Speed Buffer (Cache)

A processor cache error can result in the loss of the processor and possibly the system. The storage frame corresponding to any changed data in the cache is marked with an uncorrectable storage error. Because the cache might contain critical system data, recovery might require an IPL.

Channel Subsystem Errors

If the channel subsystem fails, the hardware generates a **channel subsystem damage** machine check interrupt. The resultant processing enters the entire system into non-restartable wait state X'A19' and issues message IOS019W.

Channel Report Words (CRWs)

When the channel subsystem detects an error, it does the following:

- Builds a CRW that describes the error
- Queues the CRW for retrieval by the operating system
- Generates a machine check interrupt with the **CRW pending** indicator set in the machine check interrupt code (MCIC)

The operating system records the CRW in a logrec data set error record. The CRW contains a code that indicates the source of the error: the channel path, the subchannel, channel configuration alert, or the monitoring facility.

Reference Book

See *Principles of Operation* for additional information on CRWs.

Channel Path Recovery

If the CRW indicates that a channel path caused the machine check, the system attempts to recover the channel path or route I/O down an alternate channel path. If multiple CRWs indicate errors on different channel paths, a failure in the hardware elements common to those channel paths may be indicated.

The channel path conditions fall into two categories:

- Expected: An expected channel path condition occurs as a result of a previous recovery action taken for an unexpected channel path error, and indicates the result of the action.
- Unexpected: An unexpected channel path error occurs with no warning.

The channel path conditions indicated in a CRW are:

- A terminating error condition on the channel path.
- A permanent error on the channel path; a system reset to the channel path has not been done.
- A permanent error on the channel path; a system reset to the channel path has been done.
- An initialized condition on a channel path, that is, an error recovered by the channel subsystem; a system reset to the channel path has been done.

Terminating Error

A terminating error condition is unexpected only; it is never the result of a previous recovery action. A terminating error condition indicates that the channel path is not permanently lost, but cannot be used until the error condition is reset. In this case, the system attempts to reset the channel path. The CRW that results from this reset is an expected CRW, and it will indicate whether the reset corrected the problem in the channel path.

For a failing channel path that has a device with an outstanding reserve, the system handles the condition in three different ways, depending on whether the device supports dynamic pathing, supports unconditional reserve, or does not support unconditional reserve. The system actions are:

- For a dynamic pathing device with multiple paths, no action is taken until the expected CRW is received.
- For a non-dynamic pathing device that supports unconditional reserve, such as a 3350 Direct Access Storage, the system issues an unconditional reserve command to the device to move the reserve to an alternate path.
- For a non-dynamic pathing device that does not support unconditional reserve, such as a 3330 Disk Storage, or for a reserved/assigned device with only one path, the system issues message IOS063E (or IOS062E) to request that the operator stop I/O to the shared devices (see Figure 9-1 on page 9-10 and the accompanying description). The system then tries to recover the channel path.

Note: Since stopping I/O to shared devices may require a certain level of multi-system disruption and coordination, users may wish to avoid this processing. Through the use of the `TERMINAL BOX_LP(device_class1,...device_classN)` statement in the `IECIOSxx` Parmlib member, users can cause devices in the specified device class to be BOXED rather than having to undergo multi-system disruption to recover the channel path to the device. For more information on the use of the `TERMINAL` statement in `IEAIOStxx`, see *z/OS MVS Initialization and Tuning Reference*.

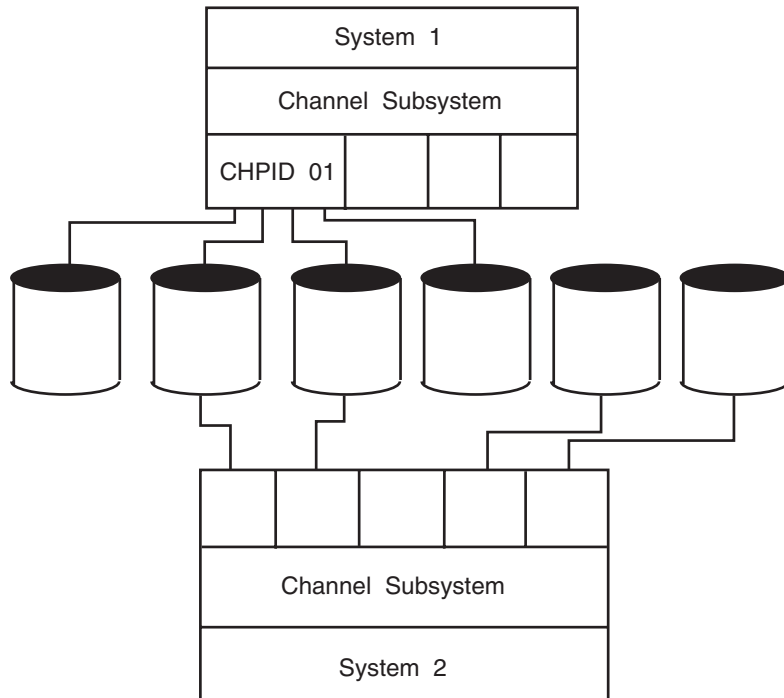


Figure 9-1. DASD Devices Shared Between Two Systems

Figure 9-1 shows two DASD devices that are shared between two systems. When system 1 encounters a channel path error on channel path 01, indicated by message IOS063E (or IOS62E), the operator should stop I/O to the shared devices from system 2 to maintain data integrity during recovery of the channel path.

Recovery Actions for a Channel Path Error with Shared DASD or Assignable Devices

1. Identify which devices on channel path 01 on system 1 are shared with system 2.
2. Enter the IOACTION STOP command on system 2 to stop I/O on the shared devices. The device numbers **may** not be the same on both systems.
3. Restart system 1.
4. Wait for the system to issue message IOS204E (or IOS201E), indicating that channel path recovery is complete.
5. Enter the IOACTION RESUME command on system 2 to allow I/O to resume to the shared devices.

Notes:

1. Do not leave devices in the stopped state any longer than necessary to perform recovery. A shortage of SQA storage can result from stopping I/O for extended periods.
2. Before stopping a device, enter the D U,DASD,ALLOC,xxx command to determine which system resource's I/O will be affected. If any system oriented I/O will be stopped, the system could appear frozen. This situation will last until I/O resumes to the device.

Permanent Error Condition

A permanent error condition, whether expected following a Reset to a channel path in an ending condition or unexpected, results in the system taking that channel path offline. Any active I/O requests are retried on alternate paths if available. If the failing channel path was the last path to any devices, those devices are boxed.

Boxing means:

- The operating system ends I/O to the device.
- Any new I/O request for the device causes a permanent I/O error.
- The operating system does not allocate the device.
- If the device was online, the operating system marks it pending offline. A pending offline device goes offline when the following occur, in this order:
 1. The device becomes no longer allocated to any job.
 2. The operating system allocates any device.

If the device was offline, it remains offline.

Initialized Condition

An initialized condition means that a previous recovery action has successfully recovered the channel path and the channel path is available for use. This condition can be expected only. The initialized condition indicates that the channel subsystem has been successful in recovering the channel path to a state where it is again usable.

For devices that support the Dynamic Path Selection (DPS) feature, such as the 3380 Direct Access Storage and 3480 Magnetic Tape Subsystem, DPS validation is called to restore dynamic pathing arrays for each DPS device attached to that channel path. For each non-dynamic pathing device that does not support unconditional reserve and that had an outstanding reserve on the failing channel path, a reserve command is issued to the device. Any previously active I/O requests are restarted.

Channel Path Alert Conditions

The operating system communicates with the operator when two other indicators are set in a CRW: **channel path temporary** and **configuration alert temporary**. In either case, the operating system performs no recovery processing.

- **Channel path temporary**: The operating system issues message IOS162A to inform the operator that the channel subsystem could not identify the device requesting service.
- **Configuration alert temporary**: The operating system issues message IOS163A to inform the operator that the channel subsystem could not associate a valid subchannel with the device requesting service.

Subchannel Recovery

If the CRW indicates that a subchannel caused the machine check, the operating system examines the error recovery code in the CRW. If the CRW indicates that the subchannel is available, the channel subsystem has recovered from a previous malfunction. I/O functions in progress and presentation of status by the device have not been affected. No program action is required.

If the CRW indicates that the subchannel is **installed parameter initialized**, the operating system determines if the device associated with the subchannel is still valid. If it is, the operating system enables the subchannel again. If, however, the device related to the subchannel is not valid, the operating system marks the device as unusable and issues message IOS151I.

Monitoring Facility Recovery

For a channel monitoring error, the operating system schedules a recovery routine.

I/O Device Errors

An error can occur in an I/O device. The following topics cover:

- “I/O Errors”
- “Missing Interrupts”
- “Hot I/O” on page 9-14
- “Recovery for Hung Devices” on page 9-15
- “Recovery for Failing Devices” on page 9-16
- “Shared Device Recovery” on page 9-17
- “3880/3380 Considerations” on page 9-19
- “DASD Maintenance and Recovery” on page 9-20
- “Recovery for a Failing Alias Unit Control Block (UCB)” on page 9-21

I/O Errors

Errors that are related to an I/O request are usually indicated in the status data provided with the I/O interrupt. These errors are:

- Device not operational on any path
- Device status errors, such as a unit check
- Subchannel status errors: interface control check, channel control check, and channel data check

The operating system processing of the interrupt may include:

- Invoking a driver exit
- Interfacing with attention routines and volume verification processing
- Invoking a device-dependent ERP for error recovery
- Processing an unconditional reserve
- Redriving the I/O request on a channel path other than the one that generated the interrupt
- Requesting an operator action by message IOS115A or by restartable wait state X'115'
- Issuing message IOS050I to inform the operator that a subchannel status error occurred

Missing Interrupts

At predefined intervals, the operating system checks devices of a specific type to determine if expected I/O interrupts have occurred. If an expected interrupt has not occurred across two of these checks, that interrupt is considered missing. The operating system then issues message IOS071I or IOS076E, writes a logrec data set error record, and tries to correct the problem. For recurring missing interrupts, the operating system issues message IOS075E together with message IOS076E or IOS077E to indicate the recurring condition on a particular device.

A feature of the IBM 3990-6 and 9340 attached devices allows MVS/ESA to automatically identify a system in a multisystem environment that is holding a reserve. After every start pending MIH condition, the system attempts to determine whether the device is not responding because of a reserve to another system. If the device is reserved to another system, message IOS431I is issued to identify the system by its central processor serial number. If the system holding the reserve is a

member of the same sysplex as the system detecting the MIH condition, message IOS431I includes the system name and the LPAR ID, if there is one.

For JES2 systems, when the reserve is held by a system in the same sysplex, the system attempts to obtain information about the job causing the reserve by routing a D GRS,DEV=devnum command to that system. JES2 systems which have JES3 installed must have JES2 started with the NOJES3 option (CON=(xx,NOJES3) in order to identify the job holding the reserve. Message ISG020I identifies the jobs holding the reserve on the failing system. The installation can use this information to determine what to do.

Some causes of missing interrupts are:

- An idle unit control block (UCB) with I/O requests queued to it
- An outstanding I/O operation that should have completed
- An outstanding mount for a tape or disk

The intervals used by the operating system to determine whether an expected interrupt is missing varies from 15 seconds for DASD to 12 minutes for 3330 Disk Storage. An installation can define in the IECIOSxx parmlib member the time intervals for all devices in the I/O configuration. These intervals override the IBM-supplied defaults.

Notes:

1. During IOS recovery processing, the system will override your time interval specification and may issue MIH messages and MIH logrec error records at this IOS determined interval.
2. During IPL (if the device is defined to be ONLINE) or during the VARY ONLINE process, some devices may present their own MIH timeout values, via the primary/secondary MIH timing enhancement, contained in the self-describing data for the device. The primary MIH timeout value is used for most I/O commands; however, the secondary MIH timeout value may be used for special operations such as long-busy conditions for long running I/O operations. Any time a user specifically sets a device or device class to have an MIH timeout value that is different from the IBM-supplied default for the device class, the value will override the device-established primary MIH time value. This implies that if an MIH time value that is equal to the MIH default for the device class is explicitly requested, IOS will **not** override the device-established primary MIH time value. To override the device-established primary MIH time value, you must explicitly set a time value that is not equal to the MIH default for the device class.

Note that overriding the device-supplied primary MIH timeout value may adversely affect MIH recovery processing for the device or device class.

Please refer to the specific device's reference manuals to determine if the device supports self-describing MIH time values.

Reference Book

See *z/OS MVS Initialization and Tuning Reference* for the IECIOSxx member.

Note: If there are missing interrupts on the devices that contain the system residence (SYSRES) or the page volumes, the operator may not receive any message, because the needed operating system routines are pageable. The operator can learn about the missing interrupts by initiating restart reason 1.

Hot I/O

A hot I/O condition occurs when a device, control unit, or channel path causes continuous unsolicited I/O interrupts. The operating system attempts to recover from a hot I/O condition so that a relPL is not required. For diagnostic purposes, the operating system indicates all hot I/O incidents in logrec data set error records.

The operating system first tries recovery at the device level by issuing the Clear Subchannel (CSCH) instruction in an attempt to clear the hot I/O condition. If the condition is cleared, processing continues normally. If the condition persists, the next recovery action is determined by one of the following:

- The parameters the installation defined in the IECIOSxx parmlib member for hot I/O recovery
- Operator response to the appropriate hot I/O message or restartable wait state for the class of device:
 - Message IOS117A, (IOS110A, or wait state X'110') for non-DASD, non-dynamic pathing device
 - Message IOS118A, (IOS111A, or wait state X'111') for DASD or dynamic pathing device that is reserved
 - Message IOS119A, (IOS112A, or wait state X'112') for DASD or dynamic pathing device that is not reserved

Because IPLs related to hot I/O are generally caused by incorrect operator actions, an installation should use the IECIOSxx parmlib member to make hot I/O recovery more automatic and reduce the need for immediate operator intervention. The following example parameters, when defined in the IECIOSxx parmlib member, tell the operating system how to handle automatic recovery from hot I/O.

Reference Books

See *z/OS MVS Initialization and Tuning Reference* for the IECIOSxx parmlib member.

Example IECIOSxx Parameters for Hot I/O Recovery

The following examples show how to specify the hot I/O recovery parameters in the IECIOSxx parmlib member. The values shown are also the IBM default values.

HOTIO DVTHRS=100

Specifies 100 repeated interrupts as the threshold for the operating system recognizing the condition.

HOTIO DFLT110=(BOX,)

For a non-DASD, non-dynamic pathing device. Box the device on the first occurrence of this condition. On recursion, prompt the operator.

HOTIO DFLT111=(CHPK,BOX)

For a DASD or dynamic pathing device that is not reserved. Attempt channel path recovery for the device on first occurrence of this condition. On recursion, box the device.

HOTIO DFLT112=(CHPK,OPER)

For a DASD or dynamic pathing device that is reserved. Attempt channel path recovery for the device on first occurrence of this condition. On recursion, prompt the operator.

Hot I/O Recommendations

Although it does require more operator intervention, additional experience has shown that a higher level of availability can be achieved by specifying hot I/O recovery options (CHPK,OPER) for all three classes of devices: DFLT110, DFLT111, and DFLT112. These options will allow the possibility of automatic recovery before the system requests operator involvement.

For non-DASD devices (DFLT110), CHPK,OPER would allow one automated recovery attempt and then request direction from the operator, depending on how critical the device is. If the device is non-critical to the operation of the system, such as a printer, the operator could then reply **BOX**. If the device is critical, such as a 3705 Communication Controller, and is properly equipped with multiple paths, such as through a type three channel adapter on the 3705, the operator could reply **CHPF**. Note that the operator should ensure that all critical devices on the same channel path (CHP) have multiple paths before replying **CHPF**.

You can choose recovery options other than the defaults or the general recommendations, because of considerations unique to the installation. For example, if all DASDs are configured with multiple paths, each through a different CHP, you might consider specifying **CHPK,CHPF** for both DFLT111 and DFLT112. This will allow one attempt to recover without loss of resources, then an attempt to recover by removing the CHP attached to the failing device or control unit, but without losing a critical device. If the installation contains DASD devices with only one path, the considerations are the same as for non-DASD.

Note: Since CHPK processing may require a certain level of multi-system disruption and coordination (to stop sharing processors), the CHPK option may not be suitable for all device classes in a particular Hot I/O device grouping. For instance, CHPK processing may be suitable for reserved DASD devices, but may not be suitable for assigned single-path tape devices. CHPK can be avoided on a device class basis by using the HOTIO BOX_LP(device_class1,...device_classN) parameter in the IECIOSxx parmlib member. Using the BOX_LP parameter forces a device to be BOXED for Hot I/O conditions that cause CHPK processing to occur.

Recovery for Hung Devices

When a device appears to be hung, an operator can consider varying the device offline to try releasing the hang condition at the device. However, the VARY OFFLINE command may obtain resources critical to the system and may attempt to issue I/O. Depending on the condition that caused the device to hang in the first place, this may cause the VARY OFFLINE command to also hang.

To avoid this problem, the operator can choose to use the VARY OFFLINE command with the FORCE parameter to mark the device offline and boxed. The FORCE parameter will not obtain resources or issue I/O, so it will complete regardless of any hardware problems with the device. Since the device is boxed, all I/O will be posted back to the I/O issuer in permanent error, which should cause all system resources previously held to be released.

Note: See “Boxed Device - Operator Actions” on page 9-21 for more information on boxing devices.

Reference Book

See *z/OS MVS System Commands* for information on the FORCE option found under the VARY command.

Recovery for Failing Devices

When a device fails, operators can enter the SWAP command to perform dynamic device reconfiguration (DDR). DDR allows the operator to move or swap a demountable volume from a device. When DDR is active, the system dynamically requests the swapping whenever a device encounters device errors. DDR tells the operator to mount the volume on another available device.

Operators can invoke DDR by issuing the SWAP command and specifying the *from* and *to* device numbers.

Reference Book

See *z/OS MVS System Commands* for the SWAP command.

When swapping tape devices, the *from* and *to* devices should have the same density, whenever possible. Swapping devices of unlike but compatible densities can fail the jobs in device allocation at the time of the swap.

On JES3 systems, DDR checks with JES3 to ensure that the *to* device has not been assigned to another job or function. When the swap is complete, DDR notifies JES3.

When a data check occurs in an IBM 3495 Tape Library Dataserver, the system cleans the tape device and retries the failing operation. If the error persists, the system initiates a swap to another eligible system-managed tape library device without involving the operator. The system will try to swap to up to five other devices. If these efforts fail, it issues an error message to the operator and fails the job.

The following devices are supported by DDR:

- 3400 series Magnetic Tape Units.
- Unit record devices. These devices are not swapped by system-initiated DDR; the operator must enter the SWAP command to swap these devices.
 - 1403 Printer
 - 2501 Card Reader
 - 2540
 - 3211 Printer
 - 3505 Card Reader
 - 3525 Card Punch
- Direct access devices. When using a 3348 Model 70F Data Module, operators must ensure that the *to* 3340 has the fixed-head feature installed. When swapping a 3340/3344 with the fixed-head feature, be sure that the *to* device also has the fixed-head feature installed.
 - 3330 Disk Storage
 - 3333 Disk Storage and Control
 - 3340 Direct Access Storage

– 3344 Direct Access Storage

The following devices are not supported by DDR:

- 3344 and 3350 Direct Access Storage with fixed-head are not supported by system-requested DDR.
- 3375 Direct Access Storage.
- 3380 Direct Access Storage.
- Shared DASD devices, unless the device is swapped to itself.
- Any device holding a permanently-resident volume, such as a system residence or page data set volume.
- Graphic or teleprocessing devices.

Shared Device Recovery

When a system, for example system A, is sharing devices with other systems, events on any one system can affect the ability of any or all the systems to access the shared devices. For example, if one of the sharing systems has an allegiance to a shared device, an I/O operation from system A to that device will receive a **device busy** condition. In this case, the I/O operation is held in system A's channel subsystem until the other system ends its allegiance. At that time, system A's I/O to that device can then be processed normally.

However, a problem either on the system that has the allegiance or in the I/O hardware could result in the allegiance not being freed. This could prevent the processing of any pending I/O operations from any of the sharing systems to the device(s) affected by the allegiance. The indication to any sharing systems that had an I/O operation hung by such a condition would normally be message IOS071I, indicating a **start pending** to the device.

There are a number of other conditions that can cause message IOS071I for a shared device:

- Poor performance of programs using the device
- Contention for the device
- Long reserves
- Application errors
- Operator errors

Operator Actions

If message IOS071I occurs for a shared device, the operator should check the operating condition of the sharing systems. If any system is not operational, it could be the cause of the start-pending because the system could still hold an allegiance that it had when it became non-operational. Operator action depends on the condition of the non-operational system:

- If the system is in the check-stop state or in a non-restartable wait state, the operator should immediately initiate an interface reset to that device from the system console of the non-operational system. If the interface reset fails to release the device, the operator should issue a system reset from the system console of the non-operational system.

These actions should be taken before trying to recover the non-operational system, to allow the other operational systems that are sharing I/O with the non-operational system to continue with as little disruption as possible.

- If the system is in the stopped state, the operator should try to determine why it is stopped and, if appropriate, start it again.

- If the system issued message IOS431I, take the actions described in the operator response for that message.

Unconditional Reserve/Alternate Path Recovery

Alternate path recovery permits recovery from control unit or channel-path failures that cause a DASD or string of DASDs to be no longer accessible to the system. Alternate path recovery is performed only after the operating system guarantees ownership of the device by ensuring that the device is reserved to this system.

When the operating system can guarantee ownership of the device, alternate path recovery is performed. This consists of issuing an unconditional reserve (UR) CCW on a non-failing, online path to the device. If the UR CCW is successful, the operating system issues message IOS428I. If the UR CCW fails or if there are no alternate paths to the device, the operating system issues message IOS429I and boxes the device.

If ownership of the device cannot be established, the operating system issues message IOS427A to determine which recovery action is to be performed.

To maintain volume data integrity during a shared DASD recovery process, the operator must stop I/O from the sharing systems until the recovery process is complete. Prior to MVS/SP 4.2.2, the operator stops I/O to a device only by one of the following:

- Cancelling all jobs allocated to the device and varying the device offline on all sharing systems.
- Stopping the sharing processors.

With MVS/SP 4.2.2 and all subsequent releases, the operator should use the IOACTION STOP command to stop I/O to the device with less disruption to system processing.

See Figure 9-2 on page 9-19 for an illustration of three systems sharing a device. Use the following recovery scenario when system 1 issues message IOS427A, indicating a device failure.

1. Determine the device number on each sharing system.
2. To stop I/O requests for the device, enter IOACTION STOP,DEV=devnum on system 2 and system 3.
If the IOACTION STOP command fails because the device is reserved or reserve pending, and repeated attempts to stop the I/O using the command continue to fail, then end the reserving task or stop the system.
3. Reply UR to message IOS427A and wait for recovery completion messages on system 1.
4. After the recovery completion messages, enter IOACTION RESUME,DEV=devnum or IOACTION RESUME,ALL on system 2 and system 3 to return I/O processing to normal.

Notes:

1. Do not leave devices in the stopped state any longer than necessary to perform recovery. A shortage of SQA storage can result from stopping I/O for extended periods.
2. Before stopping a device, enter the D U,DASD,ALLOC,devnum command to determine which system resource's I/O will be affected. If any system oriented I/O will be stopped, the system could appear frozen. This situation will last until I/O resumes to the device.

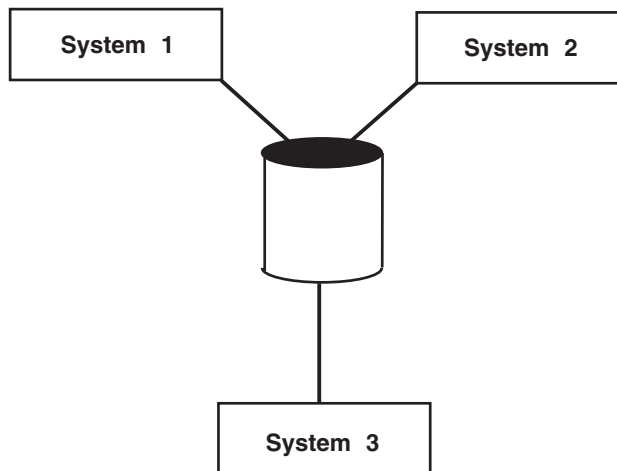


Figure 9-2. DASD Device Shared Between Three Systems

3880/3380 Considerations

The 3380 Direct Access Storage Model AA4 with the 3880 Storage Control is designed to allow concurrent maintenance at the storage director (SD) level. Prior to attempting concurrent maintenance, all paths from all processor complexes through the failing SD to the devices must be varied offline. Failure to vary all paths offline may result in various error symptoms, including interface control checks, path inoperative conditions, and out-of-sync conditions between the 3380 array and the operating system.

Prior to returning a repaired SD to the system, an IML of the SD or a power down-up sequence must be performed to establish a correct copy of the dynamic pathing support array in the 3380 for the repaired SD. The operator should enter VARY PATH ONLINE commands for all paths to all devices through the repaired SD.

Out-of-Sync Conditions

The enable/disable switch on the 3880 A box should NEVER be set to **disable** when any paths to the device are online. Setting the switch to **disable** could cause an **out-of-sync** condition between the array and the operating system. This out-of-sync condition can occur whenever the dynamic path group information maintained in the 3880 A box is reset without notification to the operating system. Any of these operator actions could cause an out-of-sync condition:

- IMLing the 3880 Storage Control
- Disabling the 3880 interface switch
- Disabling the 3380 interface switch

In addition, certain 3880/3380 hardware failures can affect the arrays.

Recovery from an Out-of-Sync Condition

Array out-of-sync conditions may be indicated by missing interrupts or path-inoperative I/O errors. The system provides automatic detection and recovery through the dynamic pathing validation support. This code detects potential out-of-sync conditions, such as missing interrupts, and then validates the physical

path group information. If the dynamic pathing validation code finds a mismatch between the hardware and software path group information, it invokes recovery to rebuild the dynamic path selection arrays.

Dynamic Pathing Support (DPS) Out-of-Sync Recovery

DPS validation is invoked because of an error condition usually symptomatic of a DPS array out-of-sync condition. MIH-start-pending messages are the most common symptom. The system detects and DPS validation repairs the DPS arrays for only one device at a time.

Usually more than one device is affected, because the cause is typically related to:

- An action on the 3880 SD, such as an IML or interface disable/enable
- An action on the channel path, such as an IFRST

Depending on the device activity and the rate of RESERVE macros per volume, the system could take several minutes to several hours to repair the DPS arrays for all affected devices. While the arrays are out-of-sync, performance is impacted because of the loss of the dynamic path reconnect function. In addition, the installation is exposed to a failure or operator action on another path that might then result in undetected loss of reserves (a data integrity exposure) or boxing of the device.

It is therefore recommended that whenever a DPS array out-of-sync condition is detected for one device, as indicated by message IEA452I, the operator should enter a VARY PATH ONLINE command to all devices associated with the suspected component.

For example, if message IEA452I indicates that the DPS arrays for device 100 through 10F on channel path 22 have to be repaired, the operator would then enter a command to vary the path through CHP 22 for all associated devices that have a path already online.

```
VARY PATH((100-10F),22),ONLINE
```

DASD Maintenance and Recovery

DASD can experience failures such as defective disk surfaces, drives, and actuators. When these failures occur, data becomes inaccessible to the operating system and could be lost. To prevent the loss of the data, an installation should consider using the following to monitor possible error conditions and correct any before they cause outages.

- The Environmental Record Editing and Printing (EREP) System Exception Report
- Device Support Facilities (ICKDSF)

When a DASD error does occur, such as a defective track, an installation can use Data Facility Data Set Services (DFDSS) to retrieve the data from the defective areas and copy it to a back-up DASD.

Reference Books

See the following:

- *IBM Disk Storage Management Guide: Error Handling*
- *ICKDSF User's Guide*
- *EREP User's Guide*
- *DFDSS User's Guide*

Boxed Device - Operator Actions

Device *boxing* is used by the MVS system during error recovery as a means of maintaining data integrity and preventing data corruption. A device is also boxed if the operator issues the VARY devnum,OFFLINE,FORCE command. When a device is boxed, all outstanding I/O operations for the device are ended with *permanent error* status, and no new allocations to the device are allowed.

It is very important to understand that in the case of shared DASD, the boxed device is boxed only to the system that originated the boxing. The device is still accessible from other systems. **This may lead to incorrect (or incomplete) data on the DASD volume.** Such a situation must be reported to the owner of the data on the boxed-DASD.

- If the data-files are shared with other systems, it is recommended to put the device in offline status on all the sharing systems. Use VARY OFFLINE or OFFLINE,FORCE commands.
- After the data sets are checked and recovered, the DASD volume may be put back online.

A device that is boxed *and offline* can be brought back online with the VARY devnum,ONLINE command. This will enable the UCW and perform online processing to the device. Assuming that the error condition has been resolved, the device will come online. If the error condition still exists, the device may remain in the boxed state.

MVS/SP releases 3.1.0 and higher

A device that is *allocated boxed* may be brought back online with the VARY devnum,ONLINE,UNCOND command, if account procedures allow. Note that in this case, if the boxed device is DASD, volume verification (that is, VOLSER checking) is **not** performed. In this case, the VOLSER information can be obtained by entering a VARY devnum,ONLINE command to the DASD device or then entering a MOUNT command.

A DASD device that was offline (either boxed or not boxed) has the VOLSER details obtained from the device through the VARY devnum,ONLINE command. The VOLSER information is placed in the UCB as part of the vary online operation, if the vary online is successful, that is, that no out-of-line situations exist, for example, it is not a duplicate volume.

Recovery for a Failing Alias Unit Control Block (UCB)

For a parallel access volume, the status of a base UCB affects the status of its alias UCBs. The status of an alias UCB, however, might not affect the status of its base UCB. When the system detects a problem with an alias UCB, a recovery action applies to the base and its alias UCBs. If an alias UCB becomes boxed, you can unbox the alias with the following DEVSERV command:

```
DEVSERV QPAVS,devn,UNBOX
```

Error messages that display in the following situations are the only indication that an alias UCB is boxed:

- The device is varied online
- A hardware change is activated
- The system is in recovery

Boxed Tapes under Tape Management System Control

If a boxed tape drive is controlled by a tape management system, the drive will remain in the A-BOX state. Unless the tape management system is taken down, the VARY devnum,ONLINE,UNCOND command must be used to return the tape drive to the online state.

Tape Boxed Due to Lost Assign

If tape CHPIDs, control unit, switches, or ESCON connections are incorrectly handled, a tape ASSIGN may be reset (lost). When this occurs on the last path to the tape drive, the MVS system will box the device.

If an ASSIGN lost condition occurs while a tape was loaded, the MVS system may not be able to unload the tape. If this happens, as indicated by the cartridge remaining in the drive after the job has completed, perform the following actions at the tape unit:

- Place the READY/NOT READY switch to the NOT READY position.
- Toggle the UNLOAD switch, and the tape should unload.

Printer Boxed Due to Lost Assign

If printer CHPIDs, control unit, switches, or ESCON connections are incorrectly handled, a printer ASSIGN may be reset (lost). When this occurs on the last path to the printer, the MVS system will box the device.

- ## Issue 'D U,,devnum,1' to Determine Device Status



The following procedures are recommended for use when a boxed condition is reported.

Cause	Operator Action
I/O Error	Correct the cause of the I/O error condition and then attempt to bring the device online

Table 9-1. Correcting Boxed Conditions (continued)

Cause	Operator Action
Lost Reserve or Assign	For tape, if mount status=private, determine if any jobs were run or accesses made to the volume from any other system while it is in the boxed state. If yes, an integrity problem may exist and the device should not be varied online until the integrity of the volume can be assured. If no, you may attempt to un-BOX the device by varying it online. If the printer Assign was lost, it is possible the printer was assigned to another host. If so, the printer can be varied online to this host until it is varied offline from the other host. Otherwise, attempt to vary the printer online to this host.
Lost Last Path	Return the paths to an operational state and then vary the device online.
Subchannel Recovery	Identify and repair resource, then vary the device online
U/R 'Boxed'	Correct the cause of the I/O error condition and then attempt to bring the device online.
VARY FORCE command	Determine why the operator entered the VARY devnum,OFFLINE,FORCE command. Correct the condition on the system and vary the device back online.
Hot I/O	Identify and repair device, then vary the device back online.
Shared Tapes	Identify and repair device, then enter VARY devnum,ONLINE,UNCOND to unbox the device and bring it online

Additional Recovery Actions

This section includes these topics:

- Recovery by CPU restart
- Continuing a vector job if a Vector Facility is offline

Recovery by CPU Restart

The operator can initiate recovery from some system incidents, such as loops and uncoded wait states, by issuing a restart to the processor that has the problem. The restart reason that is entered as part of the restart process directs the system to perform one of two recovery actions:

Restart reason 0

The system tries to display message IEA500A on the first two consoles defined as master or alternate and locally attached to the system issuing the message. The message identifies the current unit of work on the target processor. The operator can reply either:

- RESUME to allow the current unit of work to continue
- ABEND to end that unit of work with an abend X'071'

If the operating system cannot communicate with the either console to issue message IEA500A, it ends the current unit of work with an abend X'071'.

Restart reason 1

The operating system:

- Interrupts the current unit of work on the target processor
- Detects and attempts to repair errors in critical system areas

- Writes a logrec data set error record for completion code X'071' with reason code 4 when repair actions were taken
- Reports the results of some of the actions taken in message IEA501I
- Returns control to the interrupted unit of work

Note: Restart of the CPU in a restartable wait state ignores the restart reason.

Reference Book

See *z/OS MVS System Commands* for information about restart.

Continuing a Vector Job If a Vector Facility is Offline

If a Vector Facility goes offline while a job is running on the Vector Facility, the job is redispached to another Vector Facility, if one is available. If, however, no other Vector Facility is available, the job is swapped out and message IRA700I is issued:

```
IRA700I jobname xxxxxxxx WAITING FOR AVAILABILITY OF VF
```

In this case, the operator has these choices:

- Enter CONFIG VF(x) to bring the Vector Facility back online. The job is then swapped in.
- Cancel the job.
- Do nothing, in which case the job may time out, depending on the system management facility (SMF) job wait time (JWT) specified, and be cancelled by the system.

With no Vector Facility online, other jobs that try to do a vector operation will be swapped out. If no Vector Facility is brought online and these jobs remain swapped out for the interval specified as the SMF JWT, the jobs will end with abend X'522'.

Prevent these time-outs of swapped-out vector jobs by:

- TIME=NOLIMIT on the JOB or EXEC statement
- An installation exit routine

Chapter 10. Recovery from APPC Problems

Advanced Program-to-Program Communications (APPC) attempts to restart automatically when it encounters an unrecoverable error. APPC contains two address spaces, APPC and the APPC/MVS transaction scheduler. Both address spaces attempt to restart independently of each other so that, if the APPC/MVS transaction scheduler abnormally ends, APPC can continue normal processing.

For both address spaces you must perform recovery actions for the following:

- Initialization problems
- Abnormal ending with restart
- Abnormal ending without restart

Recovery for the APPC Address Space

If APPC detects an unrecoverable error, it attempts to restart itself. The system prevents continually recurring restarts as follows: If APPC abnormally ends and attempts to restart twice in one hour, the system abnormally ends APPC rather than restarting it again.

APPC Initialization Problems

If APPC cannot be initialized, the system issues message ATB008E. In this case, APPC will not perform properly if the operator enters a START APPC command.

Recovery Actions for APPC Initialization Problems

- | |
|---|
| <ol style="list-style-type: none">1. Report the problem to the system programmer.2. When the problem is fixed, relPL the system. |
|---|

Abnormal Ending of APPC with Restart

If the system detects an unrecoverable error and abnormally ends APPC, and APPC automatically restarts, message ATB005I indicates that APPC is restarting. In addition, the following occurs:

- The cross-system coupling facility (XCF) notifies the members of APPC's XCF group that APPC is unavailable.
- The system shuts down all active logical units (LU).
- The system does not save or restore active APPC conversations, either local and remote. The conversations will be lost and the system will notify the invoking applications.
- The system will not read the APPCPMxx parmlib member during restart. The parmlib members that were in effect prior to the restart will still be in effect following the restart.
- The system deletes the cache of TP profiles and side information. Following restart, the system refills the cache during normal processing.
- If APPC component trace is active before APPC abnormally ends, the system dumps the trace records when APPC abnormally ends and the APPC trace will not be active following restart.
- The system does not save or restore information regarding schedulers when APPC abnormally ends and restarts.
- XCF notifies the members of the APPC's XCF group that APPC is available once it restarts.

Recovery Actions for APPC Abnormal Ending with Restart

1. Have applications redrive the system data file manager (SDFM) activate/deactivate TP profile interface to reset the TP profile information following the restart.
2. Restart APPC component trace and all APPC conversations that were active before the error when APPC services are again available.
3. Have schedulers re-identify themselves after APPC has restarted.

Abnormal Ending of APPC without Restart

If APPC ends due to an operator CANCEL or FORCE command or is unable to restart, as indicated by message ATB006I, the following occurs:

- XCF notifies the members of APPC's XCF group that APPC is unavailable.
- The system shuts down all active LUs.
- The system does not save or restore active APPC conversations, either local and remote. The conversations are lost and the system notifies the invoking applications.
- FMH-5 requests that have been received, but not passed to the appropriate scheduler, are sent an FMH-7 notifying the partner TP of the error.

Recovery Actions for APPC Abnormal Ending without Restart

1. Wait for the system to issue message ATB002I, indicating that APPC has ended.
2. Enter a START APPC command to restart APPC.

Recovery for the APPC/MVS Transaction Scheduler

Like the APPC address space, if the APPC/MVS transaction scheduler detects an unrecoverable error, it attempts to restart itself. The system prevents continually recurring restarts as follows: If the APPC/MVS transaction scheduler abnormally ends and attempts to restart twice in one hour, the system abnormally ends the APPC/MVS transaction scheduler rather than restarting it again.

Abnormal Ending of the APPC/MVS Transaction Scheduler with Restart

If the APPC/MVS transaction scheduler abnormally ends and automatically restarts, as indicated by message ASB050I, the following occurs:

- The system does not read the ASCHPMxx parmlib member during restart. The parmlib settings that were in effect prior to the restart are still in effect following the restart.
- The system marks all APPC initiators for deletion. The system deletes all waiting initiators immediately. The system deletes the initiators of active TPs as the TPs complete.

Recovery Actions for Abnormal Ending with Restart

No recovery actions are necessary. The APPC/MVS transaction scheduler will restart automatically.

Abnormal Ending of the APPC/MVS Transaction Scheduler without Restart

If the APPC/MVS transaction scheduler ends due to an operator CANCEL or FORCE command, or if it is unable to restart as indicated by message ASB051I, reason code X'00000001', the following occurs:

- The system marks all APPC initiators for deletion. The system deletes all waiting initiators immediately and deletes the initiators of active TPs as the TPs complete.

Recovery Actions for Abnormal Ending without Restart

1. Wait for the system to issue message ASB053I, indicating that the APPC/MVS transaction scheduler has completed its ending process.
2. Enter a START ASCH command to restart the APPC/MVS transaction scheduler.

Part 2. Reconfiguration

Chapter 11. Introduction to Reconfiguration

This part of the book covers reconfiguration of hardware units and partitions.

Reconfiguration

Reconfiguration is the process of adding hardware units to, or removing hardware units from, a configuration. Reconfiguration can be done physically with the hardware connections or logically in the operating system. The system can reconfigure itself, or the operator can reconfigure the system. Reconfiguration of hardware units can:

- **Configure a unit online:** This reconfiguration process adds hardware units to make them available for system processing.
- **Configure a unit offline:** This reconfiguration process removes hardware units from the system so that they cannot be used for system processing.

This book gives generalized actions for hardware reconfiguration of central processors (CPU) and any Integrated Cryptographic Features (ICRF) attached to the processors, Vector Facilities, central storage elements, expanded storage elements, channel paths, and I/O devices.

Partitions

A processor complex is the set of hardware units that support one or more operating systems. Depending on the hardware, a processor complex can have either or both of the following types of partitions:

Physical Partitions

The system resources of a partitionable processor complex can be configured into one or two partitions, which are called *sides*. A processor complex with one side is in single-image (SI) mode and with two sides is in physically partitioned (PP) mode. Each physical partition contains a completely separate operating system.

The reconfiguration for physical partitions consists of:

- Physical partitioning from single-image mode to physically partitioned mode
- Physical merging from physically partitioned mode to single-image mode

The reconfiguration can be done as follows:

- Statically, when the system is not processing. After a system has failed because of a hardware problem, an installation can use static partitioning for recovery. After partitioning, the operator initializes the system on the working side.
- Dynamically, when the system is processing. When a system is still running after a hardware problem, an installation can use dynamic partitioning for recovery.

Logical Partitions

The system resources of a processor complex with the Processor Resource/Systems Manager (PR/SM) feature can be configured into one or more logical partitions. A logical partition is a subset of hardware resources that supports an operating system. A logical partition consists of central storage, expanded storage, channel paths, central processors, and any ICRFs attached to the processors, and Vector Facilities. CPUs can be dedicated to a logical partition or can be shared by partitions. Each logical partition contains a

completely separate operating system. In a processor complex that is physically partitioned, each side can contain one or more logical partitions.

An installation would usually use PR/SM reconfiguration to change the system resources assigned to logical partitions to meet the work load changes during the day, week, or month. PR/SM reconfiguration can also be used for recovery.

This book gives generalized actions for the following reconfigurations:

- Dynamic partitioning and merging of physical partitions
- Reconfiguration under PR/SM

RSU Parameter for Partitioning

For reconfiguration of physical or logical partitions, the RSU parameter specified in the IEASYSxx parmlib member or at system initialization must define the reserved central storage in the partition so that a second physical partition or logical partition can be placed in that storage.

By default, the RSU parameter is 0, so that the system does not attempt to keep any storage reconfigurable. To define an RSU parameter for a system to be reconfigured, first identify the kind of partition in which the system will run:

- For a physical partition, the storage increment is:
 - 2 megabytes for a ES/3090 Processor Complex Model 400
 - 4 megabytes for any other physically partitionable 3090 or ES/9000.
- For a logical partition, the size of the storage increments varies, depending on the processor in use. For example, while 1 megabyte is the correct storage increment for many processors, 4 megabytes is the correct increment for 711-based ES/9021 processors. To determine the correct central storage increment for your specific processor level, see *PR/SM Planning Guide*.

The following chapters explain the RSU parameter in more detail for the two kinds of partitions.

Chapter 12. Input/Output Configuration

Before placing a system into operation, an installation should plan the input/output (I/O) device configuration for maximum availability and to provide the maximum capability for reconfiguration. I/O configuration planning concerns the number of paths to each device and the hardware elements in each path.

MULTIPLE PATHS TO A DEVICE SHOULD INCLUDE AS FEW COMMON HARDWARE ELEMENTS AS POSSIBLE TO MINIMIZE THE EFFECT OF A MALFUNCTION, THAT IS, TO PREVENT A SINGLE MALFUNCTION FROM DISABLING ALL THE PATHS TO A DEVICE.

Multiple paths to some devices may require the installation of two-channel switches or switching devices, such as a 3814 Switching Management System.

The illustrations in this chapter show I/O configurations that will give maximum availability if a hardware element fails. Each illustration shows the hardware elements in the path to a device and shows how the device should be connected.

Note: Specific details about I/O configurations and channel paths vary among processor complex models. (See *Channel Characteristics* and *Functional Characteristics*).

This chapter covers the following:

- Channel subsystem configurations
- I/O configuration for a non-partitionable processor complex
- I/O configuration for a partitionable processor in single-image mode
- Configuration for a master console
- I/O configuration program (IOCP) considerations
- Hardware configuration definition (HCD) considerations

Channel Subsystem Configurations

The hardware elements in a path from a processor complex to a device are:

- Channel control element (CCE)
- Channel element (CHE)
- Channel path (CHP)
- Control unit

Figure 12-1 shows the relation among the hardware elements that make up the channel subsystem.

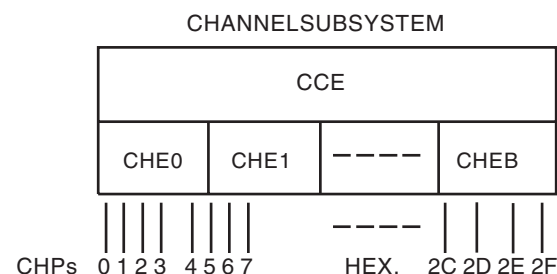


Figure 12-1. ES/3090 Processor Complex Channel Subsystem Configuration

Channel Designations on Non-partitionable 3090 Models

The non-partitionable models of the 3090, for example, models 200, 200E, and 300E, have 32 standard CHPs numbered 00 - 1F. The model 200 can have 8 or 16 additional CHPs, numbered 20 - 2F, and the models 200E and 300E can have up to 32 additional CHPs, numbered 20 - 3F. Note that unlike the 308x CHP identifiers (CHPIDs), the CHPs on the non-partitionable models of the 3090 are consecutively numbered with no gaps in the numbering.

A channel element (CHE) has four consecutively numbered channel paths. The CHEs numbered 0 - 7 relate to the 32 standard CHPs on all three non-partitionable models. For the optional CHPs, the CHEs are numbered 8 - B on the model 200, and 8 - F on the models 200E and 300E. Table 12-1 summarizes CHP and CHE numbering for several 3090 models.

Channel Designations on Partitionable 3090 Models

Table 12-1 shows the channel path (CHP) designations and the channel element (CHE) designations for various 3090 models.

Table 12-1. Channel Paths and Channel Elements on Partitionable 3090 Processor Complexes

Model Number	Model 280E	Model 400	Model 400E	Model 500E	Model 600E
Maximum Number of CHPs	64	96	128	128	128
CHPIDs - Side 0	0 - 1F	0 - 2F	0 - 3F	0 - 3F	0 - 3F
CHPIDs - Side 1	40 - 5F	40 - 6F	40 - 7F	40 - 7F	40 - 7F
Maximum Number of CHEs	16	24	32	32	32
CHE IDs - Side 0	0 - 7	0 - B	0 - F	0 - F	0 - F
CHE IDs - Side 1	10 - 17	10 - 1B	10 - 1F	10 - 1F	10 - 1F

Configuration Guidelines

When you configure a device with multiple paths to the same system, the following guidelines minimize single points of failure to allow for maximum availability of the device:

- Attach each path from the device to a separate CHE.
- Distribute paths across both odd and even numbered CHEs. Thus, one path from the device could include any of these channel paths: 0 - 3, 8 - B, 10 - 13, and 18 - 1B. Assuming two paths for wash device, the other path could be selected from channel paths 4 - 7, C - F, 14 - 17, or 1C - 1F.
- If optional channel paths (20 - 2F) are installed, distribute the paths to a device across both optional and standard channel paths. When doing this, also follow the configuration pattern of odd and even CHEs.
- For 3090 Models 280E, 400, 400E, 500E, and 600E, distribute the paths across channel control elements (CCEs).

Channel Subsystem Considerations for an ES/4381 Processor Model 92E

The hardware elements in the path from a 4381 dual processor to a device are:

- Channel path (CHP)
- Control unit

Figure 12-2 shows the relation among the hardware elements that make up the 4381 dual-processor channel subsystem.

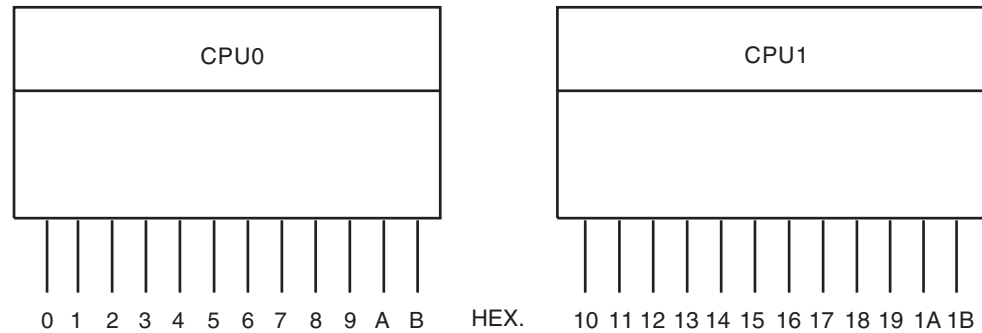


Figure 12-2. 4381 Dual-Processor Channel Subsystem Configuration

Note: The model 92E has up to 24 channels, 12 on each processor.

Where possible, I/O devices should be connected to channel paths on both central processors (CPU) of the processor complex. For recovery, the system DASD devices should be accessible from either CPU and at least one MVS operator console should be attached to each CPU. This configuration is necessary if a CPU failed, because the associated channel paths would become unusable.

The system console for 4381 dual stream processors is connected through a local channel adapter only to channel path 0 on CPU 0. It is desirable to have at least one operator console device attached through a local 3274 Control Unit on a channel path belonging to CPU 1. This configuration would still allow the operator to communicate with MVS if either CPU 0 or channel path 0 were to fail.

I/O Configuration for Non-Partitionable Processor Complex

The next four figures illustrate how the connection of a device control unit to more than one channel element (CHE) in the channel subsystem decreases the risk of channel path loss if one CHE were to fail. With such failure, the control unit would still have access to the other CHE. Although these figures illustrate connection to the 3090, the general concepts apply to all processor types supported by MVS/ESA.

The figures show the following configurations:

- Figure 12-3 on page 12-4: DASD
- Figure 12-4 on page 12-5: Tape
- Figure 12-5 on page 12-6: 3725 Communication Controller
- Figure 12-6 on page 12-7: Unit record or local display device

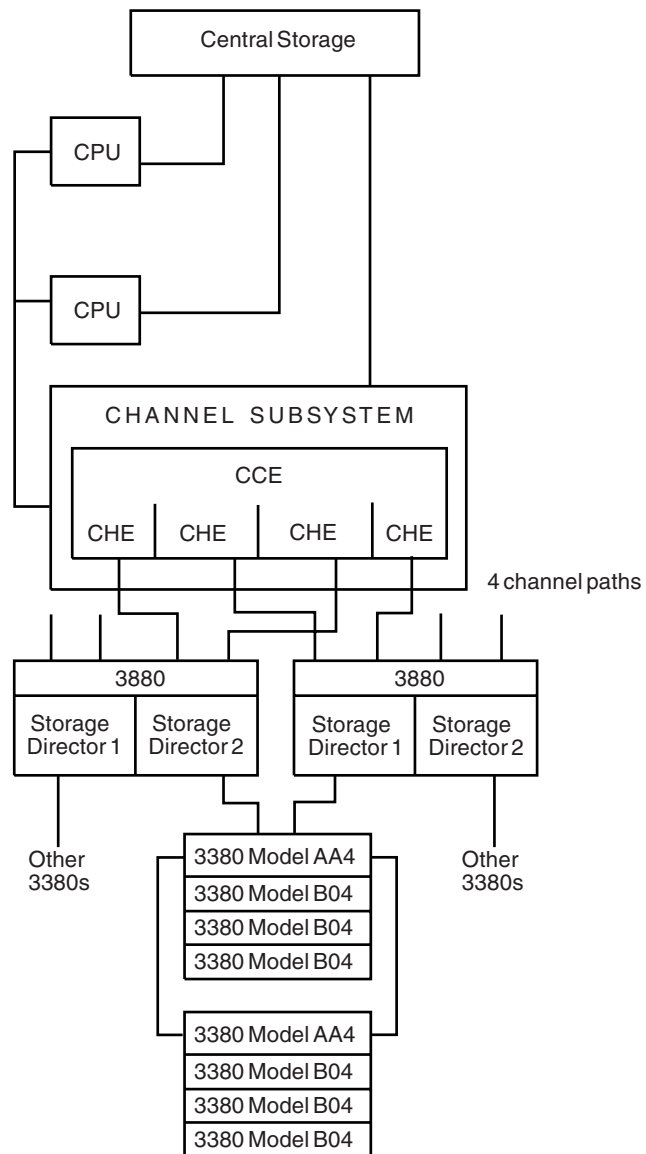


Figure 12-3. DASD Configuration for Maximum Availability with a 3090 Complex

String switching or a 3380 Model AA4 is chosen so all devices can be accessed, even if a control unit fails.

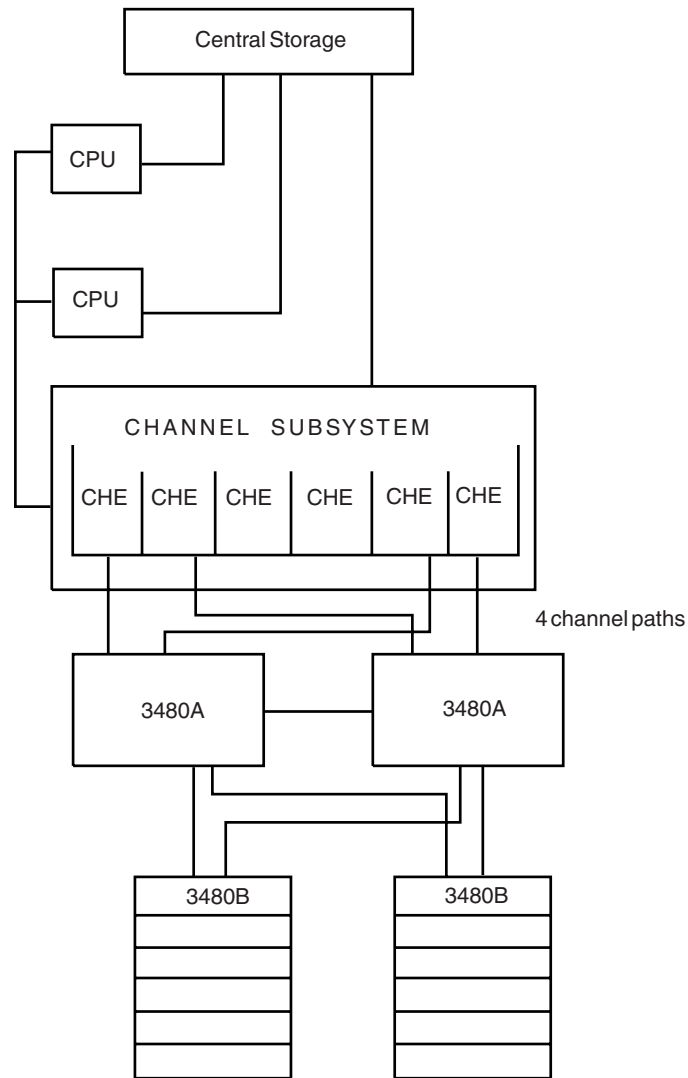


Figure 12-4. Tape Configuration for Maximum Availability (3090 Complex)

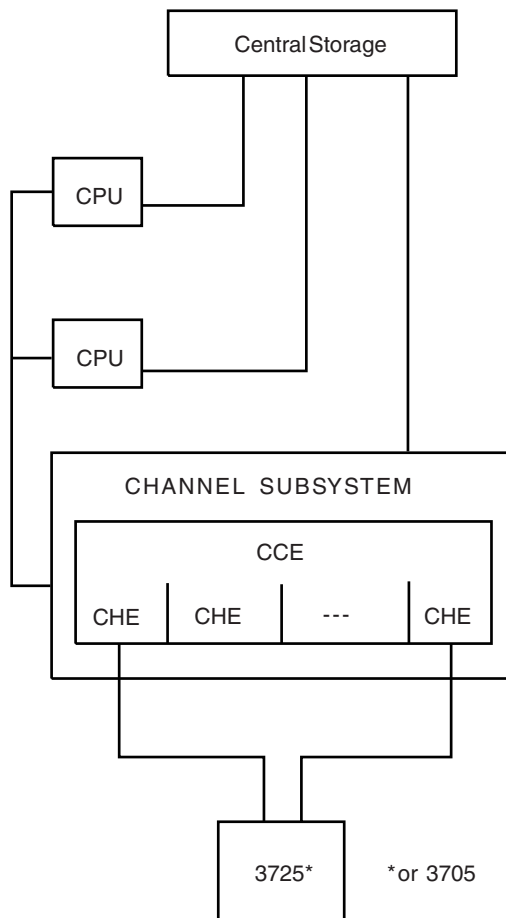


Figure 12-5. Configuration of a 3725 Communication Controller for Maximum Availability (3090 Complex)

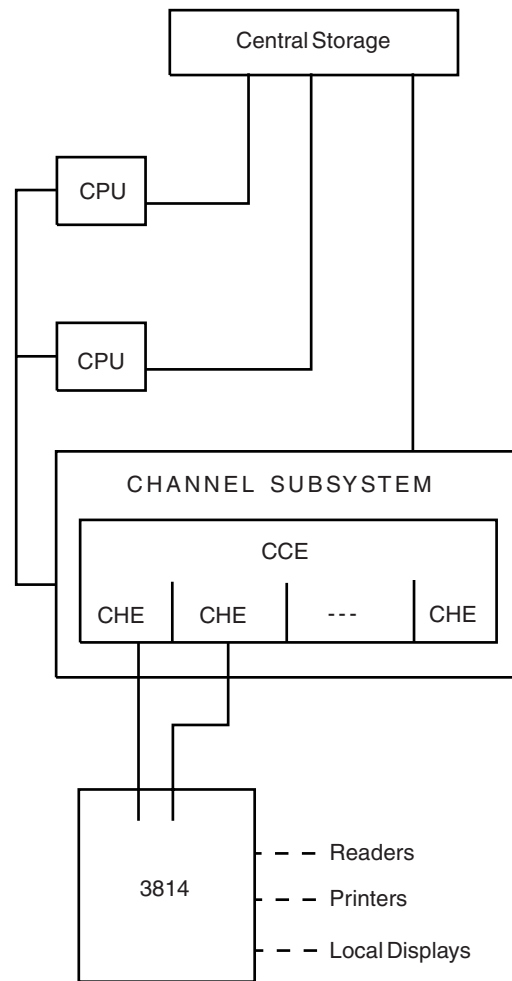


Figure 12-6. Unit Record or Local Display Device Configuration for Maximum Availability (3090 Complex)

I/O Configuration for Partitionable Processor Complex in Single-Image Mode

When an installation plans its I/O configuration for a partitionable 3090 Processor Complexes in single-image mode, the plan should address not only that mode, but physically partitioned mode as well. The resulting I/O configuration should provide maximum availability in either mode of operation.

Some general recommendations for an I/O configuration are:

- Attach control units symmetrically, whenever possible, so they can be accessed from both sides.
- Attach critical unit record or local teleprocessing (TP) device control units through a 3814 or 2914 switch, so they can be switched to either side.
- Attach a tape or DASD device to one channel path from each side and operate with two channel paths online. To provide the same availability in physically partitioned mode, attach the device to four channel paths, two from each side.

Figure 12-7 illustrates a configuration for single-image mode for maximum availability. The example shows a DASD configuration and a partitionable 3090. However, tape, TP-controller, and unit record configurations on a 3090 and all device configurations should be done similarly, using features like two-channel switches and switching units, to attach the devices to different channel paths on different sides.

String switching features are used so that all devices can be accessed if a control unit fails.

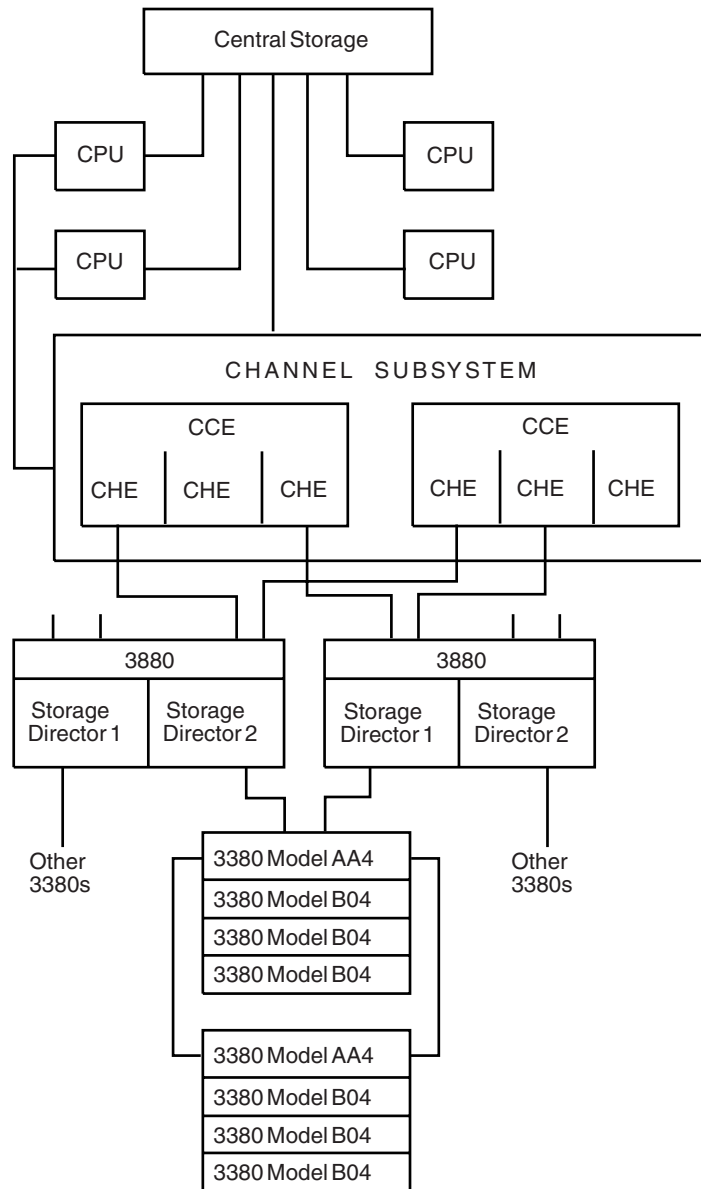


Figure 12-7. DASD Configuration for Maximum Availability (3090 in Single-Image Mode)

Configuration for Master Console

When attaching MCS consoles to a complex, use the following guidelines. Following these guidelines provides a high degree of access to the consoles and helps increase the availability of the system:

- Dedicate one control unit to the master console and a different one to its alternate.
- Ensure that, in a string of control units on the channel path, the control unit for the master or alternate console is the first terminal control unit on the channel path. Also, ensure that the control unit is set for *high* priority; this setting is performed by the hardware support representative.
- Attach the master console and its alternate so that they share the least number of common hardware elements. For systems that can be physically partitioned, attach the master and alternate consoles to channel paths on different sides.

When using SMCS consoles, attempt to eliminate as many single points of failure as possible to eliminate the possibility of losing access to all of the SMCS consoles in a sysplex. This includes not having all SMCS consoles in the sysplex active on only one system within the sysplex.

If the master console and its alternate are configured in this way, the probability of the operator receiving a critical error message and being able to respond prior to operator response timeout, approximately two minutes, is very high. If, however, the master and alternate consoles are not on dedicated control units, the ability of the operator to communicate with the operating system in certain recovery situations is impacted.

During system recovery processing for situations, such as hot I/O, the system uses synchronous WTO/WTOR macros to communicate with the operator. The synchronous WTO and WTOR macros will cause messages to be displayed to consoles defined in the SYNCHDEST console group, or if one is not defined, the master console. For more information on SYNCHDEST, see the *z/OS MVS Planning: Operations* or *z/OS MVS Initialization and Tuning Reference* books. Synchronous WTO/WTOR messages cannot be displayed on SMCS consoles, even if a SMCS console is the master console. If the message cannot be issued to the master console or any consoles in the SYNCHDEST group, or if the message is displayed on any of those consoles but the operator does not respond within the time limit, then the system will attempt to display the message on the system console. If the message cannot be issued to the system console, either the entire system or one CPU, depending on the problem, is put into a restartable wait state. To recover from the wait state, the operator must use recovery procedures that might require modification of central storage. Providing the master console with its own dedicated control unit significantly reduces the chances of encountering a restartable wait state as a result of failed processing of a branch entry WTO/WTOR macro, and forcing the operator to use a complex recovery procedure.

The following sequence shows how a problem can occur when another device is attached to the same control unit as the master console:

1. The operating system sends the message.
2. Before the operator can respond to the message, another device attached to the control unit is accessed.
3. The operator tries to respond to the message, but the control unit is busy.
4. The control unit remains busy longer than operator response timeout.
5. The operating system tries to communicate through the system console.
6. If it cannot communicate, the operating system loads a restartable wait state that indicates the error condition to which the operator is to respond.

Processing Messages at the System Console

When the operating system successfully writes a branch entry WTOR message to the system console, the alarm sounds and the existing screen image on the system console is replaced with an indication that an SCP message is pending. The operator displays the message by entering F SCPMSG on a 3090.

The SCP command line will contain the characters

R 00,

with the cursor positioned after the comma. Enter the reply.

The reply to a branch entry WTOR message on the system console does not have a timeout interval. The message remains pending on the system message facility screen image until the operator replies to it.

IOCP Considerations

When operating in single-image mode, the *dual write* function allows an installation to define duplicate copies of a new I/O configuration on each partition with a single running of the I/O control program (IOCP). The installation can then physically partition the processor complex and have the new I/O configuration available to either partition. In this way, for example, the installation can use either partition to test the new I/O configuration, or act as a common back-up.

MVS Hardware Configuration Definition Program Considerations

HCD provides a single point of control for defining a hardware configuration. It supports 4 digit device numbers and provides an interactive interface that allows you the ability to perform the following tasks online:

- Define new configuration data.
- View and modify existing configuration data.
- Activate configuration data.
- Maintain IODFs.
- Query and print configuration data.
- Migrate existing configuration data.

FEATURE=SHAREDUP is an optional parameter on the Add Device panel in HCD. Use this parameter to:

- Eliminate the overhead of the hardware device reserve/release logic when a device is attached only to partitions operating in single-image mode.
- Indicate that the device reserve/release logic is to be used only when operating in physically partitioned mode.
- Allow the sharing of a device between physical partitions.

Note: Specify FEATURE=SHARED (not FEATURE=SHAREDUP) if a device is attached to more than one processor complex. (See *z/OS HCD User's Guide*.)

FEATURE=ALTCTRL is an optional parameter on the HCD Add Device panel. Use this parameter to allow a device to be accessed through an alternate control unit. (See *z/OS HCD User's Guide*.)

Chapter 13. Reconfiguration

Reconfiguration is the process of adding hardware units to, or removing hardware units from, a configuration. Units can be either:

- **Online:** Units in use by a system are called *online*. When both physically and logically online, a unit is available to be used by the system.
- **Offline:** Units not in use by a system are called *offline*. When either physically or logically offline, a unit is not available to be used by the system.

An installation can use reconfiguration to:

- Adapt a system to changing work loads by configuring units online or offline as required.
- Perform maintenance on a part of a complex while the other part continues normal operation.
- Possibly recover system operation by configuring failing units offline.

Hardware unit or units may be put offline before initialization of the system or systems. To do this, an operator can deselect through the system console such units as central processors (CPU), storage elements, or channel paths. Note that an operator should never deselect a unit during system operation, because the operating system is not notified of the removal.

During system operation, the operating system configures failing units offline with or without any intervention. When intervention is needed, the operator:

- Can enter a CONFIG command that identifies the CONFIGxx member of Parmlib that specifies the reconfiguration. (See *z/OS MVS Initialization and Tuning Reference*.)
- Can enter a CONFIG or VARY command that directly configures units online or offline. (See *z/OS MVS System Commands*.)

For maintenance of hardware, prepare reconfiguration actions to configure hardware units offline for repair and back online so the system can use them. Also, hardware units and channel paths are reconfigured offline then online during reconfiguration of partitionable systems.

Dynamic I/O Configuration

Dynamic I/O configuration lets you change your I/O configuration without causing a system outage. In other words, you can select a new I/O configuration definition without performing a power-on-reset (POR) of the hardware or an initial program load (IPL) of the MVS system. Using I/O definition files (IODFs) created through hardware configuration definition (HCD), dynamic I/O configuration allows you to add, delete, or modify the definitions of channel paths, control units, and I/O devices to the software and hardware I/O configurations. You can change the I/O configuration definitions to both software and hardware or software only. (See *z/OS HCD Planning*.)

Dynamic I/O configuration has the following benefits:

- Increases system availability by allowing you to change the I/O configuration while MVS is running, thus eliminating the POR and IPL for selecting a new or changed I/O configuration definition.

- Allows you to make I/O configuration changes when your installation needs them rather than wait for a scheduled outage to make the changes.
- Minimizes the need to over-define an I/O configuration by logically defining hardware devices that do not physically exist.

Logical and Physical Reconfiguration

Logical reconfiguration allows or prevents use of a resource by the operating system. Physical reconfiguration allows or prevents use of a resource by the hardware. An operator can enter a CONFIG command on the console with master authority to logically and physically reconfigure the following hardware units, if applicable for the particular processor complex:

- CPUs
- Vector Facilities
- Central storage elements
- Expanded storage elements
- Central storage increments
- Channel paths
- I/O devices

Also, when an operator issues a CONFIG CPU command to reconfigure a processor, the system logically reconfigures any Integrated Cryptographic Feature (ICRF) attached to the processor.

Note: Physical reconfiguration may not be supported for all hardware units by all processor models. (See *Functional Characteristics*).

Reconfiguration Support According to Processor Types

The reconfiguration functions supported by MVS depend on the configuration of the processor complex, as follows.

Uniprocessor (UP)

Depending on the processor type, an installation can configure offline some or all of the following:

- Vector Facilities
- Central storage elements
- Expanded storage elements
- Central storage increments
- Channel paths
- I/O devices

In a UP system, the purpose of reconfiguration is to configure offline failing units to allow the system to continue operation.

Multiprocessor (MP)

Depending on the processor type, an installation can configure offline some or all of the following:

- Central processors, including any associated ICRFs
- Vector Facilities
- Central storage elements
- Expanded storage elements
- Central storage increments
- Channel paths
- I/O devices

Reconfiguring a Central Processor

When configuring a central processor offline, the operating system stops dispatching work to the processor and takes it logically offline. Then the system stops the processor removes it physically from the configuration, if physical central processor reconfiguration is supported by the machine.

Actions to Reconfigure a Central Processor Offline

1. Enter a CONFIG CPU(x),OFFLINE command on a console with master authority. The system responds on the console on which the command was entered.
The operating system rejects a CONFIG CPU(x),OFFLINE command when:
 - The target processor is the only online processor.
 - The target processor is the only processor with an operative timer.
 - Alternate CPU recovery (ACR) occurs during OFFLINE processing.
 - Any active jobs have processor (CPU) affinity with the target processor.
2. If you enter a CONFIG command for a central processor and currently scheduled jobs have CPU affinity for that processor, the system issues message IEE718I to list the jobs. Do the following:
 - a. Prevent the operating system from scheduling any additional jobs, by replying YES to message IEE718D.
 - b. Either wait for the active jobs to complete or cancel them.
Note that replying YES to message IEE718D leaves the target central processor unavailable for affinity job scheduling. If you want to restore the central processor to its original state, enter CONFIG CPU(x),ONLINE to restore the original central processor status and make the target central processor available for affinity job scheduling.
 - c. Reenter the CONFIG CPU(x),OFFLINE command.

Reconfiguring a Central Processor with an ICRF

The operator cannot directly reconfigure an ICRF. When the operator uses the CONFIG command to reconfigure a central processor with an associated ICRF, the system changes the online/offline status of both the central processor and the associated ICRF as follows:

- When the operator enters a CONFIG CPU(x), ONLINE and Integrated Cryptographic Service Facility/MVS (ICSF/MVS) is active in the processor complex, the system brings the ICRF online.
- When the operator enters a CONFIG CPU(x), OFFLINE command the system takes the ICRF offline.
- When the operator enters a CONFIG CPU(x), ONLINE and ICSF/MVS is not active, the ICRF is not brought online until ICSF/MVS is started.

Actions to Bring Online a Central Processor and its ICRF

Central processor x (CPU x) is to be brought online. The processor has an associated ICRF and ICSF/MVS is installed and active in the processor complex.

Enter one of the following commands on a console with master authority; the system responds on the console on which the command was entered:

```
CONFIG CPU(x)
```

```
CONFIG CPU(x), ONLINE
```

When the command completes, the system issues the following messages:

```
IEE504I CPU(x), ONLINE  
IEE504I CRYPTO(x), ONLINE
```

Removing the Last ICRF

When a CONFIG command is entered to remove a central processor associated with the last online ICRF, the system issues the following messages:

```
IEE109I CONFIG CPU(x), OFFLINE COMMAND WOULD REMOVE LAST CRYPTO  
IEE325D REPLY U TO CONTINUE CONFIG COMMAND. REPLY C TO CANCEL
```

After *U* is replied to message IEE325D, the system takes the central processor and the ICRF offline.

From that point on, the system abnormally ends any jobs that request cryptographic services using ICSF/MVS. This applies to new jobs and to jobs running in the processor complex at the time the system took the ICRF offline.

Actions to Take Offline a Central Processor and its ICRF

Central processor x is to be taken offline. The processor has an associated ICRF.

Enter the following command on a console with master authority; the system responds on the console on which the command was entered:

```
CONFIG CPU(x), OFFLINE
```

When the command completes, the system issues the following messages:

```
IEE505I CPU(x), OFFLINE  
IEE505I CRYPTO(x), OFFLINE
```

Reconfiguring Vector Facilities

If a processor complex has an associated Vector Facility, this processing occurs:

- When CONFIG CPU(x),OFFLINE is issued, the system takes any Vector Facility associated with CPU x offline.
- When the processor is brought back online through a CONFIG CPU(x) or CONFIG CPU(x),ONLINE command without VFON or VFOFF, the system will configure an associated Vector Facility as it had been when the processor was taken offline. The Vector Facility will be physically and logically online or offline as before.

Actions to Reconfigure a Vector Facility

1. To take a Vector Facility offline for repair or physical maintenance, do the following:
 - Take the side offline for a partitionable model
 - Shut down the entire system for a non-partitionable model

The *x* in the CONFIG VF(*x*) command should be the same as that of the associated CPU.

2. After the repair, do one of the following:
 - a. Enter a CONFIG CPU(*x*),ONLINE,VFON command, to bring online first the CPU, then its associated Vector Facility.
 - b. Enter two commands, in this order:

```
CONFIG CPU(x),ONLINE
CONFIG VF(x),ONLINE
```

Enter the CONFIG commands on a console with master authority. The system responds on the console on which the command was entered.

Removing the Last Vector Facility

If a CONFIG command asks the system to remove the last Vector Facility and vector jobs are scheduled, the system issues the following messages to the console with master authority:

```
IEE176I CONFIG {CPU(x)|VF(x)},OFFLINE COMMAND WOULD
        REMOVE LAST VF, dd VF JOBS SCHEDULED. JOBNAMES ARE:  jobname,
        [jobname...]
IEE177D REPLY 'U' TO SUSPEND VF JOBS. REPLY 'C' TO CANCEL CONFIG
        COMMAND
```

For a reply of *U*, the system puts the vector jobs into a vector wait. If the jobs specified TIME=NOLIMIT on their JCL JOB or EXEC statements, they will not time out and will not be cancelled. The following message will appear for each vector job in a vector wait:

```
IRA700I jobname  WAITING FOR AVAILABILITY OF VF
```

Later, when a Vector Facility is brought online, the vector jobs will continue on that Vector Facility.

Actions to Bring Online a Central Processor and its Vector Facility

CPU x is to be brought online physically and logically. If CPU x has a Vector Facility, the logical and physical status of its Vector Facility (online or offline) is to be the same as it was when the CPU was last online. That is, if the Vector Facility was offline, it is still offline. If it was online, it is still online.

Enter one of the following commands on a console with master authority; the system responds on the console on which the command was entered:

```
CONFIG CPU(x)
CONFIG CPU(x),ONLINE
CONFIG CPU(x),ONLINE,VFON
```

When this command completes, the system issues the following messages if the Vector Facility comes online:

```
IEE504I CPU(x), ONLINE
IEE504I VF(x), ONLINE
```

If, however, the Vector Facility does not come online or the processor has no Vector Facility, the system issues one or more messages, which may include the following:

```
IEE504I CPU(x), ONLINE
IEE239I VF(x) NOT RECONFIGURED ONLINE - VF IS POWERED OFF
IEE241I VF(x) NOT RECONFIGURED ONLINE - VF NOT AVAILABLE
IEE241I VF(x) NOT RECONFIGURED ONLINE - CPU HAS NO VF
```

Actions to Bring Online a Central Processor but Not its Vector Facility

CPU x is to be brought logically and physically online. Its Vector Facility is to be logically and physically offline.

Enter the following command on a console with master authority; the system responds on the console on which the command was entered:

```
CONFIG CPU(x),ONLINE,VFOFF
```

When this command completes, the system issues the following messages:

```
IEE504I CPU(x), ONLINE
IEE505I VF(x), OFFLINE
```

If, however, the processor has no Vector Facility, the system issues the first message and either the second or the third:

```
IEE504I CPU(x), ONLINE
IEE241I VF(x) NOT RECONFIGURED OFFLINE - VF NOT AVAILABLE
IEE241I VF(x) NOT RECONFIGURED OFFLINE - CPU HAS NO VF
```

Actions to Take Offline a Central Processor and its Vector Facility

CPU x and its Vector Facility are to be taken offline physically and logically.

Enter the following command on a console with master authority; the system responds on the console on which the command was entered:

```
CONFIG CPU(x),OFFLINE
```

When the command completes, the system issues the following messages:

```
IEE505I CPU(x), OFFLINE  
IEE505 VF(x), OFFLINE
```

Actions to Bring Online a Vector Facility for an Online Central Processor

The Vector Facility for CPU x is to be brought logically and physically online. Its central processor is already logically and physically online.

Enter one of the following commands on a console with master authority; the system responds on the console on which the command was entered:

```
CONFIG VF(x)  
CONFIG VF(x),ONLINE
```

When the Vector Facility is back online, the system issues the following message:

```
IEE504I VF(x), ONLINE
```

If, however, CPU x is offline, the system issues one of the following messages instead:

```
IEE239I VF(x) NOT RECONFIGURED ONLINE – CPU(x) NOT ONLINE  
IEE241I VF(x) NOT RECONFIGURED ONLINE – CPU NOT ONLINE
```

If CPU x is online but does not have a Vector Facility, the system issues this message:

```
IEE241I VF(x) NOT RECONFIGURED ONLINE – CPU HAS NO VF
```

Actions to Take Offline a Vector Facility When its Central Processor Remains Online

The Vector Facility for CPU x is to be taken logically and physically offline. CPU x is to remain logically and physically online.

Perform this action to take a Vector Facility offline after a re-IPL, if the Vector Facility has repeatedly caused errors before the re-IPL.

Enter the following command on a console with master authority; the system responds on the console on which the command was entered:

```
CONFIG VF(x),OFFLINE
```

When the Vector Facility goes offline, the system issues the following message:

```
IEE505I VF(x), OFFLINE
```

If, however, CPU x is offline, the system issues this message instead:

```
IEE239I VF(x) NOT RECONFIGURED PHYSICALLY OFFLINE – CPU(x) NOT ONLINE  
IEE241I VF(x) NOT RECONFIGURED OFFLINE – CPU NOT ONLINE
```

If CPU x is online but does not have a Vector Facility, the system issues this message:

```
IEE241I VF(x) NOT RECONFIGURED OFFLINE – CPU HAS NO VF
```

Reconfiguring Central and Expanded Storage

To place storage offline and then bring it online to be used again, commands can reconfigure central storage increments, central storage elements, and expanded storage elements.

Storage may be reconfigured for physical partitioning or merging. Under PR/SM, storage may be reconfigured between logical partitions. See Chapter 15, “Reconfiguration under PR/SM,” on page 15-1, for a description of storage reconfiguration between logical partitions.

Physical View of Central Storage

Central storage is physically divided into storage elements (SE). The central storage elements are composed of storage increments. Each storage increment may be composed of two sub-increments:

- One sub-increments contains the even-numbered frames of the increment, such as, 0-kilobyte, 8-kilobyte, 16-kilobyte, and so on.
- The other contains the odd-numbered frames, such as, 4-kilobyte, 12-kilobyte, 20-kilobyte and so on.

The sub-increments of an increment may reside in different storage elements, as follows:

SE0	SE2	SE1	SE3
30-32M Even Frames Reconfigurable	30-32M Odd Frames Reconfigurable	62-64M Even Frames HSA and Preferred	62-64M Odd Frames HSA and Preferred
28-30M Even Frames Preferred	28-30M Odd Frames Preferred	60-62M Even Frames Reconfigurable	60-62M Odd Frames Reconfigurable
26-28M Even Frames Preferred	26-28M Odd Frames Preferred	58-60M Even Frames Reconfigurable	58-60M Odd Frames Reconfigurable
24-26M Even Frames Preferred	24-26M Odd Frames Preferred	56-58M Even Frames Reconfigurable	56-58M Odd Frames Reconfigurable
·	·	·	·
·	·	·	·
·	·	·	·
6-8M Even Frames SQA and Preferred	6-8M Odd Frames Preferred	38-40M Even Frames Reconfigurable	38-40M Odd Frames Reconfigurable
4-6M Even Frames Preferred	4-6M Odd Frames Preferred	36-38M Even Frames Reconfigurable	36-38M Odd Frames Reconfigurable
2-4M Even Frames Preferred	2-4M Odd Frames Preferred	34-36M Even Frames Reconfigurable	34-36M Odd Frames Reconfigurable
0-2M Even Frames V=R and Preferred	0-2M Odd Frames V=R and Preferred	32-34M Even Frames Reconfigurable	32-34M Odd Frames Reconfigurable

Storage Increments and Sub-increment Sizes

The following central storage increments cannot be configured offline:

- The increment containing absolute address 0
- The highest addressable increment available at IPL
- Any increment containing preferred central storage frames

A storage element can be configured offline only if:

- It contains only non-preferred storage frames
- The preferred storage sub-increments in this storage element can be moved to another storage element containing reconfigurable storage sub-increments.

Processor Complex	Storage Element ID	Storage Element Size (megabytes)	Increment Size (megabytes)	Sub-increment Size (megabytes)	Maximum Storage Size (megabytes)
ES/3090 Model 110J	1	32	2	1	64
ES/3090 Model 100S	1	32	2	1	64
ES/3090 Model 120J	1	32	4	2	64
ES/3090 Model 120S	1	32	2	1	64
ES/3090 Model 150J	1	32	4	2	64
ES/3090 Model 150JH	1	32 - 64, then 128	4	2	128
ES/3090 Model 150S	1	32	2	1	64
ES/3090 Model 170J	1	32 - 64, then 128	4	2	128
ES/3090 Model 170JH	1	32 - 64, then 128	4	2	128
ES/3090 Model 170S	1	32	2	1	64
ES/3090 Model 180J	1	32 - 64, then 128	4	2	128
ES/3090 Model 180S	1	32 - 64, then 128	4	2	128
ES/3090 Model 200	0, 1	32	2	1	64
ES/3090 Model 200E	0, 1	32, 64	4	2	128
ES/3090 Model 200J	0, 1	64 - 128; then 128 - 256	4	2	256
ES/3090 Model 200S	0, 1	64 - 128; then 128 - 256	4	2	256
ES/3090 Model 250J	1, 3	32	4	2	128
ES/3090 Model 250JH	1, 3	On each side: 32 - 64, then 64 - 128	4	2	256
ES/3090 Model 250S	1, 3	64	4	2	128
ES/3090 Model 280E	0, 1, 2, 3	16, 32	4	2	128
ES/3090 Model 280J	1, 3	On each side: 32 - 64, then 64 - 128	4	2	256
ES/3090 Model 280S	1, 3	64 - 128, then 128 - 256	4	2	256
ES/3090 Model 300E	0, 1	32, 64	4	2	128
ES/3090 Model 300J	0, 1	64 - 128, then 128 - 256	4	2	256
ES/3090 Model 300S	0, 1	64 - 128, then 128 - 256	4	2	256
ES/3090 Model 380J	0, 1, 3	On each side: 64 - 128, then 128 - 256	4	2	512
ES/3090 Model 380S	0, 1, 3	128 - 256, then 256 - 512	4	2	512
ES/3090 Model 400	0, 1, 2, 3	32	2	1	128
ES/3090 Model 400E	0, 1, 2, 3	32, 64	4	2	256
ES/3090 Model 400J	0, 1, 2, 3	On each side: 64 - 128, then 128 - 256	4	2	512

Processor Complex	Storage Element ID	Storage Element Size (megabytes)	Increment Size (megabytes)	Sub-increment Size (megabytes)	Maximum Storage Size (megabytes)
ES/3090 Model 400S	0, 1, 2, 3	128 - 256, then 256 - 512	4	2	512
ES/3090 Model 500E	0, 1, 2, 3	32, 64	4	2	256
ES/3090 Model 500J	0, 1, 2, 3	On each side: 64 - 128, then 128 - 256	4	2	512
ES/3090 Model 500S	0, 1, 2, 3	128 - 256, then 256 - 512	4	2	512
ES/3090 Model 600E	0, 1, 2, 3	32, 64	4	2	256
ES/3090 Model 600J	0, 1, 2, 3	On each side: 64 - 128, then 128 - 256	4	2	512
ES/3090 Model 600S	0, 1, 2, 3	128 - 256, then 256 - 512	4	2	512
ES/9000 Model 620	0, 1, 2, 3	128, 192, 256, 384, 512	4	2	512
ES/9000 Model 720	0, 1, 2, 3	128, 192, 256, 384, 512	4	2	512
ES/9000 Model 820	0, 1, 2, 3	256, 384, 512, 768, 1024	4	2	1024
ES/9000 Model 900	0, 1, 2, 3	512, 768, 1024	4	2	1024

Specifying the RSU Parameter

To prepare for reconfiguration of central storage, specify an RSU parameter in the IEASYSxx parmlib member or as a parameter during system initialization. The RSU parameter specifies the number of storage units in central storage that the operating system should keep available for reconfiguration.

Note that the RSU parameter applies only to central storage, not to expanded storage.

Reference Book

See *z/OS MVS Initialization and Tuning Reference* for the RSU parameter.

The RSU storage increments are *reconfigurable* or *non-preferred*. The remaining central storage increments are *non-reconfigurable* or *preferred*.

The default RSU value is 0. If the installation does not specify an RSU value, the default means that the operating system designates ALL installed central storage as preferred. With RSU=0, the system cannot be reconfigured.

Reconfigurable Storage Increments

The central storage that is shared by all processors in a configuration is logically divided into storage increments. For the following example, the installation specified RSU=8 to make half of the storage increments reconfigurable.

60M	Hardware storage area (HSA) and preferred
-----	---

56M	Reconfigurable
52M	Reconfigurable
48M	Reconfigurable
44M	Reconfigurable
40M	Reconfigurable
36M	Reconfigurable
32M	Reconfigurable
28M	Reconfigurable
24M	Preferred
20M	Preferred
16M	Preferred
12M	System queue area (SQA) and Preferred
8M	Preferred
4M	Preferred
0M	V=R and Preferred

The HSA cannot be in a storage increment that will be reconfigured. The HSA resides in the five highest megabytes of central storage.

RSU Example

Assume that a processor complex has 128 megabytes of central storage and that the storage increment size is 4 megabytes. Such a system has 32 storage increments.

If the system initializes one side with RSU=16, the operating system allocates as reconfigurable the 16 storage increments (64 megabytes) of the offline side. Thus, when the initialized side runs in physically partitioned mode, all its online storage (64 megabytes) is preferred.

RSU Parameter Specification

The RSU values recommended for the least system overhead and maximum capability for reconfiguration are as follows:

Processor	Recommended RSU Value
Uniprocessor	RSU=0
Non-partitionable multiprocessor	RSU=0
Partitionable processor in single-image mode	$\text{RSU} = \frac{\text{Total megabytes of installed central storage}}{2 * \text{Central storage increment size in megabytes}}$
Partitionable processor in physically partitioned mode	$\text{RSU} = \frac{\text{Total megabytes of installed central storage}}{2 * \text{Central storage increment size in megabytes}}$

Assignment of Storage Frames

Storage containing fixed pages cannot be reconfigured. The operating system assigns storage frames as follows:

Storage	Assignment
Non-preferred and preferred	Normal page allocation requests
Non-preferred and preferred	Short-term page fixes
Preferred	Long-term page fixes for non-swappable jobs

However, if a long-term page fix for a non-swappable job requires storage but the preferred storage units are full, the operating system may convert some non-preferred storage to preferred storage. If so, the amount of storage available for reconfiguration will be less than that specified in the RSU parameter. The operating system issues message IAR005I to notify the operator.

If the operator then tries to configure storage offline in preparation for partitioning, the system tries to free enough central storage to support the request. Central storage and the address ranges assigned to that storage cannot be configured offline either logically or physically until the required amount of storage is available.

The operating system normally tries to assign requests for long-term fixed pages to preferred storage frames when the requesting job was initiated as non-swappable. However, an authorized job can be initiated as swappable and, while running, issue a SYSEVENT macro to make itself non-swappable for a short period of time. The job may request long-term fixed pages that are assigned to non-preferred storage. Usually this request is not a problem because the job shortly makes itself swappable again. The system can free the storage that backs the long-term fixed pages when the job is swapped out when the storage is required for storage reconfiguration.

However, a long-running job may make itself non-swappable for long times and also request short-term fixed pages that cannot be freed until the job ends normally. Some of these requests may be satisfied from non-preferred storage. Because the frames cannot be freed by paging them out or by swapping out the job, storage reconfiguration may not be possible.

To resolve this problem, specify such jobs in the program properties table (PPT) in the SCHEDxx parmlib member.

Reference Book

See *z/OS MVS Initialization and Tuning Reference* for the SCHEDxx member.

Actions to Reconfigure Central Storage

Enter at the console with master authority or in a CONFIGxx parmlib member:

1. CONFIG STOR(E=id),OFFLINE command to configure a central storage element offline
2. CONFIG STOR(E=id),ONLINE command to configure a central storage element online

The system responds on the console on which the command was entered or on the master console for the system processing the CONFIGxx member.

When configuring a central storage element offline, the system may issue message IEE575A to indicate that central storage configuration is waiting to complete. The system may cancel the message in less than a minute. The system may issue and cancel the message several times.

If the message remains on the display for a long time, it indicates that the operating system cannot find sufficient reconfigurable central storage to satisfy the configuration request.

If the system displays message IEE575A for a long time:

1. Enter a DISPLAY M=STOR command to identify the job using the central storage that cannot be freed.
2. Do one of the following:
 - a. Cancel any jobs that are using the storage to allow the storage configuration to complete.
 - b. Reply C to end the storage configuration process.

Any central storage already configured offline remains offline. If desired, bring this central storage element back online by entering a CONFIG STOR(E=x),ONLINE command.

Actions to Reconfigure Expanded Storage

Enter at the console with master authority or in a CONFIGxx parmlib member:

1. CONFIG ESTOR(E=id),OFFLINE to configure an expanded storage element offline
2. CONFIG ESTOR(E=id),ONLINE to configure an expanded storage element online

The system responds on the console on which the command was entered or on the master console for the system processing the CONFIGxx member.

Enter a separate CONFIG command for each expanded storage element to go offline or to go online. The system can have as many as four elements, two per side, numbered 0 - 3. Specify these numbers, one per command, in the E=id keyword.

To determine the status of installed expanded storage before issuing the CONFIG command, enter a DISPLAY M=ESTOR(E) command.

Reconfiguring Channel Paths

Channel paths may be reconfigured for partitioning and merging. Under PR/SM, channel paths may be reconfigured to move a channel path from one logical partition to another.

Reconfigure channel paths carefully. Reconfiguration can remove connections to DASDs or MCS consoles that are critical to the operation of MVS. If this occurs, system operation will fail.

Actions to Reconfigure Channel Paths

To configure channel paths, enter the following:

- To configure channel paths individually offline or online:

```
CONFIG CHP(x),OFFLINE  
CONFIG CHP(x),ONLINE
```

- To configure all channel paths owned by a side of a partitionable processor complex:

```
CONFIG CHP(ALL,x),OFFLINE  
CONFIG CHP(ALL,x),ONLINE
```

These commands are particularly useful when partitioning or merging the complex.

- To configure a range of channel paths:

```
CONFIG CHP(x-y),OFFLINE  
CONFIG CHP(x-y),ONLINE
```

The system determines which devices are connected to a channel path and if that path is the last path to a device. To configure offline the last path to a device, enter a CONFIG command with one of the following operands:

- UNCOND operand: To configure offline the last path to an unallocated, online device
- FORCE operand: To configure offline the last path to a device regardless of the state of the device.

To make sure that FORCE is intended, the operating system issues message IEE800D to ask whether to continue or stop the CONFIG command processing.

Enter all the CONFIG commands at a console with master authority. The system responds on the console on which the command was entered.

Reconfiguring I/O Devices

To reconfigure I/O devices, enter a VARY ONLINE/OFFLINE command at the console with master authority. For a VARY OFFLINE command for a device that is currently in use, the operating system marks the device 'pending offline'. The operating system makes no further allocations to the device unless the volume mounted on the device is specifically requested.

Because vary offline processing cannot complete until a device is unallocated, either wait until the jobs using the device complete or cancel them.

Before reconfiguring a complex from single-image mode to physically-partitioned mode, complete or cancel any tape volume mounts for devices to be reconfigured. Then enter a CONFIG CHP(ALL,n),OFFLINE,UNCOND command.

If a tape mount is pending when the CONFIG command is entered, the tape drive(s) might not start after they are mounted and the system has been partitioned. Another way to avoid this problem is to enter a VARY device online command for the tape drive(s).

Reconfiguring a Coupling Facility

A coupling facility can be reconfigured to perform maintenance or to upgrade its level of control code. Before reconfiguring a coupling facility, you should have a plan in place for its orderly shutdown, during which time structures that are in the coupling facility can be either relocated to another coupling facility or deallocated

(with a possible data loss). Applications using the coupling facility for its structure(s) should have documented procedures for how to relocate or deallocate them.

Refer to *Parallel Sysplex Configuration Assistant* at <http://www.ibm.com/s390/psa/psotool> for help with the configuration of a coupling facility.

Chapter 14. Dynamic Partitioning and Merging

Reconfiguration of a partitionable processor complex can go beyond adding and removing hardware units; it can partition the complex into two sides and it can merge the two sides.

- **Partitioning** consists of changing the processor complex from single-image (SI) mode to physically partitioned (PP) mode. Partitioning divides the central and expanded storage into two sides and divides the channel paths and central processors (CPU) between the two sides.
- **Merging** consists of changing the processor complex from physically partitioned mode to single-image mode. Merging brings the two sides together so that all of the central and expanded storage, the channel paths, and the CPUs work together as one system.

Note that, when a processor complex is in single-image mode, the jobs running on it have more processing power and storage available than when the same processor complex is in physically partitioned mode.

Reference Book

See *Processor Recovery Guide* for partitioning and merging using the hardware frames on the system console.

Sequence for Partitioning and Merging

CONFIG commands are used to perform the partitioning and merging. Their order can affect the function and performance of the system. The recommended order in which elements should be configured offline during partitioning is:

1. **Channel paths** should go offline first to reduce the load on the CPUs and to allow capturing of channel activity data.
2. **CPUs** should go offline next to reduce the workload before central storage goes offline. If a Vector Facility or Integrated Cryptographic Feature (ICRF) is associated with a central processor that is going offline, the system also takes the VF or ICRF offline.

Because Vector Facilities go offline with their associated CPUs, do not enter CONFIG VF commands during the partitioning process.

3. **Expanded storage elements**, if applicable, should go offline next while central storage is still available to be used to migrate data from expanded storage to auxiliary storage.
4. **Central storage elements** should be the last to go offline.

For merging, configure hardware units back online in reverse order. During merging, the time-of-day (TOD) clocks in both partitions must be synchronized before the system can start channel measurements. Using the recommended merge order, configure the CPUs online before the channel paths; in these operations, the system suspends channel measurements for about 16 seconds. If, however, the channel paths are configured online before the processors, the system suspends channel measurements for a longer time: from the time the first channel path is configured online until after the processors are configured online and the TOD clocks are synchronized.

The order used in the CONFIGxx parmlib member preserves channel measurement.

Reconfiguration of JES3 Systems

To reconfigure storage when the operating system includes JES3, use MAINPROC statements when initializing JES3. Include MAINPROC statements for all planned configurations.

Partitioning or merging a JES3 processor (main) affects job selection. Consider the resources available for each processor and define the job selection modes appropriately. If needed, modify the selection modes for a processor (main) by entering the *F,G,main,S,selectname command.

Using JES3, jobs can control the processors on which they are to run:

- The //*MAIN SYSTEM statement
- Job class or group, which may be associated with a certain processor

Jobs directed to a processor that is configured offline will not run. Similarly, jobs requiring resources attached only to a partitioned-off processor will not run. If the processor is not going to be returned to the system, restart these jobs in the SETUP phase.

RSU Parameter

The following table lists the RSU parameter values calculated for central storage sizes for different partitionable processor complexes. The formula used to calculate the RSU values is:

$$\text{RSU} = \frac{\text{Total megabytes of installed central storage}}{2 * \text{Central storage increment size in megabytes}}$$

For more information on calculating the RSU parameter value, see “Specifying the RSU Parameter” on page 13-11.

Processor Complex	Maximum Installed Central Storage (megabytes)	Storage Increment (megabytes)	RSU Value
ES/3090 Model 250J	128	4	16
ES/3090 Model 250JH	256	4	32
ES/3090 Model 250S	128	4	16
ES/3090 Model 280E	128	4	16
ES/3090 Model 280J	256	4	32
ES/3090 Model 280S	256	4	32
ES/3090 Model 380J	512	4	64

Processor Complex	Maximum Installed Central Storage (megabytes)	Storage Increment (megabytes)	RSU Value
ES/3090 Model 380S	512	4	64
ES/3090 Model 400	128	2	32
ES/3090 Model 400E	256	4	32
ES/3090 Model 400J	512	4	64
ES/3090 Model 400S	512	4	64
ES/3090 Model 500E	256	4	32
ES/3090 Model 500J	512	4	64
ES/3090 Model 500S	512	4	64
ES/3090 Model 600E	256	4	32
ES/3090 Model 600J	512	4	64
ES/3090 Model 600S	512	4	64
ES/9000 Model 620	512	4	64
ES/9000 Model 720	512	4	64
ES/9000 Model 820	1024	4	128
ES/9000 Model 900	1024	4	128

Central Storage in a Partitionable Processor Complex

Central storage is divided into storage frame. The following example shows storage frames and their use.

60M	Hardware storage area (HSA) and preferred
56M	Reconfigurable
52M	Reconfigurable
48M	Reconfigurable
44M	Reconfigurable
40M	Reconfigurable
36M	Reconfigurable
32M	Reconfigurable
28M	Reconfigurable

24M	Preferred
20M	Preferred
16M	Preferred
12M	System queue area (SQA) and Preferred
8M	Preferred
4M	Preferred
0M	V=R and Preferred

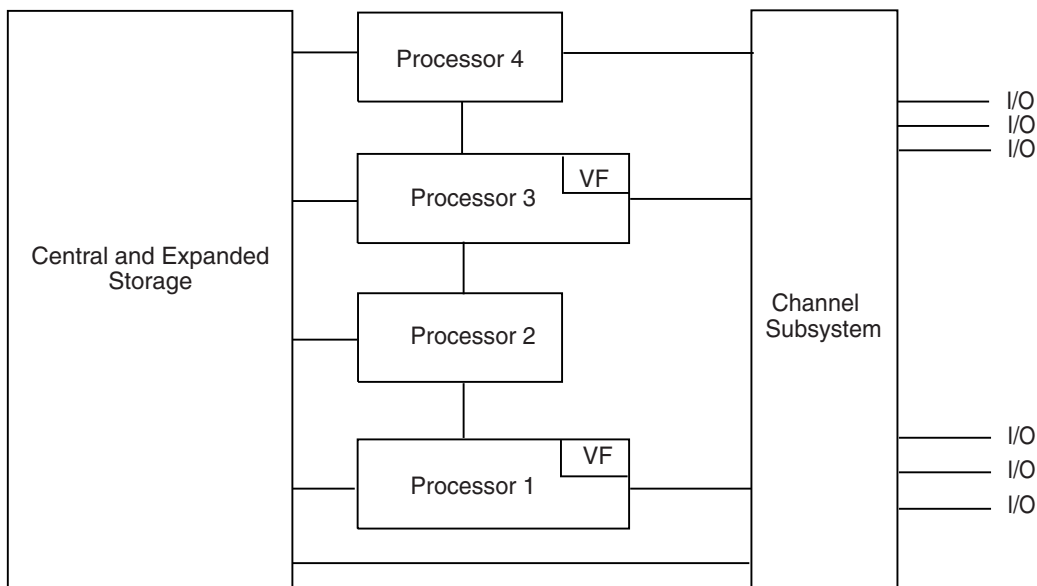
Characteristics of a Partitionable Processor

The partitioning and merging examples given in this chapter assume the following:

- ES/3090 Model 400.
- 96 channel paths.
- Installed central storage of 128 megabytes consisting of four 32-megabyte central storage elements with 2-megabyte storage increments.
- RSU=32 was specified at IPL.
- V=R area is contained in central storage increment 0-2 megabyte.
- System queue area (SQA) is contained in 12-16 megabyte even frames.
- Central storage ranges 0-58 and 122-128 megabytes cannot be reconfigured.
- Central storage range 58-122 megabytes is reconfigurable.
- The hardware save area (HSA) is contained in central storage increment 126-128 megabytes.
- Installed expanded storage is 512 megabytes and consists of four 128-megabyte expanded storage elements, two on each side.

Configuration in Single-Image Mode

Before configuring from single-image mode to physically partitioned mode, the processor complex appears as follows:



Before configuring from single-image mode to physically partitioned mode, assume the storage layout is as follows. This view of central storage assumes a power-on reset (POR) in single-image mode and an IPL with RSU=32.

Side0		Side1	
SE0	SE1	SE2	SE3
Online	Online	Online	Online
58-60M OddFrames Reconfigurable	62-64M OddFrames Reconfigurable	122-124M OddFrames Preferred	126-128M OddFrames Preferred
58-60M EvenFrames Reconfigurable	62-64M EvenFrames Reconfigurable	122-124M EvenFrames Preferred	126-128M EvenFrames Preferred
56-58M OddFrames Preferred	60-62M OddFrames Reconfigurable	120-122M OddFrames Reconfigurable	124-126M OddFrames Preferred
56-59M EvenFrames Preferred	60-62M EvenFrames Reconfigurable	120-122M EvenFrames Reconfigurable	124-126M EvenFrames Preferred
.	.	.	.
.	.	.	.
.	.	.	.
2-4M EvenFrames Preferred	6-8M EvenFrames Preferred	66-68M EvenFrames Reconfigurable	70-72M EvenFrames Reconfigurable
0-2M OddFrames Preferred	4-6M OddFrames Preferred	64-66M OddFrames Reconfigurable	68-70M OddFrames Reconfigurable
0-2M EvenFrames Preferred	4-6M EvenFrames Preferred	64-66M EvenFrames Reconfigurable	68-70M EvenFrames Reconfigurable

Notes

SE0	SE1	SE2	SE3
2reconfigurable subincrements	4reconfigurable subincrements	30reconfigurable subincrements	28reconfigurable subincrements
30preferred subincrements	28preferred subincrements	2preferred subincrements	4preferred subincrements

Example of Partitioning

The following actions show how to configure from single-image mode to physically partitioned mode. Side 1 is to be configured offline. This example is for an ES/3090 Model 400.

1. **Enter DISPLAY M=SIDE** to determine the status of resources on each side of the system.

The system status would look like this:

```
IEE174I 05.43.18 DISPLAY M
SIDE STATUS
SIDE              0              1
STATUS:           ONLINE        ONLINE
CPU:              1-2            3-4
VF:               1              3
CHP:              0-2F           40-6F
STOR(E=X):        0-1            2-3
ESTOR(E=X):       0-1            2-3
```

```
TOTAL STOR:      128M          UNASSIGNED: 0M
TOTAL ESTOR:     512M
```

2. **Enter CONFIG CHP(ALL,1),OFFLINE,UNCOND**

Note: When partitioning, before entering the CONFIG CHP(ALL,n),OFFLINE,UNCOND command, complete or cancel any mounts that may be affected by this command.

The following messages are displayed on the console with master authority:

```
IEE172I ALL CHANNEL PATHS ON SIDE 1 NOW OFFLINE
IEE503I CHP(ALL,1),OFFLINE
IEE712I CONFIG PROCESSING COMPLETE
```

1. **Enter CONFIG CPU(3,4),OFFLINE**

The following messages are displayed on the console with master authority:

```
IEE505I CPU(3),OFFLINE
IEE505I VF(3),OFFLINE
IEE505I CPU(4),OFFLINE
IEE712I CONFIG PROCESSING COMPLETE
```

The Vector Facility associated with CPU3 goes offline with its CPU. When CPU3 is later brought online, the Vector Facility will come online also.

1. **Enter CONFIG ESTOR(E=2),OFFLINE** to configure offline an expanded (extended) storage element on side 1.

Messages on the console with master authority indicate that the command was accepted:

```
IEE510I EXTENDED STORAGE LOCATIONS 256M TO 384M OFFLINE
IEE526I EXTENDED STORAGE ELEMENT(2) OFFLINE
IEE712I CONFIG PROCESSING COMPLETE
```

1. **Enter CONFIG ESTOR(E=3),OFFLINE** to configure the other expanded (extended) storage element offline.

Messages on the console with master authority indicate that the command was accepted:

```
IEE510I EXTENDED STORAGE LOCATIONS 384M TO 512M OFFLINE
IEE526I EXTENDED STORAGE ELEMENT(3) OFFLINE
IEE712I CONFIG PROCESSING COMPLETE
```

The reconfiguration of expanded storage elements on side 1 is now complete.

1. **Enter CONFIG STOR(E=2),OFFLINE** to configure central (real) storage offline.

These messages are displayed on the console with master authority:

```
IEE510I REAL STORAGE LOCATIONS 58M TO 60M OFFLINE
IEE510I REAL STORAGE LOCATIONS 64M TO 68M OFFLINE
IEE510I REAL STORAGE LOCATIONS 72M TO 76M OFFLINE
IEE510I REAL STORAGE LOCATIONS 80M TO 84M OFFLINE
IEE510I REAL STORAGE LOCATIONS 88M TO 92M OFFLINE
IEE510I REAL STORAGE LOCATIONS 96M TO 100M OFFLINE
IEE510I REAL STORAGE LOCATIONS 104M TO 108M OFFLINE
IEE510I REAL STORAGE LOCATIONS 112M TO 116M OFFLINE
IEE510I REAL STORAGE LOCATIONS 120M TO 122M OFFLINE
IEE526I REAL STORAGE ELEMENT(2) OFFLINE
IEE712I CONFIG PROCESSING COMPLETE
```

After configuring SE2 offline, storage appears as shown in the following view. Because SE2 contains some HSA and preferred storage (the 122 - 124 megabyte range as shown on “Physical View of Central Storage” on page 13-8), the reconfiguration process consists of swapping this group of frames with the frames in the 58 - 60 megabyte range in SE0. This is why the storage from 122 - 124 megabytes is now in SE0, and the storage from 58 - 60 megabytes is in SE2 and offline.

Side 0

Side 1

SE 0 Online	SE 1 Online	SE 2 Offline	SE 3 Offline
122 - 124M Odd Frames Preferred	SE1 is the same as in Single Image Mode.	58 - 60M Odd Frames	SE3 is the same as in Single Image Mode.
122 - 124M Even Frames Preferred		58 - 60M Even Frames	
The rest of SE0 is the same as in Single Image Mode.		The rest of SE2 has the same address assignments as in Single Image Mode.	
1 Reconfigurable Subincrement	4 Reconfigurable Subincrements	32 Offline Subincrements	28 Reconfigurable Subincrements
31 Preferred Subincrements	28 Preferred Subincrements		4 Preferred Subincrements

- **Enter CONFIG STOR(E=3),OFFLINE** to configure central (real) storage offline.

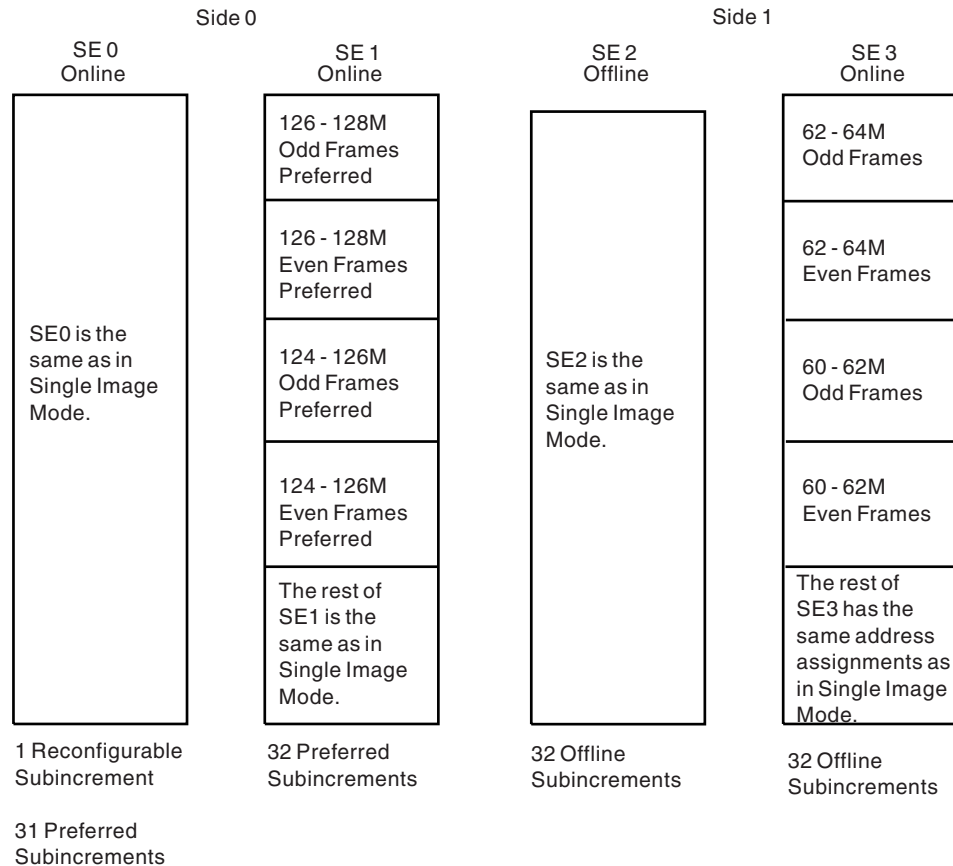
The following messages are displayed on the console with master authority. The system can issue more than one IEE575A message before the IEE510I messages.

```

IEE575A CONFIG STORAGE WAITING TO COMPLETE-REPLY C TO CANCEL
IEE510I REAL STORAGE LOCATIONS 60M TO 64M OFFLINE
IEE510I REAL STORAGE LOCATIONS 68M TO 72M OFFLINE
IEE510I REAL STORAGE LOCATIONS 76M TO 80M OFFLINE
IEE510I REAL STORAGE LOCATIONS 84M TO 88M OFFLINE
IEE510I REAL STORAGE LOCATIONS 92M TO 96M OFFLINE
IEE510I REAL STORAGE LOCATIONS 100M TO 104M OFFLINE
IEE510I REAL STORAGE LOCATIONS 108M TO 112M OFFLINE
IEE510I REAL STORAGE LOCATIONS 116M TO 120M OFFLINE
IEE526I REAL STORAGE ELEMENT(3) OFFLINE
IEE712I CONFIG PROCESSING COMPLETE

```

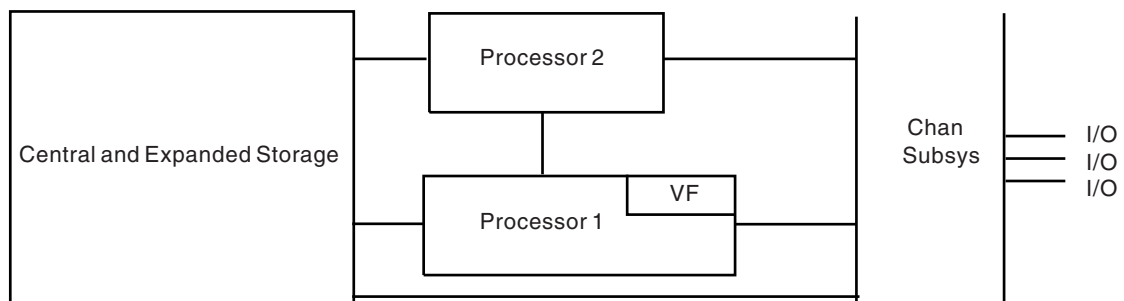
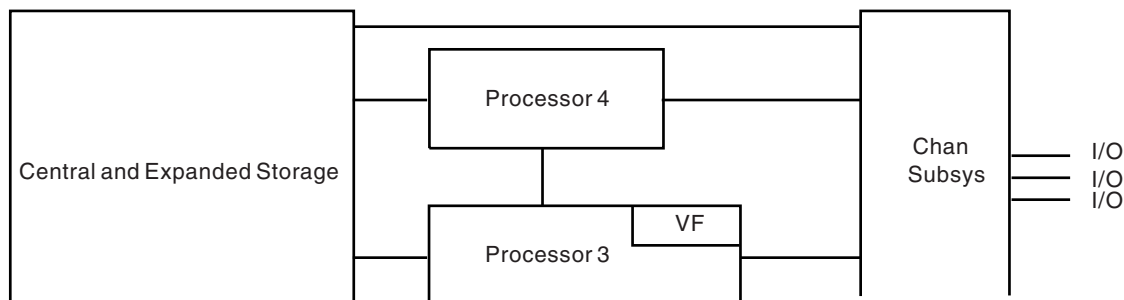
After configuring SE3 offline, storage appears as shown in the following:



Note: The preferred sub-increments 124 - 126 megabytes even, 124 - 126 odd, 126 - 128 even, and 126 - 128 odd have been swapped into storage element 1. They have been replaced in SE3 with reconfigurable sub-increments 60 - 62 even, 60 - 62 odd, 62 - 64 even, and 62 - 64 odd.

- This step and the next step in partitioning involve several actions that use the system console. On the system console, use the partition control frame (PARCTL) to vary side 1 offline. When processing completes, the side-1 system console becomes active.
- To bring up that side, perform the following on the side-1 system console:
 1. Define the configuration, using the Configuration frame.
 2. Perform power-on reset (POR).
 3. RelPL the system.

At this point, the installation has two separate systems, each monitored by its own service processor and each with its own system and service consoles.



DISPLAY M=SIDE Information

At the side-0 console with master authority, enter if desired a DISPLAY M=SIDE command to verify the physical partitioning of the system.

```

IEE174I hh.mm.ss MATRIX DISPLAY
SIDE STATUS
SIDE:           0                      1
STATUS:         ONLINE                 UNAVAILABLE
CPU:            1-2
VF:             1
CHP:            0-2F
STOR(E=x):      0-1
ESTOR(E=x):     0-1

TOTAL STOR:     128M                   UNASSIGNED: 0M
TOTAL ESTOR:    512M
*=OFFLINE
  
```

DISPLAY M Information

If more information is needed, enter if desired a DISPLAY M command.

```

IEE174I hh.mm.ss DISPLAY M nnn
PROCESSOR STATUS
CPU STATUS      SERIAL
 1 ONLINE VFON 1700903090
 2 ONLINE      2700903090

CHANNEL PATH STATUS
 0 1 2 3 4 5 6 7 8 9 A B C D E F
0 + + + + + + + + + + + + + + +
1 + + + + + + + + + + + + + + +
  
```

```

2  + + + + + + + + + + + + + + + +
***** SYMBOL EXPLANATIONS *****

+ ONLINE          - OFFLINE          . DOES NOT EXIST
HSA STATUS
ADDRESS=7F80000 LENGTH=512K
ADDRESS=7DD0000 LENGTH=192K

STORAGE SIZE STATUS
HIGH REAL STORAGE ADDRESS IS 128M
HIGH EXTENDED STORAGE ADDRESS IS 512M

REAL STORAGE STATUS
ONLINE-NOT RECONFIGURABLE
  FIRST 4K OF EVERY 8K FROM 0M TO 2M
  2M-58M
  122-128M
ONLINE-RECONFIGURABLE
  SECOND 4K OF EVERY 8K FROM 0M TO 2M
PENDING OFFLINE
  NONE
0M IN OFFLINE STORAGE ELEMENT(S)
0M UNASSIGNED
64M IN ANOTHER CONFIGURATION

REAL STORAGE ELEMENT STATUS
  0: OWNED STORAGE=32M UNASSIGNED STORAGE=0M
STATUS=ONLINE
  1: OWNED STORAGE=32M UNASSIGNED STORAGE=0M
STATUS=ONLINE
  STOR(E=2) IS PART OF ANOTHER CONFIGURATION-NO STATUS
OBTAINED
  STOR(E=3) IS PART OF ANOTHER CONFIGURATION-NO STATUS
OBTAINED

EXTENDED STORAGE STATUS
ONLINE-RECONFIGURABLE
  0M-256M
PENDING OFFLINE
  NONE
0M IN OFFLINE STORAGE ELEMENT(S)
256M IN ANOTHER CONFIGURATION

EXTENDED STORAGE ELEMENT STATUS
0: OWNED STORAGE=128M STATUS =ONLINE
1: OWNED STORAGE=128M STATUS =ONLINE
ESTOR(E=2) IS PART OF ANOTHER CONFIGURATION-NO
STATUS OBTAINED
ESTOR(E=3) IS PART OF ANOTHER CONFIGURATION-NO
STATUS OBTAINED

SIDE STATUS
SIDE:                0                1
STATUS:              ONLINE            UNAVAILABLE
CPU:                 1-2
VF:                  1
CHP:                 0-2F
STOR(E=x):           0-1
ESTOR(E=x):          0-1

TOTAL STOR:          128M              UNASSIGNED: 0M
TOTAL ESTOR:          512M
*=OFFLINE

```

Example of Merging Sides

The following actions show how to merge sides, which consists of configuring from a physically partitioned mode to single-image mode. Merging is essentially the reverse of partitioning. In this example, side 1 of the partitioned system is to be merged with the system running on side 0. This example is for an ES/3090 Model 400.

1. **Shut down the processing on side 1.** Perform an orderly shutdown of whatever programs are running on side 1. Ensure that the operating system on side 1 is stopped.
2. **Vary side 1 offline** by use of the Partition Control frame (PARCTL) on the side-1 system console.
3. **Vary side 1 online**, this time using the PARCTL on the side-0 system console.
Note: During the merging process, the hardware will be initializing the backup processor controller DASD. This hardware action does *not* prevent the operating system from configuring online the side 1 resources.

When side 1 has come online, enter the commands in the following steps at the side-0 console with master authority.

1. Enter **CONFIG STOR(E=2),ONLINE** and **CONFIG STOR(E=3),ONLINE** for the two side-1 storage elements. The expected response to each of these commands is a series of IEE524I messages that indicate that storage ranges have come online.

Responses for CF STOR(E=2),ONLINE at the side-0 console with master authority are:

```
IEE524I REAL STORAGE LOCATIONS 58M TO 60M ONLINE
IEE524I REAL STORAGE LOCATIONS 64M TO 68M ONLINE
IEE524I REAL STORAGE LOCATIONS 72M TO 76M ONLINE
IEE524I REAL STORAGE LOCATIONS 80M TO 84M ONLINE
IEE524I REAL STORAGE LOCATIONS 88M TO 92M ONLINE
IEE524I REAL STORAGE LOCATIONS 96M TO 100M ONLINE
IEE524I REAL STORAGE LOCATIONS 104M TO 108M ONLINE
IEE524I REAL STORAGE LOCATIONS 112M TO 116M ONLINE
IEE524I REAL STORAGE LOCATIONS 120M TO 122M ONLINE
IEE526I REAL STORAGE ELEMENT(2) ONLINE
IEE712I CONFIG PROCESSING COMPLETE
```

Responses for CF STOR(E=3),ONLINE at the side-0 console with master authority are:

```
IEE524I REAL STORAGE LOCATIONS 60M TO 64M ONLINE
IEE524I REAL STORAGE LOCATIONS 68M TO 72M ONLINE
IEE524I REAL STORAGE LOCATIONS 76M TO 80M ONLINE
IEE524I REAL STORAGE LOCATIONS 84M TO 88M ONLINE
IEE524I REAL STORAGE LOCATIONS 92M TO 96M ONLINE
IEE524I REAL STORAGE LOCATIONS 100M TO 104M ONLINE
IEE524I REAL STORAGE LOCATIONS 108M TO 112M ONLINE
IEE524I REAL STORAGE LOCATIONS 116M TO 120M ONLINE
IEE526I REAL STORAGE ELEMENT(3) ONLINE
IEE712I CONFIG PROCESSING COMPLETE
```

At this point all the storage in storage elements 2 and 3 is reconfigurable. All the preferred storage is in storage elements 0 and 1.

If, however, the storage in a specified storage element does not come online, it is because it does not have assigned storage addresses. You will receive a single message, instead of the usual series:

```
IEE574I NO STORAGE TO COME ONLINE IN REAL STORAGE ELEMENT(x).
```

In this case, enter any remaining **CONFIG STOR(E=x)** commands not yet entered, then do the following two steps:

- a. Enter **DISPLAY M=STOR** to find out the amount (ddM UNASSIGNED) of storage that does not have assigned addresses.
- b. Enter **CONFIG STOR(ddM),ONLINE** to assign storage addresses to this storage.

The previously unavailable storage in storage element x should now come online.

If all central storage elements are now online, continue with the next step.

1. Enter **CONFIG ESTOR(E=2),ONLINE**

The response at the side-0 console with master authority is:

```
IEE524I EXTENDED STORAGE LOCATIONS 256M TO 384M ONLINE
IEE526I EXTENDED STORAGE ELEMENT(2) ONLINE
IEE712I CONFIG PROCESSING COMPLETE
```

1. **Enter CONFIG ESTOR(E=3),ONLINE**

The response at the side-0 console with master authority is:

```
IEE524I EXTENDED STORAGE LOCATIONS 384M TO 512M ONLINE
IEE526I EXTENDED STORAGE ELEMENT(3) ONLINE
IEE712I CONFIG PROCESSING COMPLETE
```

2. **Enter CONFIG CPU(3,4),ONLINE**

The response at the side-0 console with master authority is:

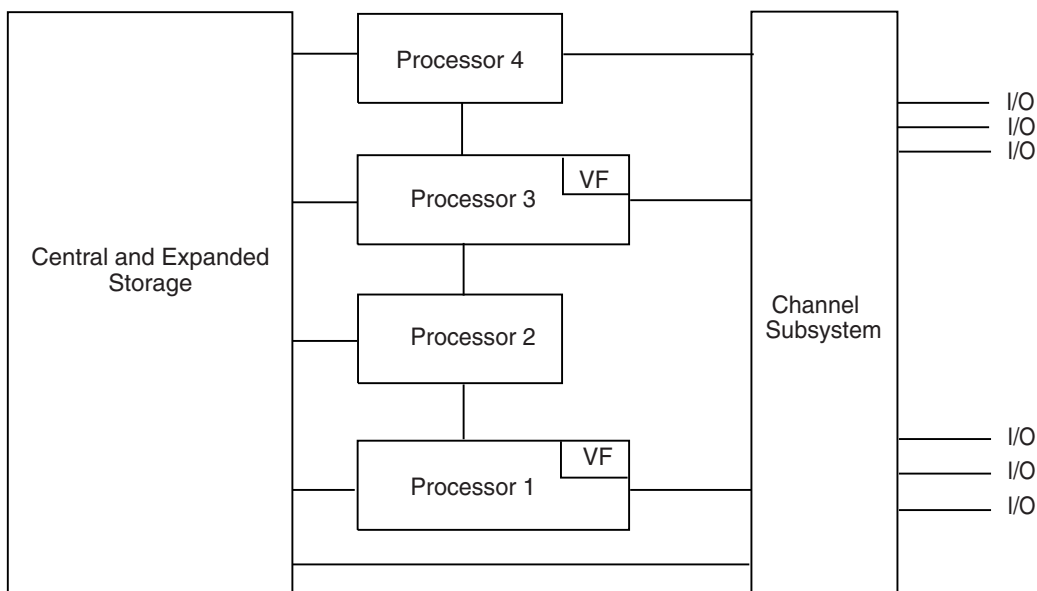
```
*09 IEA889A DEPRESS TOD CLOCK SECURITY SWITCH R 09,Y
IEE600I REPLY TO 09 IS;Y
*10 IEA889A DEPRESS TOD CLOCK SECURITY SWITCH R 10,Y
IEE600I REPLY TO 10 IS;Y
IEE504I CPU(3),ONLINE
IEE504I VF(3),ONLINE
IEE504I CPU(4),ONLINE
IEE712I CONFIG PROCESSING COMPLETE
```

3. **Enter CONFIG CHP(ALL,1),ONLINE**

The response at the side-0 console with master authority is:

```
IEE754I NOT ALL PATHS BROUGHT ONLINE WITH CHP(4B)
IEE754I NOT ALL PATHS BROUGHT ONLINE WITH CHP(4E)
IEE754I NOT ALL PATHS BROUGHT ONLINE WITH CHP(5D)
IEE172I ALL CHANNEL PATHS ON SIDE 1 ARE NOW ONLINE
IEE502I CHP(ALL,1),ONLINE
IEE712I CONFIG PROCESSING COMPLETE
```

After all the channel paths have been configured online, the system is operating in single-image mode. The processor complex now appears as follows.



DISPLAY M=SIDE Information

Enter DISPLAY M=SIDE to verify single image mode.

```

IEE174I hh.mm.ss MATRIX DISPLAY
SIDE STATUS
SIDE:                0                1
STATUS:              ONLINE          ONLINE
CPU:                 1-2              3-4
VF:                  1                3
CHP:                 0-2F             40-6F
STOR(E=x):           0-1              2-3
STOR(E=x):           0-1              2-3
ESTOR(E=x):          0-1              2-3

TOTAL STOR:          128M              UNASSIGNED: 0M
TOTAL ESTOR:          512M
*=OFFLINE

```

DISPLAY M Information

If more information is needed, enter a DISPLAY M command.

```

IEE174I hh.mm.ss DISPLAY M

PROCESSOR STATUS
CPU STATUS      SERIAL
 1 ONLINE VFON 1700903090
 2 ONLINE      2700903090
 3 ONLINE VFON 3700903090
 4 ONLINE      4700903090

CHANNEL PATH STATUS
 0 1 2 3 4 5 6 7 8 9 A B C D E F
0 + + + + + + + + + + + + + + +
1 + + + + + + + + + + + + + + +
2 + + + + + + + + + + + + + + +
3 + + + + + + + + + + + + + + +
4 + + + + + + + + + + + + + + +
5 + + + + + + + + + + + + + + +
6 + + + + + + + + + + + + + + +
***** SYMBOL EXPLANATIONS *****

+ ONLINE          - OFFLINE          . DOES NOT EXIST
HSA STATUS
ADDRESS=7F80000 LENGTH=512K
ADDRESS=7DD0000 LENGTH=192K

STORAGE SIZE STATUS
HIGH REAL STORAGE ADDRESS IS 128M
HIGH EXTENDED STORAGE ADDRESS IS 512K

REAL STORAGE STATUS
ONLINE-NOT RECONFIGURABLE
  FIRST 4K OF EVERY 8K FROM 0M TO 2M
  2M-58M
  122M-128M
ONLINE-RECONFIGURABLE
  SECOND 4K OF EVERY 8K FROM 0M TO 2M
  58M-122M
PENDING OFFLINE
  NONE
0M IN OFFLINE STORAGE ELEMENT(S)
0M UNASSIGNED
0M IN ANOTHER CONFIGURATION
REAL STORAGE ELEMENT STATUS
0: OWNED STORAGE=32M UNASSIGNED STORAGE=0M STATUS=ONLINE
1: OWNED STORAGE=32M UNASSIGNED STORAGE=0M STATUS=ONLINE
2: OWNED STORAGE=32M UNASSIGNED STORAGE=0M STATUS=ONLINE

```

3: OWNED STORAGE=32M UNASSIGNED STORAGE=0M STATUS=ONLINE

EXTENDED STORAGE STATUS

ONLINE-RECONFIGURABLE

0M-512M

PENDING OFFLINE

NONE

0M IN OFFLINE STORAGE ELEMENT(S)

0M IN ANOTHER CONFIGURATION

EXTENDED STORAGE ELEMENT STATUS

0: OWNED STORAGE=128M STATUS=ONLINE

1: OWNED STORAGE=128M STATUS=ONLINE

2: OWNED STORAGE=128M STATUS=ONLINE

3: OWNED STORAGE=128M STATUS=ONLINE

SIDE STATUS

SIDE:	0	1
STATUS:	ONLINE	ONLINE
CPU:	1-2	3-4
VF:	1	3
CHP:	0-2F	40-6F
STOR(E=X):	0-1	2-3
ESTOR(E=X):	0-1	2-3

TOTAL STOR: 128M UNASSIGNED: 0M

TOTAL ESTOR: 512M

*=OFFLINE

Chapter 15. Reconfiguration under PR/SM

In a processor complex with the Processor Resource/Systems Manager (PR/SM) feature, the system resources of the processor complex or of the complex's side can be divided into two or more logical partitions. A logical partition is a subset of hardware resources that supports an operating system. The logical partitions consist of central storage, expanded storage, channel paths, Vector Facilities, and central processors (CPU), with the CPUs dedicated to one logical partition or shared by partitions. Each logical partition contains a completely separate operating system.

Reconfiguration can dynamically place resources online to a logical partition and offline from a logical partition. When a resource is placed offline from one logical partition and then placed online to another partition, the resource is effectively moved from one partition to another.

PR/SM dynamic storage reconfiguration allows storage to be configured online or offline to a logical partition while the system in it continues running. Use PR/SM dynamic storage reconfiguration to configure storage offline to an active logical partition and to activate a new logical partition in the offline storage.

A channel path can also be configured offline to a logical partition and then configured online to another logical partition.

Uses for Logical Partitions

An installation usually uses logical partitions to control operations and balance work loads. For example, an installation can allocate all of its central and expanded storage to a production logical partition during the day. In the evening, the installation could reconfigure some of the production logical partition's storage into another logical partition that would handle overnight batch processing.

Uses in a Sysplex

In a sysplex on a processor complex with the PR/SM feature, an installation can use logical partitions to provide high availability for important work. For example, an alternate system can take over the work load of a failing system and can acquire the system resources previously used by the failing system. The goal of is reconfiguration is to make an application available quickly and without operator actions when a failure occurs.

Reference Book

See *z/OS MVS Setting Up a Sysplex* for the XCF PR/SM policy, which can establish the automatic recovery for high availability.

Actions to Use Logical Partitions

To use logical partitions, an installation does the following:

- Defines the logical partitions on the Logical Partition Definition (LPDEF) frame on the system console.

- For each system that will run in a logical partition, defines the RSU parameter in the IEASYSxx parmlib member or at system initialization. The RSU parameter identifies the amount of central storage that can be reconfigured.
- Initializes a system into a new logical partition.
- Enters CONFIG operator commands on the master console to reconfigure the storage online or offline to logical partitions.

This chapter describes these actions and ends with examples.

Reference Book

See *PR/SM Planning Guide* for information about logical partitions.

Defining Logical Partitions for PR/SM Dynamic Storage Reconfiguration

Define logical partitions on the LPDEF frame of the system console. Indicate which logical partitions are dynamically reconfigurable by specifying a reserved storage allocation value for the logical partition. A reserved storage allocation is the amount of central and expanded storage that you can dynamically configure online or offline to a logical partition. Specify the following for the reconfigurable logical partitions:

- **Initial storage allocation** value for central and expanded storage to be assigned to the logical partition.

Define an origin for the initial allocation if the logical partition is to be placed in the reserved storage defined for another logical partition.

- **Reserved storage allocation** value for central storage, expanded storage, or both. The reserved value is the maximum amount of storage to be configured online or offline to the logical partition. If reserved storage is not specified, no reconfiguration can occur.

If reserved storage is specified for a logical partition, an origin must be specified; otherwise, the system will not accept the definition of the logical partition.

- **Origin** for central storage and, if specified, expanded storage.

The origin for central storage defines the starting megabyte where the central storage begins for the logical partition. If enough central storage is not available to satisfy the initial storage allocation starting at this origin, logical partition activation will fail.

The origin for expanded storage defines the starting megabyte where the expanded storage begins for the logical partition. If enough expanded storage is not available to satisfy the initial storage allocation starting at this origin, logical partition activation will fail. If there is no initial amount of expanded storage defined, the reserved amount of expanded storage starts at this origin.

For both central and expanded storage, the specification of the origin parameter provides the only way to overlap storage definitions. For example, the reserved storage definition for one logical partition can overlap an initial storage definition of another logical partition when the origin parameter is specified for both logical partitions.

Logical Storage Elements

For central storage, the initial storage allocation is *logical storage element 0*. The reserved storage allocation is *logical storage element 1*.

For expanded storage, *logical storage element 0* is either the initial storage allocation, if specified, or is the reserved storage allocation if no initial storage allocation is specified. For expanded storage, if an initial storage allocation is specified, the reserved storage allocation is *logical storage element 1*.

Central Storage Logical Storage Elements

Initial Storage Allocation (megabytes)	Logical Storage Element	Reserved Storage Allocation (megabytes)	Logical Storage Element	Comments
>0	E=0	0		No storage reconfiguration
>0	E=0	>0	E=1	Storage reconfiguration

Expanded Storage Logical Storage Elements

Initial Storage Allocation (megabytes)	Logical Storage Element	Reserved Storage Allocation (megabytes)	Logical Storage Element	Comments
0		0		No expanded storage reconfiguration
0		>0	E=0	Expanded storage reconfiguration
>0	E=0	0		Expanded storage reconfiguration in logical storage element 0 if origin is specified.
>0	E=0	>0	E=1	Expanded storage reconfiguration

Defining Channel Paths for Reconfiguration

Channel paths can be moved from one logical partition to another. For this, the channel paths must be defined as reconfigurable to the input/output control program (IOCP).

On a 3090, the LPCHNA frame identifies the reconfigurable channel paths.

Reference Book

See *IOCP User's Guide and Reference* for information about IOCP.

Defining the RSU parameter

As with physical partitioning, the amount of central storage that is reconfigurable must also be specified on the RSU parameter in the IEASYSxx parmlib member or at system initialization, so that the system will treat the required amount of storage as reconfigurable. For a logical partition, the RSU parameter specifies the amount

of reconfigurable central storage as a certain number of storage increments. The size of these storage increments varies, depending on the processor in use. For example, while 1 megabyte is the correct storage increment for many processors, 4 megabytes is the correct increment for 711-based ES/9021 processors.

The value you specify for the RSU parameter should correspond to the amount of reserved central storage specified on the LPDEF frame. For example, specify a value of:

- 48 for processors with a storage increment of 1 megabyte and 48 megabytes of reserved central storage specified on the LPDEF frame.
- 12 for processors with a storage increment of 4 megabytes (such as 711-based ES/9021 processors) and 48 megabytes of reserved central storage specified on the LPDEF frame.

Reference Books

See for a description of the IEASYSxx parmlib member and the RSU parameter. See *PR/SM Planning Guide* for information about the correct RSU parameter values for specific processor levels.

Initializing a System into a Logical Partition

To initialize a system into a logical partition that is defined with reserved storage (logical storage element 1), IPL the system with either of the following:

- *Load Clear*
- *System Reset Clear* followed by *Load*

Load Clear or *System Reset Clear* followed by *Load* will configure offline the reserved central storage. This action is necessary to prevent MVS from fixing control blocks in the storage that is to be reconfigured. If this procedure is not followed, key MVS control blocks are fixed in the reserved storage and it cannot be reconfigured offline.

Stand-Alone Dump IPL in Logical Partitions

For logical partitions defined with reserved storage, load the stand-alone dump program into the logical partition using either of the following:

- *Load*
- *System Reset* followed by *Load*

Either of these loads will keep the reserved central storage online, so that the dump program can dump the contents of the reserved storage.

Using the CONFIG Command to Configure Storage

Use the CONFIG command to configure logical storage elements online or offline to logical partitions. Enter the command from the master console.

Note: Only logical storage *elements* can be configured online or offline to logical partitions. The RSU parameter specifies the number of megabyte central storage increments that are reconfigurable; however, MVS treats the reconfigurable storage as one storage element. For central storage, the initial storage allocation is *logical storage element 0*. The reserved storage allocation is *logical storage element 1*.

Configuring Storage Online to the Logical Partition

Configure central and expanded reserved storage elements online to the logical partition by entering the following commands:

```
CONFIG STOR(E=1),ONLINE  
CONFIG ESTOR(E=0),ONLINE or CONFIG ESTOR(E=1),ONLINE
```

This causes the reserved central storage element and the reserved expanded storage element (E=0, if no initial storage was specified, or E=1, if initial was specified) to be configured online to the logical partition.

Configuring Storage Offline to the Logical Partition

Configure central and expanded reserved storage elements offline to the logical partition by entering the following commands:

```
CONFIG STOR(E=1),OFFLINE  
CONFIG ESTOR(E=0),OFFLINE or CONFIG ESTOR(E=1),OFFLINE
```

The CONFIG command will configure offline to the logical partition the reserved central storage element and the reserved expanded storage element (E=0, if no initial storage was specified, or E=1, if initial was specified).

When Only Some of the Reserved Central Storage is Available

When only a portion of the reserved *central* storage element is available (not being used by another active logical partition) when you enter CONFIG, the system will configure online to the receiving logical partition the available portion of storage. The storage that is still allocated to another active logical partition is not configured online to the logical partition you are reconfiguring. If the remainder of the reserved central storage element becomes available later, you must first configure the entire reserved central storage element offline, and then configure it online, to acquire the remaining portion.

When Only Some of the Reserved Expanded Storage is Available

When only a portion of reserved *expanded* storage element is available when you enter CONFIG, the system will fail your request. The entire amount of reserved expanded storage must be available before you can configure it online to another logical partition.

Automating Dynamic Storage Reconfiguration

The system forces the reserved central storage element offline at IPL when the system programmer enters (1) LOAD CLEAR or (2) SYSTEM RESET CLEAR followed by LOAD. You can cause the reserved central storage element to be configured online automatically at system initialization by placing the following command in the COMMNDxx parmlib member:

```
CONFIG STOR(E=1),ONLINE
```

Dynamically Configuring Central Storage on an ES/9021 Processor

As for other processors, you also can specify central storage element 1 in a CONFIG command to reconfigure central storage on a 711-based model of an ES/9021 processor. However, for 711-based models with less than 2 gigabytes of central storage, you can configure central storage online or offline by specifying

megabytes of storage, rather than specifying a storage element ID, on the CONFIG command. The amount you specify must be a multiple of 4 megabytes.

When you configure central storage offline by specifying megabytes rather than the storage element ID, you can specify storage either from the reserved central storage element (logical storage element 1) or from the initial storage allocation (logical storage element 0). The total amount of storage to be configured offline must be specified through the RSU parameter. For example:

- Assume that you want to configure offline only the reserved central storage element for logical partition LP_A, which is defined with an initial allocation of 100 megabytes and a reserved allocation of 32 megabytes. To configure storage element 1 offline, you would enter:

```
CONFIG STOR (32M),OFFLINE
```

In this case, the value of the RSU parameter should be 8 (32 megabytes divided by 4, the size of a storage element for a 9021 processor).

- Given the same logical partition LP_A, assume that you want to configure offline both the reserved storage element and an additional 32 megabytes from storage element 0. You would enter:

```
CONFIG STOR (64M),OFFLINE
```

In this case, the value of the RSU parameter should be 16 ((32+32)/4).

Examples of Dynamic Storage Reconfiguration

Assume that an installation with a processor complex having 128 megabytes of central storage and 128 megabytes of expanded storage uses two logical partitions, PROD1 and TEST. The logical partitions are defined on the LPDEF frame as follows:

PARTITION	--CENTRAL (MB)--			--EXPANDED (MB)--		
	INIT	RSVD	ORIG	INIT	RSVD	ORIG
PROD1	100	22	0	108	20	0
TEST	23	0	100	20	0	108

Notes:

1. The initial central storage is 100 plus 23 megabytes, or 123 megabytes total. The remaining 5 megabytes are for the HSA, which occupies the highest area of central storage and is sized according to the installation's configuration.
2. A logical partition cannot be activated if its reserved central storage contains any portion of the HSA (because the HSA cannot be configured offline at IPL). When defining logical partitions, please be aware that the HSA can increase in size dynamically (for part of a megabyte) when the installation changes the I/O configuration data set (IOCDS) and then IMLs the system. Therefore, in this example, the installation has defined PROD1 with 22 megabytes of reserved storage, rather than 23, to ensure that PROD1 can be activated if the HSA extends down into address 122MB — 123MB.

The initial expanded storage is 128 megabytes total.

The LPDEF definition specifies reserved (RSVD) storage, which allows the central and expanded storage that is defined for the TEST logical partition to be configured

online to the PROD1 logical partition when TEST is not active. The origins (ORIG) for the reserved storages place them immediately above the initial allocations.

The following examples show the online/offline status of the processor's storage each time the installation reconfigures the storage. Note that the online/offline status is from the point of view of PROD1 and not that of TEST.

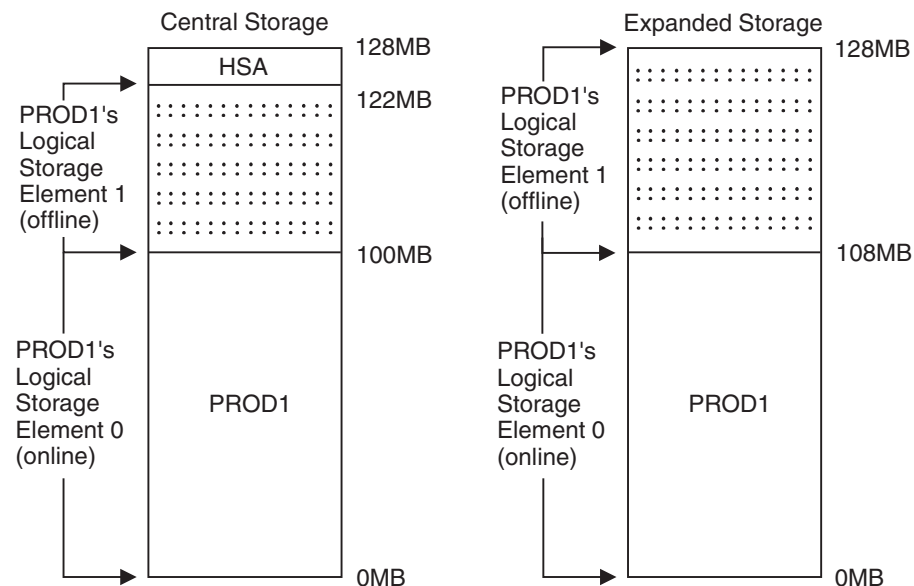
Setting up the First Logical Partition

During the first shift, the installation runs production jobs only. To use PROD1, the system programmer:

- Activates the PROD1 logical partition from the system console
- Initializes the PROD1 MVS system in the logical partition

At initialization, the MVS system's initial central storage allocation of 100 megabytes is online and its reserved central storage allocation of 22 megabytes is offline. The reserved central storage is offline at IPL, so that it will be reconfigurable.

The system programmer sets the PROD1 system's RSU parameter to 22 to match the PROD1 logical partition's reserved storage allocation. The result is as follows:

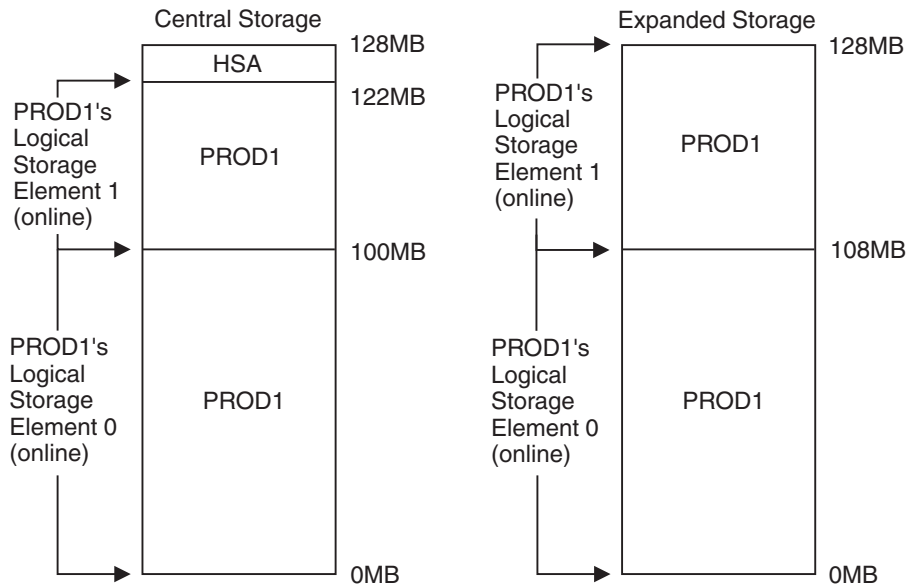


Example 1: Configuring Reserved Storage Online to the Logical Partition:

After the IPL is completed, the system programmer configures online to PROD1 its reserved storage. From the master console, the system programmer enters the following command:

```
CONFIG STOR(E=1),ONLINE
CONFIG ESTOR(E=1),ONLINE
```

This causes the reserved logical storage elements for central storage and expanded storage to be configured online to PROD1's initial storage allocation. The result is as follows:



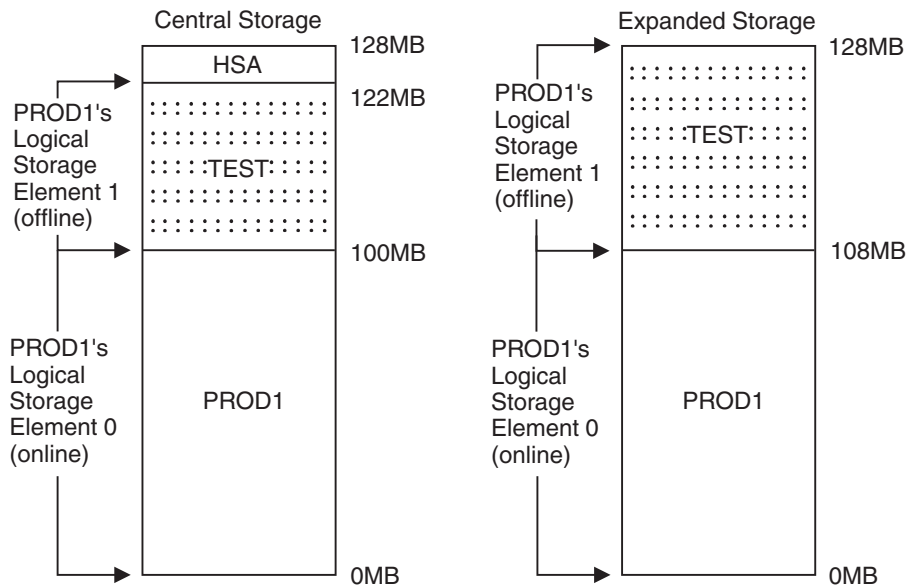
Note that the CONFIG commands can be placed in the COMMNDxx parmlib member so that the reserved storage will automatically be configured online to the logical partition at system initialization.

Example 2: Activating a Second Logical Partition in the Reserved Storage:

During the second shift, the installation's production work load is lighter. At this time, the installation tests new programs on a second MVS system, TEST. To activate TEST without disrupting PROD1, the system programmer first configures offline PROD1's reserved storage with the following commands:

```
CONFIG STOR(E=1),OFFLINE
CONFIG ESTOR(E=1),OFFLINE
```

Then, the system programmer activates TEST and initializes into TEST an MVS system. The result is as follows:

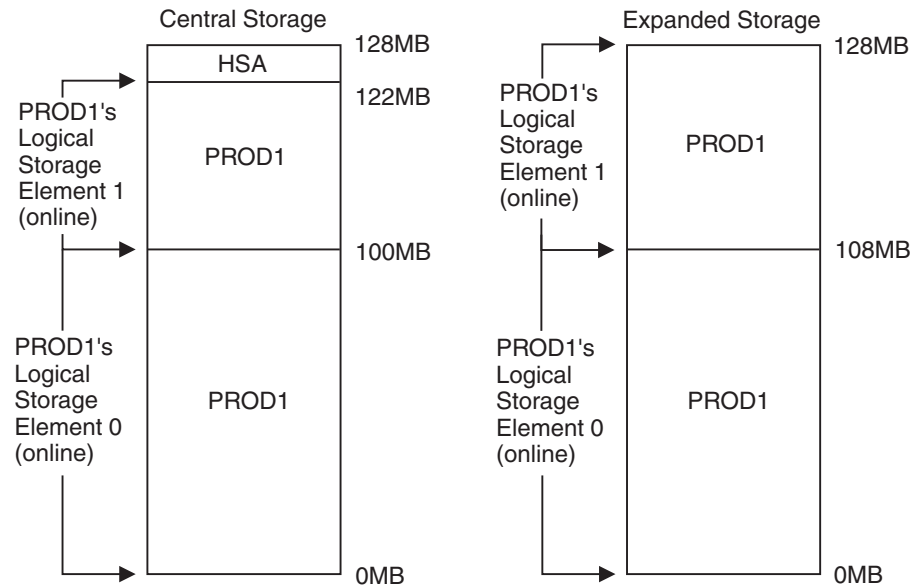


Example 3: Acquiring Storage from Another Logical Partition: On the following day, PROD1 needs all the storage resources again. The system programmer does the following:

1. Quiesces the TEST system
2. Deactivates the TEST logical partition
3. Enters the following commands to configure the reserved central and expanded storage online to PROD1:

```
CONFIG STOR(E=1),ONLINE
CONFIG ESTOR(E=1),ONLINE
```

The result is as follows:



Using the CONFIG Command to Move Channel Paths

Use the CONFIG command to configure a channel path from one logical partition to another. Do the following:

1. On the master console for the old owning partition, enter the following to remove the channel path from the partition. The *nn* is the channel path identifier.

```
CONFIG CHP(nn),OFFLINE
```

2. On the master console for the partition that is to own the channel path, enter:

```
CONFIG CHP(nn),ONLINE
```

Use this last command to configure a channel path online to a logical partition.

A channel path that is reconfigurable shows on the display for a DISPLAY M=CHP command for both partitions. One partition will show the channel path as online and the other partition as offline.

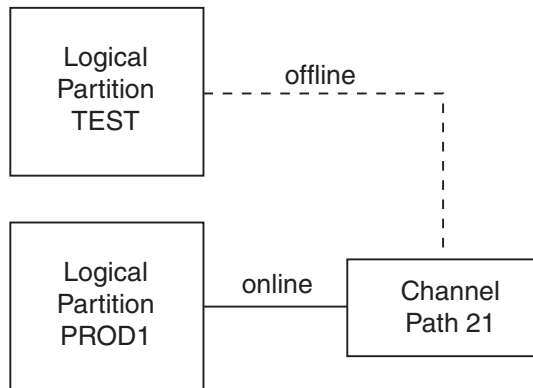
If a logical partition contains an MVS system, use the MVS CONFIG operator command. Channel paths can also be moved using service language commands on the system console. However, if the SLC are used, the MVS operating system does not recognize the move; then, errors will occur during I/O operations.

Reference Book

See *PR/SM Planning Guide* for details on moving channel paths.

Example of Moving a Channel Path from One Logical Partition to Another

For the example, assume that channel path 21 is online to the PROD1 logical partition, as follows:

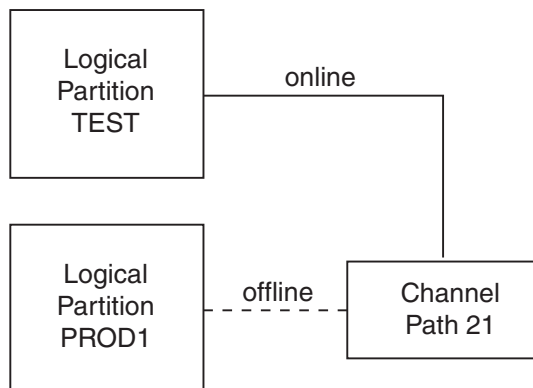


To move channel path 21 to the TEST logical partition, enter the following command on the master console for the operating system running in the PROD1 logical partition:

```
CONFIG CHP(21),OFFLINE
```

Then enter the following command on the master console for the operating system running in the TEST logical partition:

```
CONFIG CHP(21),ONLINE
```



Appendix A. Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/OS enable users to:

- Use assistive technologies such as screen-readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard
- Customize display attributes such as color, contrast, and font size

Using assistive technologies

Assistive technology products, such as screen-readers, function with the user interfaces found in z/OS. Consult the assistive technology documentation for specific information when using it to access z/OS interfaces.

Keyboard navigation of the user interface

Users can access z/OS user interfaces using TSO/E or ISPF. Refer to *z/OS TSO/E Primer*, *z/OS TSO/E User's Guide*, and *z/OS ISPF User's Guide Volume I* for information about accessing TSO/E and ISPF interfaces. These guides describe how to use TSO/E and ISPF, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

Appendix B. Creating Path Group Identifiers (PGIDs)

Any operating system that shares devices with MVS needs to use the same method as MVS to create path group identifiers (PGIDs).

In order to use the dynamic pathing feature of devices attached to system 390 and z/OS, MVS performs standard initialization at each device during IPL to uniquely identify the system to the device. Each physical path to the device receives a path group identifier (PGID) from MVS. All paths to a device containing the same PGID form a path group. This enables the device to be reserved or assigned on all paths to a single system when any one path is reserved. In addition, the device can disconnect on one path and reconnect on any other path to the system in the same path group.

MVS creates PGIDs every time the system is IPLed. If two systems produce the same PGID, the systems are open to data integrity exposures. To guarantee that each system has a unique PGID, operating systems that share devices with MVS must use a standard method for creating PGIDs.

The PGID consists of eleven bytes of data. They come in two different formats:

- Format 0 PGID, which is created on either a system running on a pre-z990 processor or on a system running on a z990 processor with the z990 compatibility support installed.
- Format 1 PGID, which is created on a system running on a z990 processor with the z990 V1R4 exploitation support or higher installed.

Format 0 PGID: The format 0 PGID consists of eleven bytes of data in the following format:

Bytes 0–1	CPU address, as returned by the STAP instruction
Bytes 2–4	CPU serial number, as returned by STIDP
Bytes 5–6	CPU model number, as returned by STIDP
Bytes 7–10	High-order word of TOD clock

Format 1 PGID: The format 1 PGID created on a z990 processor with z990 exploitation support consists of eleven bytes of data in the following format:

Byte 0	Bits 0 and 1 of byte zero contain the format identifier. A bitstring value of B'00' indicates that the PGID is in the pre-z990 processor format zero, while a bitstring value of B'01' indicates that the PGID is in the z990 processor format. Bits 2 and 3 are reserved for future use. Bit 4 indicates that the VM Hypervisor created the PGID.
Byte 1	CSSID (channel subsystem ID) of the logical channel subsystem assigned to the partition.
Byte 2	LPAR ID
Byte 3–4	CPU serial number
Bytes 5–6	CPU model number, as returned by STIDP
Bytes 7–10	High-order word of TOD clock

Notices

This information was developed for products and services offered in the USA.

IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Station P300
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

- BookManager
- CICS
- Enterprise System/3090
- Enterprise Systems Architecture/370
- ESCON
- ES/3090
- ES/4381
- ES/9000
- IBM
- IBMLink
- IMS
- MVS/ESA
- MVS/SP
- OS/390
- Processor Resource/Systems Manager
- PR/SM
- Resource Link
- Resource Measurement Facility
- RMF
- SP
- Sysplex Timer
- VTAM
- z/Architecture
- z/OS
- z/OS.e

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

NetView is a trademark of IBM Corporation or Tivoli Systems Inc. in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Glossary

This glossary defines technical terms and abbreviations used in documentation. If you do not find the term you are looking for, refer to the index of the appropriate manual or view IBM Glossary of Computing Terms, located at:

<http://www.ibm.com/ibm/terminology>

configuration. A set of hardware units that can support a single operating system.

dual processor. A non-partitionable multiprocessor that has two central processors, each having its own integrated channel paths. That is, each central processor's channel paths work only with that central processor and cannot be accessed by the other central processor.

hardware unit. A central processor, storage element, channel path, device, and so on.

initial storage allocation. The amount of central and expanded storage to be assigned to a logical partition.

Integrated Cryptographic Feature (ICRF). The cryptographic hardware including an inboard key storage unit and TCM. The combination of the Integrated Cryptographic Feature and ICSF/MVS provide secure high speed cryptographic services.

Integrated Cryptographic Service Facility/MVS (ICSF/MVS). A licensed program that runs under MVS/System Product 3.1.3 or higher and provides the access method to the Integrated Cryptographic Feature for programming applications. The combination of the Integrated Cryptographic Feature and ICSF/MVS provide secure high speed cryptographic services.

logical partition. The hardware resources subset that supports an operating system.

LPAR mode. The processor complex mode that is available on the Configuration (CONFIG) frame when the PR/SM feature is installed. LPAR mode allows the operator to allocate the hardware resources of the processor complex among several logical partitions.

master console. The console used for communications between the operator and the operating system.

merging. The configuring of a partitionable processor from physically partitioned mode to single-image mode. This is the reverse of physical partitioning.

multiprocessor (MP). A processor complex that has more than one central processor.

partitioning. The process of forming multiple configurations from one configuration.

physically partitioned (PP) mode. The state of a processor complex when its hardware units are divided into two separate operating configurations or *sides*.

processor complex. The set of hardware units that support one or more operating systems.

reserved storage allocation. The amount of central and expanded storage that you can dynamically configure online or offline to a logical partition.

service processor. The part of a processor complex that provides for the maintenance of the complex and may perform:

- Some or all of the functions associated with operator facilities
- Recovery actions associated with machine-check handling
- Reconfiguration operations

A service processor is also called a "processor controller."

side. One of the configurations formed by physical partitioning.

single-image (SI) mode. The state of a partitionable processor complex when all of its hardware resources are in a single configuration.

system. The interactive combination of a configuration (hardware) and the operating system (software).

system console. The console used by the operator to enter hardware commands and to receive hardware messages.

uniprocessor (UP). A processor complex that has one central processor.

Vector Facility. An optional processing facility to do vector mathematics, available for all 3090 models. There can be one or more Vector Facilities for each processor complex, but only one Vector Facility is associated with each central processor.

Index

Numerics

- 3088 Multisystem Channel Communication Unit
 - global resource serialization 7-3
- 3380 Direct Access Storage
 - Model AA4
 - recovery 9-19
- 3480 Magnetic Tape Subsystem
 - stand-alone dump
 - recovery 5-7
- 3725 Communication Controller
 - configuration example 12-6
- 3880 Storage Control
 - controlling 3380 9-19
- 4381
 - channel subsystem
 - dual processor 12-2
 - dual processor
 - channel subsystem 12-2
- 4381 dual processor
 - channel subsystem
 - configuration 12-2

A

- ABEND parameter
 - Parmlib EXSPATxx member
 - excessive spin recovery 3-5
- accessibility A-1
- ACR (alternate CPU recovery) 9-5
 - excessive spin
 - recovery 3-4, 3-5
- Add Device panel
 - using 12-10
- address instruction trace
 - recovery 3-3
- alternate link
 - global resource serialization 7-7
 - recovery
 - global resource serialization 7-7
- alternate path
 - recovery 9-18
- APPC/MVS (Advanced Program-to-Program Communication/MVS)
 - abend 10-1
 - address space 10-1
 - initialization problem 10-1
 - recovery
 - address space 10-1
 - transaction scheduler 10-2
 - transaction scheduler 10-2
- automation
 - recovery 1-1

B

- binary synchronous communication
 - See* BSC
- blank
 - console screen 8-6
 - recovery
 - console screen 8-6
- BSC (binary synchronous communication)
 - RJP (remote job processing)
 - recovery 5-6
- BSC RJP facility
 - See also* BSC
 - processing
 - recovery 5-6

C

- central storage
 - frame 14-3
 - physical view 13-8
 - reconfiguring 13-14
 - specifying RSU parameter
 - example 13-11
- central storage element
 - configuring online
 - logical partition 15-5
 - reserved
 - configuring online 15-5
- channel path
 - alert condition 9-11
 - CONFIG command
 - logical partition 15-9
 - initialized condition 9-11
 - outstanding reserve 9-9
 - permanent error 9-11
 - reconfiguring 13-14
 - recovery 9-8
 - terminating error 9-9
- channel subsystem
 - 4381 12-2
 - configuration 12-1
 - error 9-8, 9-14
 - missing interrupt 9-12
 - monitoring facility
 - recovery 9-12
- checkpoint data set
 - recovery 5-2
- checkpoint I/O
 - recovery 4-1
- cold start
 - JES3 5-2
- command
 - no system response 3-3
- communication
 - console
 - locked out 3-3

- concurrent maintenance mode
 - recovery 1-4
- CONFIG command
 - configuring logical storage element
 - offline 15-4
 - online 15-4
 - example
 - central reserved storage element 15-5
 - expanded reserved storage element 15-5
 - merging
 - hardware unit 14-1
 - moving channel path 15-9
 - partitioning
 - hardware unit 14-1
 - reconfiguration 13-1
- configuration
 - master console 12-8
- configuration guideline
 - device
 - multiple path 12-2
- CONFIGxx member
 - SYS1.PARMLIB
 - reconfiguration 13-1
- console
 - blank screen 8-6
 - control unit
 - dedicated 12-9
 - disabled loop
 - locked out 3-3
 - error condition 8-1
 - hardware error 8-6
 - locked keyboard 8-6
 - locked out 3-3
 - master console 8-7
 - message backed up 8-2
 - message centered on screen 8-6
 - message in message area 8-2
 - no message 8-2
 - no-console condition 8-7
 - recovery
 - blank screen 8-6
 - error condition 8-1
 - hardware error 8-6
 - locked keyboard 8-6
 - master console 8-7
 - message backed up 8-2
 - message centered on screen 8-6
 - message in message area 8-2
 - no-console condition 8-7
 - system error 8-2
 - system error 8-2
- CONSOLxx member
 - Parmlib
 - MLIM parameter 8-2
 - RLIM parameter 8-2
 - RMAX parameter 8-2
- control unit
 - console
 - dedicated 12-9
 - dedicated
 - console 12-9

- COUPLExx member
 - Parmlib
 - INTERVAL parameter 6-6
- coupling facility
 - reconfiguration 13-15
- CPU
 - error 9-2
 - hard 9-3
 - soft 9-2
 - terminating 9-4
 - hard error 9-3
 - recovery
 - hard error 9-3
 - restart 9-24
 - soft error 9-2
 - terminating error 9-4
 - restart 9-24
 - soft error 9-2
 - terminating error 9-4
- CRW (channel report word)
 - describing error 9-8
- CTC (channel-to-channel)
 - alternate
 - recovery 5-6
 - failure 5-5
 - link 7-5
 - primary
 - recovery 5-6
 - recovery
 - alternate 5-6
 - failure 5-5
 - link 7-5
 - primary 5-6

D

- DASD (direct access storage device)
 - configuration
 - example 12-4, 12-8
 - maintenance 9-20
 - recovery 9-20
- DDR (dynamic device reconfiguration)
 - failing device
 - recovery 9-16
- dedicated control unit
 - console 12-9
- device
 - 4381 12-2
 - configuration 12-1
 - failing
 - recovery 9-16
 - hung
 - recovery 9-15
 - local display
 - configuration example 12-7
 - multiple path
 - configuration guideline 12-2
- Device Support Facilities
 - See ICKDSF
- disability A-1

- disrupted ring
 - global resource serialization
 - recovery 7-1
- documents, licensed xiii
- DSI (dynamic system interchange)
 - JES3
 - recovery 5-3
 - recovery 5-4
- dummy wait
 - recovery 2-2
- dynamic I/O configuration 13-1
- dynamic pathing support
 - array out-of-sync
 - recovery 9-20

E

- enable/disable switch
 - 3380 9-19
- error
 - CPU 9-2
 - storage 9-6
- ES/3090
 - RSU parameter
 - maximum installed central storage 14-2
 - recommended value 14-2
 - storage increment 14-2
 - storage element size 13-10
- ETR (External Time Reference)
 - machine check 9-3
- excessive spin loop 3-4
- excessive spin recovery 3-4
 - Parmlib EXSPATxx member
 - ABEND parameter 3-4, 3-5
 - SPIN parameter 3-4, 3-5
 - TERM parameter 3-4, 3-5
- expanded storage
 - reconfiguring 13-14
- expanded storage element
 - reserved
 - configuring online 15-5
- EXSPATxx member
 - Parmlib
 - ABEND parameter 3-4, 3-5
 - excessive spin recovery 3-4, 3-5
 - persistent excessive spin 3-5
 - SPIN parameter 3-4, 3-5
 - TERM parameter 3-4, 3-5

F

- FEATURE=ALTCTRL parameter
 - Add Device panel 12-10
- FEATURE=SHAREDUP parameter
 - Add Device panel 12-10
- FSS (functional subsystem)
 - failure
 - recovery 5-7

G

- global processor
 - disabling 5-3
- global resource serialization
 - automatic restart
 - example 7-9
 - complex
 - purging system 7-4
 - quiesced system 7-15
 - reactivating quiesced system 7-15
 - recovery 7-1, 7-8
 - system failure 7-8
 - disrupted ring
 - recovery 7-1
 - manual restart
 - example 7-11
 - quiesced system
 - reactivating 7-15
 - restart
 - two-system complex 7-13
 - ring
 - recovery 7-1
 - star
 - recovery 7-1
- GRSCNFxx member
 - Parmlib
 - example 7-11, 7-13
 - REJOIN parameter 7-9, 7-11, 7-13
 - RESTART parameter 7-9, 7-11, 7-13
 - restarting two-system complex 7-13

H

- hang
 - address space
 - recovery 2-3
 - job
 - recovery 2-3
 - recovery 2-1
 - subsystem
 - recovery 2-3
 - system
 - recovery 2-2
- hard error
 - CPU 9-3
 - storage 9-7
- hardware
 - CPU error 9-2
 - problem
 - recovery 9-1
 - recovery 1-1, 9-1
- hardware unit
 - deselecting 13-1
 - logical reconfiguration 13-2
 - merging
 - CONFIG command 14-1
 - partitioning
 - CONFIG command 14-1
 - recovery
 - configuring offline 1-3

- hardware unit *(continued)*
 - recovery *(continued)*
 - configuring online 1-3
- HCD (Hardware Configuration Definition program)
 - using 12-10
- high speed buffer
 - error 9-8
- hot I/O
 - recovery 9-14
- hot start
 - JES3 5-1

I

- I/O configuration
 - availability
 - planning 12-1
 - non-partitionable processor complex
 - device 12-3
 - partitionable 3090
 - single-image mode 12-7
 - planning
 - availability 12-1
- I/O device
 - error 9-12
 - hung
 - recovery 9-15
- I/O interrupt
 - hot I/O 9-14
 - missing 9-12
- ICKDSF (Device Support Facilities) 9-20
- ICRF (Integrated Cryptographic Feature)
 - CPU with ICRF
 - bringing online 13-4
 - reconfiguring 13-3
 - taking offline 13-4
 - removing last ICRF 13-4
- IEASYSxx member
 - Parmlib
 - RSU parameter 15-3
- IECIOSxx member
 - Parmlib
 - I/O device time interval 9-12
- IFRST service language command
 - recovery 1-4
- increment
 - storage 13-9
- indication
 - problem 1-2
- initialization
 - JES2
 - recovery 4-1
 - sysplex
 - recovery 6-1
- interface reset
 - recovery 1-4
- interrupt
 - missing
 - enabled wait 2-2

- INTERVAL parameter
 - Parmlib COUPLExx member
 - recovery 6-6
- intervention
 - operator
 - recovery 1-1
- IOCP (input/output control program)
 - using 12-10
- IOSP (Input/Output Support Processor)
 - concurrent maintenance mode
 - recovery 1-4
- IPL (initial program load)
 - global resource serialization
 - failure 7-5
 - recovery 7-5
 - recovery 1-4
- ISG309W 7-2
- ISG318I 7-2
- ISG323A 7-1
- ISG325I 7-2

J

- JES2 subsystem
 - initialization
 - recovery 4-1
 - recovery 4-1
- JES3 address space
 - restart 5-1
- JES3 global processor
 - restart 5-1
- JES3 subsystem
 - global processor
 - local processor 5-4
 - starting 5-4
 - reconfiguration 14-2
 - recovery 5-1

K

- key
 - storage
 - program 3-2
- keyboard A-1
 - locked
 - console 8-6
 - recovery 8-6

L

- licensed documents xiii
- local processor
 - JES3 global processor
 - starting 5-4
- logical partition 11-1, 15-1
 - Dynamic Storage Reconfiguration
 - defining 15-2
 - example 15-6
 - sysplex
 - high availability 15-1
 - using 15-1

- logical reconfiguration
 - hardware unit 13-2
- LookAt message retrieval tool xiv
- loop
 - disabled
 - identifying 3-3
 - recovery 3-3
 - tracing 3-3
 - enabled
 - identifying 3-1
 - recovery 3-2
 - stopping 3-1
 - recovery 3-1
 - spin
 - recovery 3-4

M

- machine check
 - accompanying data 9-1
 - CPU
 - hard error 9-3
 - soft error 9-2
 - terminating error 9-4
 - ETR (External Time Reference) 9-3
 - Vector Facility error 9-4
- master console
 - configuration 12-8
 - recovery
 - system action 8-9
- message
 - backed up 8-2
 - centered on screen 8-6
 - console message area 8-2
 - lost
 - recovery 8-9
 - recovery
 - backed up 8-2
 - centered on screen 8-6
 - console message area 8-2
 - system console
 - recovery 8-9
- message retrieval tool, LookAt xiv
- missing I/O interrupt
 - recovery 9-12
- MLIM parameter
 - Parmlib CONSOLxx member
 - recovery 8-2
- monitoring facility
 - recovery
 - channel subsystem 9-12
- multiprocessor
 - non-partitionable
 - RSU parameter 13-12
 - RSU parameter
 - recommended value 13-12
 - terminating error
 - CPU 9-4

N

- NJE (network job entry)
 - error
 - recovery 4-3
- no work wait
 - recovery 2-2
- non-partitionable processor complex
 - I/O configuration
 - device 12-3
- Notices C-1
- nucleus initialization
 - spin loop
 - recovery 6-4

O

- operator
 - intervention 1-1
- out-of-sync condition
 - 3380
 - recovery 9-19
 - recovery 9-20
- output writer FSS
 - failure
 - recovery 5-7

P

- Parmlib CONSOLxx member
 - MLIM parameter 8-2
 - recovery
 - MLIM parameter 8-2
 - RLIM parameter 8-2
 - RMAX parameter 8-2
 - RLIM parameter 8-2
 - RMAX parameter 8-2
- Parmlib COUPLExx member
 - INTERVAL parameter
 - recovery 6-6
- Parmlib EXSPATxx member
 - excessive spin recovery
 - ABEND parameter 3-4, 3-5
 - SPIN parameter 3-4, 3-5
 - TERM parameter 3-4, 3-5
 - persistent excessive spin
 - recovery 3-5
- Parmlib GRSCNFxx member
 - REJOIN parameter
 - automatic restart 7-9
 - example 7-9, 7-11, 7-13
 - manual restart 7-11
 - restarting two-system complex 7-13
 - RESTART parameter
 - automatic restart 7-9
 - example 7-9, 7-11, 7-13
 - manual restart 7-11
 - restarting two-system complex 7-13
- Parmlib IECIOSxx member
 - I/O device time interval 9-12

- partition
 - logical 11-1, 15-1
 - physical 11-1
 - reconfiguration 14-1
 - processor complex 11-1
- partitionable processor complex
 - I/O configuration 14-4
- PCE (processor controller element)
 - recovery
 - restart 1-4
 - warm start (IPL) 1-4
- physical reconfiguration
 - hardware unit 13-2
- physically partitioned
 - See PP
- physically partitioned mode
 - processor complex
 - example 14-4
 - merging 14-1
 - RSU parameter
 - recommended value 13-12
- POR (power on reset)
 - recovery 1-4
- PP (physically partitioned)
 - mode
 - partitioning 11-1
- PR/SM (Processor Resource/Systems Manager)
 - Dynamic Storage Reconfiguration 15-1
 - defining logical partition 15-2
 - logical partition 11-1, 15-1
 - example 15-6
 - reconfiguration 15-1
- primary link
 - global resource serialization complex
 - recovery 7-5
- problem
 - cause 1-2
 - indication 1-2
- Process Resource/Systems Manager
 - See PR/SM
- processor complex
 - I/O configuration
 - non-partitionable 12-3
 - partitionable 14-4
 - merging
 - physically partitioned mode 14-1
 - single-image mode 14-1
 - non-partitionable 12-3
 - partitionable 14-4
 - partitioning
 - example 14-4, 14-6
 - physically partitioned mode 14-6
 - physically partitioned mode
 - partitioning 14-1
 - recovery
 - dynamically partitioning 1-3
 - power off/on 1-4
 - statically partitioning 1-4
 - single-image mode
 - example 14-12
 - merging 14-1, 14-12

- program
 - storage key 3-2

Q

- quiesced system
 - reactivating
 - global resource serialization 7-15

R

- reconfiguration 11-1
 - central processor 13-3
 - central storage 13-8, 13-14
 - channel path 13-14
 - coupling facility 13-15
 - expanded storage 13-8, 13-14
 - failing device
 - recovery 9-16
 - hardware unit
 - adding 13-1
 - logical 13-2
 - physical 13-2
 - removing 13-1
 - I/O device 13-15
 - ICRF (Integrated Cryptographic Feature) 13-3
 - JES3 subsystem 14-2
 - last Vector Facility 13-5
 - physical partition 14-1
 - processor type 13-2
 - Vector Facility 13-4
- recovery
 - action 1-3, 9-24
 - approach 1-3
 - automation 1-1
 - CTC
 - alternate 5-6
 - primary 5-6
 - global resource serialization
 - complex 7-1
 - manual restart 7-11
 - ring 7-1
 - goal 1-2
 - hang 2-1
 - hardware 9-1
 - hardware problem 1-1
 - hardware unit
 - configuring offline 1-3
 - configuring online 1-3
 - interface reset 1-4
 - IPL (initial program load) 1-4
 - operating system 9-1
 - operator intervention 1-1
 - PCE (processor controller element)
 - restart 1-4
 - warm start (IPL) 1-4
 - POR (power on reset) 1-4
 - problem
 - hardware 1-1
 - software 1-1

- recovery (*continued*)
 - processor complex
 - dynamically partitioning 1-3
 - power off/on 1-4
 - statically partitioning 1-4
 - restarting CPU 1-3
 - software problem 1-1
 - SYSIML (system initial microprogram load) 1-4
 - wait 2-1
 - written action 1-1
- Recovery
 - I/O
 - hung 9-15
- REJOIN parameter
 - Parmlib GRSCNFxx member
 - automatic restart 7-9
 - example 7-9, 7-11, 7-13
 - manual restart 7-11
 - restarting two-system complex 7-13
- remote job processing
 - See RJP
- restart
 - global resource serialization
 - automatic 7-9
 - manual 7-11
 - two-system complex 7-13
 - JES3 global processor 5-1
 - JES3 local processor 5-2
 - JES3 subsystem
 - after failure 5-1
 - recovery 1-3
- RESTART parameter
 - Parmlib GRSCNFxx member
 - automatic restart 7-9
 - example 7-9, 7-11, 7-13
 - manual restart 7-11
 - restarting two-system complex 7-13
- ring
 - global resource serialization
 - disrupting IPL 7-5
 - recovery 7-1
- RJE (remote job entry)
 - error
 - recovery 4-2
- RJP (remote job processing)
 - recovery 5-6
- RLIM parameter
 - Parmlib CONSOLxx member
 - recovery 8-2
- RMAX parameter
 - Parmlib CONSOLxx member
 - recovery 8-2
- RSU parameter
 - example 13-11
 - logical partition storage
 - defining 15-3
 - Parmlib EASYSxx member
 - defining logical partition storag 15-3
 - recommended value 13-12

S

- service processor
 - damage
 - recovery 9-6
 - permanent failure 9-6
 - temporary failure 9-6
- shared device
 - operator action 9-17
 - recovery 9-17
- shortcut keys A-1
- SI (single-image)
 - mode
 - merging 11-1
- signalling path
 - IPL
 - problem 6-3
 - recovery 6-8
- single-image
 - See SI
- single-image mode
 - processor complex
 - merging 14-1
 - RSU parameter
 - recommended value 13-12
- soft error
 - CPU 9-2
 - storage 9-7
- spin loop 3-1
 - recovery 3-4
 - operator action 3-6
- sysplex
 - recovery 6-4
- SPIN parameter
 - Parmlib EXSPATxx member
 - excessive spin recovery 3-4, 3-5
- spool data set
 - recovery 4-1
- status update missing 6-6
- storage
 - central
 - reconfiguration 13-8
 - configuring storage element
 - CONFIG command 15-5
 - logical partition 15-5
 - online 15-5
 - element
 - configuring offline 15-5
 - error 9-6
 - hard 9-7
 - soft 9-7
 - expanded
 - reconfiguration 13-8
 - storage element
 - reserved central
 - configuring online 15-5
 - reserved expanded
 - configuring online 15-5
 - size
 - ES/3090 processor complex 13-10
- storage increment
 - size 13-9

- storage key
 - program 3-2
- storage sub-increment
 - size 13-9
- sub-increment
 - storage 13-9
- subchannel
 - recovery 9-11
- supervisor state
 - 100% utilization 3-3
- SWAP operator command
 - failing device
 - recovery 9-16
- SYSIML (system initial microprogram load)
 - recovery 1-4
- sysplex
 - failed system 6-6
 - logical partition
 - high availability 15-1
 - problem 6-1
 - recovery
 - failed system 6-6
 - problem 6-1
- Sysplex couple data set
 - recovery 6-11
- Sysplex Timer
 - recovery 6-4, 6-10
- system
 - purging
 - global resource serialization complex 7-4
 - reinitializing 7-4
- system console
 - problem determination mode 8-10
 - recovery
 - system action 8-9
- system failure
 - global resource serialization complex
 - recovery 7-8

T

- tape
 - configuration
 - example 12-5
- TERM parameter
 - Parmlib EXSPATxx member
 - excessive spin recovery 3-4, 3-5
- terminating error
 - CPU 9-4
 - multiprocessor 9-4
- time interval
 - IBM-supplied default value
 - changing 9-12
- trace
 - address instruction trace
 - recovery 3-3
- two-system complex
 - restart 7-13

U

- unconditional reserve
 - recovery 9-18
- uniprocessor
 - RSU parameter
 - recommended value 13-12
- unit record
 - configuration example 12-7

V

- VARY command
 - reconfiguration 13-1
- Vector Facility
 - offline
 - continuing job 9-25
 - reconfiguration 13-4
 - recovery 9-4

W

- wait
 - coded wait state 2-1
 - disabled 2-1
 - dummy 2-2
 - enabled 2-2
 - no work 2-2
 - recovery 2-1
 - coded wait state 2-1
 - disabled 2-1
 - dummy 2-2
 - enabled 2-2
 - no work 2-2
- wait state code
 - 0A3 7-2
- warm start
 - JES3 5-1
- writer
 - failure
 - recovery 5-7
- WTO macro
 - branch entry
 - recovery 12-9
- WTOR macro
 - branch entry
 - recovery 12-9
- WTOR message
 - sysplex
 - recovery 6-4

Readers' Comments — We'd Like to Hear from You

z/OS
MVS Recovery and Reconfiguration Guide

Publication No. SA22-7623-02

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

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

Name

Address

Company or Organization

Phone No.



Cut or Fold
Along Line

Fold and Tape

Please do not staple

Fold and Tape



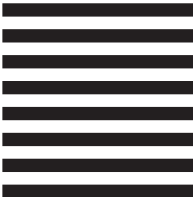
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Department 55JA, Mail Station P384
2455 South Road
Poughkeepsie, NY
12601-5400



Fold and Tape

Please do not staple

Fold and Tape

Cut or Fold
Along Line



Program Number: 5694-A01

Printed in USA

SA22-7623-02

