z/OS

JES2 Migration

z/OS

JES2 Migration

Fourth Edition, September 2002

This is a major revision of GA22-7538-02.

This edition applies to Version 1 Release 4 of z/OS (5694-A01), Version 1 Release 4 of z/OS.e (5694-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: mhvrcfs@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.

Contents

Figures

Tables

About This Document

This document supports z/OS (5694-A01) and z/OS.e(5655-G52).

This document provides information to assist Job Entry Subsystem 2 (JES2) users in migrating to z/OS Version 1 Release 4 JES2. z/OS Version 1 Release 4 JES2 is supported under the HJE7707 FMID.

This document is intended as a high-level overview for use by experienced system programmers responsible for migrating to z/OS Version 1 Release 4 JES2 from earlier releases of JES2. It is divided into two logical sections:

- Chapters 1 and 2 give clear technical descriptions of the enhancements provided by each release of JES2.
- Following chapters, beginning with Chapter 3, describe the migration actions that customers **must** take to ensure an orderly transition to a new level of JES2.

Who Should Use This Document

Readers of this document should have an in-depth knowledge of the configuration and procedures of the current installation as well as the target installation. The system programmer reading this document might need to inform other users (operators, application programmers, other system programmers) about the migration actions they need to take for their migration.

This document works best when considered as a companion document to other documents:

- Read *z/OS and z/OS.e Planning for Installation* to develop an installation plan.
- Use the documentation shipped with your order to install the system.
- Use documents such as *z/OS JES2 Initialization and Tuning Guide* and *z/OS JES2 Initialization and Tuning Reference* to activate new JES2 functions.

How to Use This Document

In this document, the following abbreviations indicate the JES2 component:

Table 1. JES2 Release Abbreviations

Abbreviation	Release Title
JES2 OS/390 V2R5	OS/390 JES2 Version 2 Release 5
JES2 OS/390 V2R7	OS/390 JES2 Version 2 Release 7
JES2 OS/390 V2R8	OS/390 JES2 Version 2 Release 8
JES2 OS/390 V2R10	OS/390 JES2 Version 2 Release 10
JES2 z/OS V1R2	z/OS JES2 Version 1 Release 2
JES2 z/OS V1R4	z/OS JES2 Version 1 Release 4

To **learn about the release** of JES2 that you are installing, read the following chapters:

- Chapter 2, "Migration Roadmap" summarizes compatible and incompatible additions, changes, and deletions for each JES2 release in terms of the following categories:
  - Initialization
  - Installation Modfications
  - Operations
  - Applications
  - Problem Determination and Diagnosis
  - Accounting
  - Migration Actions

Each category for the JES2 release references that section in the Release Overview where specific changes are explained. A **compatible** change does not require action to ensure that functions available before converting to the new release remain available. An **incompatible** change may require changes to installation-written modifications, operating procedures, or installation policy to ensure the continued availability of existing functions. Guidance and reference information about how to implement and use new functions or functions that are compatible with the previous release are discussed in other documents in the JES2 library. Information about how to implement and use new functions that may be incompatible with the previous release is located in the "overview chapters" and the "migration actions" chapters that follow.

To **determine which migration actions you need to take,** select the chapter that applies to your migration path, which begins with Chapter 3. Look for the "migration actions" that your installation must take. Some of the migration actions are required for all installations "passing through" that release. Other migration actions are required only if you are using a function (such as an exit) that has changed.

Read **one** of the following chapters:
- Chapter 8, "Migration Actions: JES2 OS/390 V2R5 to JES2 z/OS V1R4".
- Chapter 7, "Migration Actions: JES2 OS/390 V2R7 to JES2 z/OS V1R4".
- Chapter 6, "Migration Actions: JES2 OS/390 V2R8 to JES2 z/OS V1R4".
- Chapter 5, "Migration Actions: JES2 z/OS V1R1 to JES2 z/OS V1R4".
- Chapter 4, "Migration Actions: JES2 z/OS V1R2 to JES2 z/OS V1R4"

The following appendices also are provided:
- Appendix B, "Sample Exits" lists sample exits for JES2.
- Appendix C, "Split/New/Deleted/Resequenced Parts for JES2" lists modules that have changed in each release.
- Notices lists the notices.
- Chapter 9, "Version Summary Tables" summarizes the new, changed, and deleted interfaces for JES2 releases starting with OS/390 JES2 Version 1 Release 1. The chapter provides charts for changes to initialization statements, commands, macros, mapping macros, and installation exits.

Where to Find More Information

Most licensed documents were declassified in OS/390 V2R4 and are now included on the z/OS Online Library Collection. The remaining licensed documents appear in unencrypted BookManager softcopy and PDF form on the z/OS Licensed Product Library.

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.

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

Using LookAt to look up message explanations

LookAt is an online facility that allows you to 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 where you can access a TSO/E command line (for example, TSO/E prompt, ISPF, z/OS UNIX System Services running OMVS). You can also download code from the *z/OS Collection* (SK3T-4269) and the LookAt Web site that will allow you to access LookAt from a handheld computer (Palm Pilot VIIx suggested).

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 **News** section on the LookAt Web site.

---

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

Some messages have information in more than one document. For those messages, LookAt displays a list of documents in which the message appears.

Additional Information

Additional information about z/OS elements can be found in the following documents.

Title	Order Number	Description
z/OS Introduction and Release Guide	GA22-7502	Describes the contents and benefits of z/OS as well as the planned packaging and delivery of this new product.
z/OS and z/OS.e Planning for Installation	GA22-7504	Contains information that lets users: • Understand the content of z/OS • Plan to get z/OS up and running • Install the code • Take the appropriate migration actions • Test the z/OS system
z/OS Information Roadmap	SA22-7500	Describes the information associated with z/OS including z/OS documents and documents for the participating elements.
z/OS Summary of Message Changes	SA22-7505	Describes the changes to messages for individual elements of z/OS. **Note:** This document is provided in softcopy only on the message bookshelf of the z/OS collection kit.

Determining If a Publication Is Current

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

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

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

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

Summary of changes

This document contains information previously presented in *z/OS JES2 Migration*, GA22-7538-02, which supports z/OS Version 1 Release 3.

This major revision supports z/OS Version 1 Release 4 JES2 (HJE7707) and documents the following:

New information
- Information has been added to indicate this document supports z/OS.e.
- Improved the efficiency of SPOOL I/O using the HASP access method (HAM).
- Improved an installation's ability to recover if checkpoint data is corrupted.
- Eliminated WAITs in the EOM and the EOT SSI code.
- The PSO divorce consists primarily of using the PSO as the queuing element for work rather than the SJB.
- The STAC divorce consists of creating a new block called the STAC and removing indicators and pointers from the SJB.
- Freeing of SJBs will be done in the JES2 address space.
- Addition of JES2 health monitor for the JES2 address space.
- INCLUDE Initialization statement enhancements to read from MVS logical PARMLIB concatenation.
- Support for the // XMIT JCL statement.
- Support zero JES2 mode initiators.
- SPOOL I/O SSI can return in-storage buffers.
- JES2 no longer controls duplicate TSO logons.

Changed information
- Better WLM initiator work load balancing.

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

This document includes 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.

Summary of changes
for GA22-7538-02
z/OS Version 1 Release 3

This document contains information previously presented in *z/OS JES2 Migration*, GA22-7538-01, which supports z/OS Version 1 Release 2.

This major revision supports z/OS Version 1 Release 3 JES2 (HJE7705) and documents the following:

New information

- A new appendix, Appendix A, "An Example of ACTIVATING Your System" on page 147 gives a description of how a JES2 installation uses the $ACTIVATE command in migrating from one JES2 release to the next.

Changed information

- Section, "Provide for JES2 IPCS Support"

  This section is added to the Problem Determination and Diagnosis Topic in each Migration Actions Chapter of this publication.

- Several updates were made to indicate that beginning with z/OS Version 1 Release 2 JES2, the command migration aid exit routine, HASX5CTR in module HASX05C for HJE7705, is no longer automatically installed and enabled.

**Summary of changes
for GA22-7538-01
z/OS Version 1 Release 2**

This document contains information previously presented in *z/OS JES2 Migration*, GA22-7538-00, which supports z/OS Version 1 Release 1.

This major revision supports z/OS Version 1 Release 2 JES2 and documents the following:

New information

- **Checkpoint Improvements**

  Switching between checkpoint modes does not now require an all member warm start.

- **Constraint Relief**

  The number of JQEs, JOEs, BERTs, and TG space that you can define in the system has been increased

- **Large Volume Support**

  A new interface reads records from JES2 SPOOL. This interface is an option on SSI 71.

- **Dynamic JES2 PROCLIB and INCLUDE Initialization Statement**

  Information in the JES2 PROC is minimized (thus limiting the chance that some change in the system makes the PROC unusable) and there are new ways to expand your initialization stream.

- **Job Numbers to a maximum of 999,999**

  JES2 allows you to define up to 999,999 job numbers surpassing the older limit of 65,534.

- **Spinning JESlog Data Sets for Long Running Jobs**

  JES2 allows installations to "spin off" or "suppress" JESlog data sets as a job runs.

- **$ACTIVATE Command for z2 Mode**

  You must issue the $ACTIVATE command with LEVEL=z2 to use the support for maximizing job numbers to 999,999.

Changed information

- **Warm Start Redesign**

  JES2 z/OS V1R2 improves warm start processing that in the past would cause system malfunctions (hangs, repeated ABENDs, and other erroneous situations) when a shortage of BERTs existed.

- **$HASP443 Message Enhancement**

  The HASP443 message is updated to better clarify the reason for a SPOOL allocation failure.

- **Exits 14 and 49**

  An indicator has been provided that controls class list optimization.

- **Macro Changes**

  Various macros have changed, such as $XMPOST, $QJQE, $#JOE, $DOGJQE, $JOEJNUM, $JBIDBLD, $QLOCNXT.

- **APPLCOPY**

  Support for APPLCOPY is eliminated

Chapter 1. Migration Overview

Your plan for migrating to the new level of JES2 should include information from a variety of sources. These sources of information describe topics such as coexistence, service, hardware and software requirements, installation and migration procedures, and interface changes.

The following documentation, which is supplied with your product order, provides information about installing your z/OS system. In addition to specific information about JES2, this documentation contains information about all of the z/OS elements.

- *z/OS and z/OS.e Planning for Installation*

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

- *z/OS Program Directory*

  This document, which is provided with your z/OS product order, leads you through the specific installation steps for JES2 and the other z/OS elements.

- *ServerPac Installing Your Order*

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

Within this publication, you can find information about the specific updates and considerations that apply to this release of JES2.

This publication documents z/OS Version 1 Release 4 JES2 at this level and describes the JES2 migration paths that culminate at this target.

Up to "four" consecutive JES2 releases can run on the z/OS Version 1 Release 4 BCP:

- z/OS Version 1 Release 4 (HJE7707)
- z/OS JES2 Version 1 Release 2 (HJE7705)
- OS/390 JES2 Version 2 Release 10 (HJE7703)
- OS/390 JES2 Version 2 Release 8 (HJE6608)

While the "lower level" JES2 releases can run on z/OS Version 1 Release 4 BCP, they can also run on "lower level" OS/390 or z/OS BCP releases. z/OS JES2 Version 1 Release 4, however, can only run on z/OS Version 1 Release 4 BCP.

This publication has the following structure:

Table 2. Publication Layout

Topic	Page
Chapter 2, "Migration Roadmap"	9

Table 2. Publication Layout (continued)

Topic	Page
This section identifies the JES2 releases and itemizes the content of each release. Specific content descriptions of each release is referenced in the "release overview" section of the publication. These release overviews help you to assess each JES2 release with respect to your current level of JES2.	
The migration path you choose is defined as your current JES2 release and the target JES2 release you choose to migrate to. A "Migration Actions" section of each JES2 product roadmap references the migration path description in this publication that applies. For example, if your present OS/390 JES2 level is Version 2 Release 7 and you are reading this JES2 z/OS V1R4 level of the publication, you are planning a migration from Version 2 Release 7 to JES2 z/OS V1R4. This migration path defines how you use this publication: • Read "JES2 z/OS V1R4 Summary" on page 10, which references "z/OS JES2 Version 1 Release 4 Overview" on page 15 and "OS/390 JES2 Version 2 Release 7 Overview" on page 34. • Read "z/OS JES2 Version 1 Release 4 Overview" on page 15, which describes the changes to JES2 z/OS V1R4. • Read Chapter 7, "Migration Actions: JES2 OS/390 V2R7 to JES2 z/OS V1R4" on page 87, which specifically describes the migration actions you'll need to consider when you do the migration to JES2 z/OS V1R4.	
Chapter 3, "Release Overviews"	15
This chapter describes the specific updates that were made to JES2 for each release and provides an overview of the changes, a description of any migration and coexistence tasks that may be considered, and where you can find more detailed information.	
Chapter 9, "Version Summary Tables"	129
This chapter provides a summary of the changes that are made to JES2 user and programming interfaces.	

Terms You Need to Know

This section describes some terms you may need to know as you use this book.

Base Release The JES2 release you are currently running.

Target Release The JES2 release you are choosing to migrate to.

Compatible Change A compatible change does not require action to ensure that functions available before converting to the new release remain available.

Incompatible Change An incompatible change may require changes to installation-written modifications, operating procedures, or installation policy to ensure the continued availability of existing functions.

Migration Activities that relate to the installation of a new version or release of a program to replace an earlier level. Completion of these activities ensures that the applications and resources on your system will function correctly at the new level.

Coexistence Two or more systems at different levels (for example, software, service or operational levels) that share resources. Coexistence includes the ability of a system to respond in the following ways to a new function that was introduced on

another system with which it shares resources: ignore a new function, terminate gracefully, support a new function. The following are examples of multisystem configurations in which resource sharing can occur:

- A single system running multiple LPARs
- A single processor that is time-sliced to run different levels of the system (for example, during different times of the day)
- Two or more systems running separate processors
- A Parallel Sysplex configuration (also includes a basic sysplex)

Multi-Access Spool (MAS)   Multiple systems sharing the JES2 input, job, and output queues.

Member of MAS   A JES2 instance of a MVS system.

Developing a Migration Strategy

The recommended steps for migrating to a new release of JES2 are:

1. Become familiar with the supporting migration and installation documentation for the target release.

   You should determine what updates are needed for products that are supplied by IBM, system libraries, and non-IBM products. Review *z/OS and z/OS.e Planning for Installation* and *z/OS Introduction and Release Guide* for information about JES2 and other z/OS elements.

2. Develop a migration plan for your installation.

   When planning to migrate to a new release of JES2, you must consider high-level support requirements, such as machine and programming restrictions, migration paths, program compatibility, and MAS considerations.

3. Obtain and install any required program temporary fixes (PTFs) or updated versions of the operating system to all members of the MAS to ensure functional compatibility and coexistence.

   It is important that all those PTFs required on all OTHER members of the MAS be installed "before" a member with the new JES2 release is run in the MAS.

   Call the IBM Software Support Center to obtain the preventive service planning (PSP) upgrade for JES2, which provides the most current information about PTFs for JES2. Check RETAIN again just before testing JES2. For information about how to request the PSP upgrade, refer to z/OS Program Directory. Although z/OS Program Directory contains a list of the required PTFs, the most current information is available from the IBM Software Support Center.

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

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

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

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

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

8. Exercise the new functions now available on this member.

9. Repeat steps #4 thru #8 on the other members of the MAS.

Reviewing changes to JES2 processing

As you define your installation's migration plan, consider how the new and changed JES2 support might affect the following areas of JES2 processing. In Chapter 3, "Release Overviews" on page 15, you should review the specific release overview that is the target of your migration path. Then review the "Migration Actions" sections for your migration path to determine how, or if, the support affects the tasks that are performed at your installation.

Initialization

You must be aware of how changes introduced by a new product release can affect JES2 initialization. Changes in initialization can affect real and virtual storage requirements, performance, security, and integrity, as well as automatic command processing and networking.

Installation Modifications

To meet the specific requirements of your installation, you can customize JES2 functions to take advantage of new support after the product is installed. For example, you can tailor JES2 through the use of installation exit routines, class descriptor table (CDT) entries, or options to improve performance. This book lists changes to JES2 that might require your installation to tailor the product, either to ensure that JES2 runs as before or to accommodate new security controls that your installation may need.

Application development

Application development programmers must be aware of new functions introduced in a new release of JES2. To ensure that existing programs run as before, your application programmers need to know about any changes in data areas and processing requirements. This book provides an overview of the changes that might affect existing application programs.

Operations

The new JES2 release might introduce changes to its operating characteristics, such as changed commands, new or changed messages, or in the methods of implementing new functions. This book identifies those changes for which you should provide user education before running this release of the product.

Problem Determination and Diagnosis

Each release of JES2 introduces changes that affect the way you perform problem determination and diagnose problems. Some of these changes include new traces and logs, modified and new messages, and other diagnositic information that you would need in solving and fixing system problems.

Reviewing changes to JES2 interfaces

When defining your installation's migration plan, also consider that JES2 interfaces may also be affected by the new or changed functions that are introduced in this release. These interfaces include:

- Initialization Statements
- Commands
- Installation Exits
- Macros
- Mapping Macros
- SSI Function Codes
- Messages

- Panels
- SMF Records
- SYS1.SAMPLIB members
- Utilities

Chapter 2, "Migration Roadmap" on page 9 provides a "roadmap" for your migration. This roadmap itemizes the changes to each release of JES2, references more specific overviews of that release, and references specific migration actions you'll need to consider when migrating from your base release to the target release represented by this publication. Chapter 9, "Version Summary Tables" on page 129 provides a summary of the changes that affect these interfaces for the release.

Common Migration Activities

General migration actions for JES2 include examining areas of processing in JES2 that involve:

- Customization and installation exits
- Macros
- Initialization statements and processing
- Operations and commands
- Applications
- Problem determination and diagnosis
- Installation modifications

Release Migrations

Migration to the JES2 component of z/OS is based on the current coexistence policy of four consecutive releases. When you migrate from JES2 z/OS V1R2 (or from any subsequent release), the release you migrate to is required to be within "four" consecutive releases to be fully supported. That is, migration forward and backward should be made within "four" consecutive releases. This applies to release migrations for:

- Single system configurations
- Individual systems within a multisystem configuration
- Cases where a simultaneous IPL is used to migrate all systems in a multisystem configuration at the same time.

You should take this into account to ensure that you are appropriately positioned for future software migrations. Since each release can normally be ordered for only a six month window, it is very important that you order the required releases while they are available.

Coexistence

z/OS give you optimum compatibility and flexibility as you migrate systems in a multisystem configuration by allowing up to "four" consecutive releases to coexist. Coexistence considerations apply to JES2 multisystem configuration in which there is resource sharing (for example, MAS members reading the JES2 checkpoint data set). This includes nonParallel Sysplex and Parallel Sysplex multisystem configurations.

The JES2 component of z/OS allows "four" consecutive FMIDs to coexist. For JES2, coexist means that each JES2 FMID can be a member in a MAS and read the same JES2 checkpoint data set. Since not every consecutive JES2 FMID matches exactly a z/OS or OS/390 release, you can have more than "four" consecutive z/OS releases

coexist for the "four" JES2 FMIDs. Table 3 summarizes the correspondence between JES2 FMIDs and z/OS or OS/390 releases.

Coexistence allows systems within a multisystem configuration to be upgraded to a new release level of the operating system one system at a time. This is contingent on the fact that the release you are migrating to is within three releases of the lowest release running in your multisystem configuration (for a total of up to "four" consecutive releases). "Four" releases is the general migration and coexistence policy that you should assume for future z/OS releases, except where special provisions have been provided.

This publication treats OS/390 Version 2 Release 10 and z/OS Version 1 Release 1 as a single coexistence level, rather than two coexistence levels. This is due to the unique characteristics of z/OS Version 1 Release 1 and z/OS Version 1 Release 1 Upgrade Package for OS/390 Version 2 Release 10, which can be installed on OS/390 Version 2 Release 10.

Note: This special treatment only applies to OS/390 Version 2 Release 10 and z/OS Version 1 Release 1. This upgraded package, referred to in this publication as (z/OS V1R1 & OS/390 V2R10) or z/OS Version 1 Release 1, is base upon which z/OS Version 1 Release 4 is built. z/OS Version 1 Release 4 is the <u>target</u> of all migration paths described in this publication.

A summary of this coexistence is as follows:

Table 3. JES2 FMIDs and z/OS Releases

JES2 FMID	z/OS Release
HJE6605 (*)	OS/390 V2R5
	OS/390 V2R6
HJE6607(*)	OS/390 V2R7
HJE6608	OS/390 V2R8
	OS/390 V2R9
HJE7703	OS/390 V2R10
	z/OS V1R1
HJE7705	z/OS V1R2
	z/OS V1R3
HJE7707	z/OS V1R4

Note:

* JES2 (HJE6607) and JES2 z/OS V1R4 (HJE7707) cannot coexist in the same MAS. However, JES2 (HJE6607) can process the JES2 z/OS V1R4 (HJE7707) checkpoint data set running as a member of a MAS that has no member later than HJE7705. Once you have activated to **z2 Mode** for JES2 z/OS V1R4, your JES2 releases earlier than JES2 z/OS V1R2 can no longer use the **z2 Mode** checkpoint data set. See "z2 Mode for $ACTIVATE" on page 17, which describes activating to **z2 Mode**.

Service

The following PTFs are required for prior levels of JES2 to run with z/OS Version 1 Release 4 JES2 in a multi-access spool (MAS) configuration.

Table 4. Migration PTFs for z/OS

Target z/OS Release	Origin JES2 Release (and FMID)	Required PTF(s)
z/OS V1R4	OS/390 JES2 V2R8 (HJE6608)	UW87739
	OS/390 JES2 V2R10 (HJE7703)	UW87740
	z/OS JES2 V1R2 (HJE7705)	UW87741

Year 2000 Support for JES2

JES2 is an element of z/OS and is Year 2000 ready. When used in accordance with its associated documentation, JES2 is capable of correctly processing, providing, and/or receiving data within and between the twentieth and twenty-first centuries, provided that all products (for example, server, software, and firmware) used with JES2 properly exchange accurate data with it.

For more information about Year 2000 support, refer to *The Year 2000 and 2-Digit Dates: Guide*, GC28-1251.

Chapter 2. Migration Roadmap

This section describes the migration paths that are supported by the current release of JES2. It also provides references to release-specific information for your present release and your target release. Migration paths for JES2 are as follows:

- z/OS JES2 Version 1 Release 2 (HJE7705) to JES2 z/OS V1R4
- z/OS JES2 Version 1 Release 1 (HJE7703) to JES2 z/OS V1R4
- OS/390 JES2 Version 2 Release 8 (HJE6608) to JES2 z/OS V1R4
- OS/390 JES2 Version 2 Release 7 (HJE6607) to JES2 z/OS V1R4

What follows is a summary of each release, references to the release overview, and reference to migration actions that will take you from your base release to the target release (JES2 z/OS V1R4) that represents the highest release level for this publication.

Note to Users

This section is designed to show the migration paths that are supported to the current release of JES2. For each release, this section also summarizes the main functional items. Functional items are grouped into the following categories:

- Initialization
- Installation Modifications
- Operations
- Applications
- Problem Determination and Diagnosis
- Accounting
- Migration Actions

The ″Migration Actions″ category is a reference to a specific chapter in this publication that describes the appropriate actions that are needed to migrate to the target release of JES2.

9

JES2 z/OS V1R4 Summary

This section describes additions, changes, and deletions to JES2 introduced by JES2 z/OS V1R4.

Table 5. JES2 z/OS V1R4 Release Summary

Migration Topic	Page
Initialization	
INCLUDE initialization statement	15
Support zero JES2 mode initiators	15
Support // XMIT JCL statement	15
Installation Modfications	
Checkpoint Data Error Recovery	16
Operations	
SPOOL I/O SSI enhancement	15
Spool I/O Performance Improvements	16
End of Memory Restructure	16
WLM workload balancing	16
JES2 no longer controls duplicate logons	16
Applications	
None	None
Problem Determination and Diagnosis	
JES2 Health Monitor	16
Accounting	
None	None
Migration Actions	
Not Applicable	

JES2 z/OS V1R2 Summary

This section describes additions, changes, and deletions to JES2 introduced by JES2 z/OS V1R2.

Table 6. JES2 z/OS V1R2 Release Summary

Migration Topic	Page
Initialization	
Support for a Maximum of 999,999 Job Numbers	18
Constraint Relief	18
Dynamic JES2 PROCLIB and INCLUDE Initialization Statement	26
Installation Modfications	
JES2 Control Block Field Changes	19
Macro Updates Relating to JOE and JQE Changes	25
Other New and Changed Macros	29
Command Syntax Migration Aid Removal (Exit 5)	28
Exit 14 and Exit 49	28
Operations	
z2 Mode for $ACTIVATE	17
JESNEWS	25
Large Volume Support	25
Spinning JESlog Data Sets for Long Running Jobs	27
Warm Start Redesign	29
Applications	
APPLCOPY Application	29
Problem Determination and Diagnosis	
Message HASP443 - Updated	28
Accounting	
Counting of Null Records	29
Migration Actions	
Not Applicable	

JES2 z/OS V1R1 Summary

This section describes additions, changes, and deletions to JES2 introduced by JES2 z/OS V1R1.

Table 7. JES2 z/OS V1R1 Release Summary

Migration Topic	Page
Initialization	
$ACTIVATE	29
SPOOLDEF FENCE= Changes	30
TPDEF SNABUF Changes	31
TRACEDEF PAGES= Changes	32
BLOB Size	31
SPOOLDEF TGBPERVL= Dropped	30
ZAPJOB Initialization Statement	32
Installation Modfications	
Exit 11 and Exit 12	31
New and Changed Macros	31
Operations	
Spool Affinity	30
$HASP834 Deleted	32
$ZAPJOB Command	32
Applications	
Spool Data Set Browse Documented	32
Active DCTs	32
Problem Determination and Diagnosis	
Multi-System Dumps	32
Accounting	
None	
Migration Actions	
Chapter 5, "Migration Actions: JES2 z/OS V1R1 to JES2 z/OS V1R4"	45

JES2 OS/390 Version 2 Release 8 Summary

This section describes additions, changes, and deletions to JES2 introduced by OS/390 JES2 Release 8.

Table 8. JES2 OS/390 V2R8 Release Summary

Migration Topic	Page
Initialization	
None	
Installation Modfications	
CF System Managed Rebuild Support	33
Operations	
None	
Applications	
None	
Problem Determination and Diagnosis	
Message HASP069 - Updated	33
Message HASP263 - Updated	33
Message HASP292 - Updated	33
Message HASP538 - Updated	33
JES2 Dispatcher Rolling Trace - New	34
Accounting	
None	
Migration Actions	
Chapter 6, "Migration Actions: JES2 OS/390 V2R8 to JES2 z/OS V1R4"	65

JES2 OS/390 Version 2 Release 7 Summary

This section describes additions, changes, and deletions to JES2 introduced by OS/390 JES2 Release 7.

Table 9. JES2 OS/390 V2R7 Release Summary

Migration Topic	Page
Initialization	
None	
Installation Modfications	
Changes to Exit 8	34
Changes to Exit 25	34
Changes to Exit 46 and Exit 47 for NJE Network Processing	34
Fiber Channel Support	35
Checkpoint Performance	35
Operations	
Changes that Affect JES2 Operations	35
Applications	
None	

Table 9. JES2 OS/390 V2R7 Release Summary (continued)

Migration Topic	Page
Problem Determination and Diagnosis	
New Record for Trace ID 17	35
$DPCE Command - Additional Diagnostic Information	35
Message HASP291 Enhanced	36
Message HASP064 - New Message	36
$CBIO Services - Additional Error Recording	36
Accounting	
None	
Migration Actions	
Chapter 7, "Migration Actions: JES2 OS/390 V2R7 to JES2 z/OS V1R4"	87

JES2 OS/390 Version 2 Release 5 Summary

This section describes additions, changes, and deletions to JES2 introduced by OS/390 JES2 Release 5.

Table 10. JES2 OS/390 V2R5 Release Summary

Migration Topic	Page
Initialization	
None	
Installation Modfications	
None	
Operations	
None	
Applications	
JES2-Provided Client Print Services	36
Problem Determination and Diagnosis	
None	
Accounting	
None	
Migration Actions	
Chapter 8, "Migration Actions: JES2 OS/390 V2R5 to JES2 z/OS V1R4"	105

Chapter 3. Release Overviews

For descriptions of the functional enhancements provided by recent JES2 releases, read the following:

- "z/OS JES2 Version 1 Release 4 Overview" below
- "z/OS JES2 Version 1 Release 2 Overview" on page 17
- "z/OS JES2 Version 1 Release 1 Overview" on page 29
- "OS/390 JES2 Version 2 Release 8 Overview" on page 33
- "OS/390 JES2 Version 2 Release 7 Overview" on page 34
- "OS/390 JES2 Version 2 Release 5 Overview" on page 36

For each release the functional enhancements are listed, then summarized. The summaries can help you in your planning. For example, after reading about a new or changed function, you may choose to exploit that function if your installation can benefit from its use.

The summaries briefly describe changes made in each release. Where appropriate, each summary briefly explains the purpose of the affected function, how the function worked in the previous release, how the function has changed, and how to determine the significance of the change to your installation.

Some of the summaries in this chapter may require certain actions on the part of the installation to achieve or maintain full function. Specific migration instructions are provided in the remaining chapters of this book.

z/OS JES2 Version 1 Release 4 Overview

INCLUDE initialization statement

The enhancements include:

- Provide ability to specify reading from the MVS logical parmlib concatenation.
- Provide ability to read from another member of the current data set without specifying data set name.
- Support ability to have a default logical parmlib member name used when starting JES2 instead of supplying a HASPPARM DD statement.

See *z/OS JES2 Initialization and Tuning Guide* for more information.

Support // XMIT JCL statement

This statement specifies how to transmit jobs to another node.

Support zero JES2 mode initiators

INITDEF PARTNUM=0 now supported. When PARTNUM=0 is specified, no $PIT is created.

SPOOL I/O SSI enhancement

The SPOOL I/O SSI can be used to read in storage buffers.

Spool I/O Performance Improvements

JES2 z/OS V1R4 supports improved efficiency of SPOOL I/O using the HASP access method (HAM). This includes reads and writes of SYSOUT/SYSIN data sets but does not include any control block I/O.

- There will be 2 ways of reading data from SPOOL, the older single record reads for non-track celled data and standard track cell reads for track celled data.
- SVC 111 is now used only for internal reader (INTRDR) processing.

JES2 no longer controls duplicate logons

If the SYSIKJUA ENQ has not been updated to be a SYSTEMS ENQ, a given TSO userid can logon multiple times in the JES2 MAS. This is because JES2 no longer checks for duplicate userids. To revert to pre-z/OS V1R4 behavior, use the supplied JES2 sample exit for exit 44. Using this exit may result in a delayed re-logon for a given userid with its accompanying mis-descriptive error message.

WLM workload balancing

Sampling data is extended to give WLM the ability to perform workload balancing.

Checkpoint Data Error Recovery

JES2 z/OS V1R4 provides support for an installation's ability to recover if checkpoint data is corrupted, whether due to an installation error, such as trying to start JES2 using the wrong checkpoint data set, or some other form of corruption.

If a sufficient number of errors are found during initialization, HASP863 is issued to ask the operator whether JES2 initialization should continue.

End of Memory Restructure

In JES2 z/OS V1R4 supports reduced bottlenecks and WAIT/POST interfaces in MVS EOM processing.

- PSO — The PSO divorce will consist primarily of using the PSO as the queuing element for work rather than the SJB (which in turn has the address of the PSO).
- STAC (Status Cancel) — The STAC divorce will consist of creating a new block called the STAC and removing indicators and pointers from the SJB.
- PSO and STAC — PSO and STAC elements are contained in unique data spaces
- SJB — Freeing of SJBs will be done in the JES2 address space. This includes the ENQ which serializes the removal with the JOBLOG processing for Branch Entry WTOs.
- HETSOUT — HETSOUT is called to cleanup PSO and STATUS/CANCEL work elements still pending. There is supporting code in the main task which deals with the pending elements.
- Main Task Processing — At EOM, SJBs will be placed on a new End of Memory SJB queue (CCTJTEOM) and removed from HASB/HAVT queue.

JES2 Health Monitor

JES2 z/OS V1R4 JES2 z/Os V1R4 provides support for a monitor for JES2 to diagnose non-responsive main-task situations. This monitor will run in a separate address space (JES2MON) and wake up on a regular basis via a timer. The monitor supports operator commands to display diagnostic information about the JES2 address space as well as status information for the monitor itself and history information about resource utilization. The monitor also issues messages when it discovers anomalies in the JES2 main task.

Please note that all monitor messages are of the form HASP9nnn and all monitor commands start with $J.

z/OS JES2 Version 1 Release 2 Overview

z2 Mode for $ACTIVATE

A new level of $ACTIVATE is available in JES2 z/OS V1R2. This activation level, called **z2 Mode** supports JES2's ability to process increases in the following:

- Job numbers to 999,999
- JQEs to a maximum of 200,000
- JOEs to a maximum of 500,000
- BERTs to a maximum of 500,000
- TGSPACE to a maximum of 16,581,181 bytes

All members of the MAS must be at the JES2 z/OS V1R2 level or higher when the $ACTIVATE to **z2 Mode** is performed. The $ACTIVATE command has a required subparameter in which you specify the level you intend to activate. For example, you would specify $ACTIVATE,LEVEL=z2 to activate your MAS to **z2 Mode**.

Once running in **z2 Mode**, no level of JES2 prior to JES2 z/OS V1R2 is allowed to join the MAS nor is allowed to start using the JES2 z/OS V1R2 checkpoint. At cold start, JES2 starts in **z2 Mode** unless PARM=(UNACT,COLD) is specified. To revert back to **R4 Mode**, you must either issue $ACTIVATE,LEVEL=R4 or perform an all member warm start or all member hot start with the UNACT parameter (START JES2,PARM=UNACT).

Note: In **R4 Mode**, JES2 operates as in JES2 z/OS V1R1.

See *z/OS JES2 Commands* to use the $ACTIVATE command.

JES2 z/OS V1R2 must run at least in **R4 Mode**. You do not need to $ACTIVATE to **z2 Mode**. You gain all the **compatible** functional enhancements of JES2 z/OS V1R2 except the following processing increases:

- Job numbers to 999,999
- JQEs to a maximum of 200,000
- JOEs to a maximum of 500,000
- BERTs to a maximum of 500,000
- TGSPACE to a maximum of 16,581,181 bytes

(These are the increases you get in **z2 Mode**.)

z/OS JES2 Version 1 Release 2 increases the potential number of job queue elements (JQEs) and output elements (JOEs) above the previous limits. All shared data areas (SPOOL and checkpoint) used by JES2 are compatible with previous releases of JES2. However, the data structures in **R4 Mode** cannot meet the increased limits. Many binary job number fields are only 2 bytes long and chaining fields use 3 byte offsets which cannot address the new limits. **z2 Mode** binary job number fields are 4 bytes long and chaining is accomplished using 3 byte indexes rather than 3 byte offsets. These changes are incompatible with any exit routine or other unique code that examines checkpointed control blocks or processes job numbers. You may need to update these routines to support **z2 Mode**.

You can use the $ACTIVATE,LEVEL=R4 command to revert to **R4 Mode** of the checkpoint, and your installation does NOT have to do a restart to accomplish this. Job number limits and checkpoint data become compatible with JES2 z/OS V1R1.

Operator commands that process jobs by job number are improved to handle the processing increases in **z2 Mode**.

$T JOBDEF,RANGE= is updated to allow you to set a job number limit up to a maximum of 999,999. Also, the following $T command changes allow you to take advantage of new maximums for the number of jobs, number of JOEs and number of BERTs as well as to decrease their settings:
- $T JOBDEF JOBNUM= for the number of jobs
- $T OUTDEF JOENUM= for the number of JOEs
- $T CKPTSPACE BERTNUM= for the number of BERTs

See *z/OS JES2 Commands* for using these commands.

Constraint Relief

JES2 z/OS V1R2 provides constraint relief by increasing the number of JQEs, JOEs, BERTs, and TG space that you can define in the system.

z2 Mode allows you to use more JQE, JOE and BERT control blocks and define more track groups as well (not just more job numbers). In z2 mode, there can be up to 200,000 JQEs defined, 500,000 JOEs defined, 500,000 BERTs defined and SPOOLDEF TGSPACE=MAX= can be set to 16,581,184.

R4 Mode retains job number and control block number limits as in JES2 z/OS V1R1 checkpoint processing.

Support for a Maximum of 999,999 Job Numbers

In **z2 Mode**, JES2 allows you to define up to 999,999 job numbers (expanding the limit from 65534).

The JOBDEF initialization statement is changed to expand the amount of job numbers you can define. With a new maximum initialized, JES2 interprets and formats job numbers as follows:
- If the job number is > 99,999, then the jobid is Jxxxxxxx
- If the job number is <= 99,999 then the jobid is JOBxxxxx
- If the job number is > 99,999, then the jobid is Txxxxxxx
- If the job number is <= 99,999 then the jobid is TSUxxxxx
- If the job number is > 99,999, then the jobid is Sxxxxxxx
- If the job number is <= 99,999 then the jobid is STCxxxxx

Note: JES2 uses the new job number formats in displays of job information.

JES2 uses the appropriate jobid format in displays of job information. There is a transition period after JOBDEF RANGE= is increased above 99,999 where the "old" format jobids may still be displayed. This can occur for jobs that were created when the old range was being used. Until these jobs are purged from the system, the old format for jobids can be displayed as part of informational messages (and SMF records) about those specific jobs. See *z/OS JES2 Initialization and Tuning Reference* for the specific changes to the JOBDEF initialization statement

JES2 Control Block Field Changes

General Field and Control Block Changes

Getting JES2 to use up to 999,999 job numbers requires that you "ACTIVATE" it to **z2 Mode**. Unless this activation is performed, JES2 operates in **R4 Mode** with the job number limitiations as follows:

- JOBDEF JOBNUM max = 65534 ($MAXJQE_R4)
- JOBDEF RANGE max = 65534 ($MAXJNM_R4))
- OUTDEF JOENUM max = 161314 ($MAXJOE_R4)
- CKPTSPACE BERTNUM max = 262136 (4*$MAXJQE_R4)
- SPOOLDEF TGSPACE=MAX, max= 9,935,680

Note: R4 Mode EQUATE names are in parentheses above.

After you have performed an ACTIVATE to **z2 Mode**, the new limits become:
- JOBDEF JOBNUM max = 200,000 ($MAXNJQE)
- JOBDEF RANGE max = 999,999 ($MAXJNUM)
- OUTDEF JOENUM max = 500,000 (JOTMXJOE)
- CKPTSPACE BERTNUM max = 500,000 (4*$MAXBERT)
- SPOOLDEF TGSPACE=MAX, max= 16,581,184 ($MAXTGS)

Note: z2 Mode EQUATE names are in parentheses above.

Activation to **z2 Mode** changes the three byte offset pointers for JQE chains, JOE chains, and JOE to JQE chains to indexes, and job numbers are stored in four bytes instead of two bytes. JES2 z/OS V1R2 in **z2 Mode** forces an **incompatible** change to your JES2 system.

Changes to control blocks and data areas that support increasing job numbers to 999,999 focus on increasing the number of JQEs and JOEs above current limits.
- JQE and JOE chaining fields are changed from three byte offsets to three byte indexes in the checkpoint data set.
- Processing the job queue index (JIX) to convert job numbers is changed from using two bytes entries to a hash table.
- A new JQT CTENT can now hold 20,000 JQE extensions. JQE extensions are used to track spool usage for executing jobs.
- The format of the control information for the checkpoint data set limited the amount of data tha could be stored in the data set. The format of the control information is updated to allow a larger data set.
- Job number fields in control blocks have increased their length from two bytes to four bytes.

The implications of changing JES2 supporting a maximum of 999,999 job numbers manifests itself in the jobid format. Based on the upper limit specified for JOBDEF RANGE=, jobid formats can exist as follows:
- JOBnnnnn format is used for upper limit less than or equal to 99,999.
- J0nnnnnn format is used for upper limit greater than 99,999.
- STCnnnnn becomes S0nnnnnn for a limit greater than 99,999.
- TSUnnnnn becomes S0nnnnnn for a limit greater than 99,999.
- NJE jobs can use the new jobid format (local jobs cannot) when the original job number is greater than 99,999, the job number is available, and JOBDEF

RASSIGN=YES is specified. You should specify JOBDEF RASSIGN=NO if, for jobs from NJE nodes, you want to use job numbers in the local range.

All member of your MAS must be at JES2 z/OS V1R2 or higher to activate to **z2 Mode**. The following list summarizes the changes that take place when you perform such an activation:

- New limits can be used.
- Binary job numbers are four bytes long.
- JOE and JQE chaining is by three byte index.
- Checkpoint data set format changes are made.
- JIX is a hash table (not a look-up table).
- Jobids take the form J0nnnnnn if the job number range is greater than 99,999.
- JES2 can track up to 8 million track groups per job.

Note: You should review the general description of activating your JES2 MAS to **z2 Mode**. See "z2 Mode for $ACTIVATE" on page 17 for the details.

z2 Mode Field Changes: This section summarizes field and field name changes to aid in migrating exit routines. Fields that have had definition changes have also had their names changed. Assembler errors will occur in any exit that references the "old" field names. When these errors occur, you must correct the use of the fields as well as the field names.

Table 11 lists mode-sensitive JQE pointers. In **R4 Mode**, the pointers are offsets; in **z2 Mode**, they are indexes. The names of the fields are changed to ensure the fields are properly used. The field names JOEJQE and JQENEXT refer to composite fields and are eliminated to ensure no one misuses composite fields.

Table 11. JQE Indexes

Pre z/OS V1R2 Field Name	z/OS V1R2 Field Name	z/OS V1R2 Field Length (Bytes)
$JQFREE in $HCT	$JQFREEI	4
$JQHEADS in $HCT	$JQHEADI	47*4
$JQRBLD in $HCT	$JQRBLDI	4
JOEJQEB in $JOE	JOEJQEI	3
JQENEXTB in $JQE	JQENEXTI	3
CATQHEAD in $CAT	CATQHDI	4
Note: All uses of the old field names must be changed to use the new field names. Also you need to consider whether the checkpoint is **R4 Mode** or **z2 Mode**; this determination tells you whether the field contains an offset or an index, respectively.		

A JQE offset is converted to an address by adding the value in $JOBQPTR to the offset. A JQE index is converted to an address by multiplying the index by the field, $JQELEN, and then adding $JOBQPTR to the resulting offset.

Note: You should consider using the $QJQE service to process JQE queues instead of processing them manually.

Table 12 on page 21 lists mode-sensitive JOE pointers. In **R4 Mode**, the pointers are offsets; in **z2 Mode**, they are indexes. The names of the fields are changed to ensure that the fields are properly used. The field names JQEJOE, JOENEXT,

JOEPREV, JOEJQNXT, JOECHAR, JOECHNXT, JOEWKPTR refer to composite fields and are eliminated to ensure no one misuses composite fields.

Table 12. JOE Indexes

Pre z/OS V1R2 Field Name	z/OS V1R2 Field Name	z/OS V1R2 Field Length (Bytes)
JOTFREQ in $JOT	JOTFREQI	4
JOTCHRQ in $JOT	JOTCHRQI	4
JOTPURGQ in $JOT	JOTPRGQI	4
JOTHOLDQ in $JOT	JOTHLDQI	4
JOTCLSQ in $JOT	JOTCLSQI	3*36*4
JOTNTWKQ in $JOT	JOTNTWQI	4
JOTRDYWQ in $JOT	JOTRDWQI	4 * JOTNUMWQ
JOTRBLDQ in $JOT	JOTRBLQI	4
JQEJOEB in $JQE	JQEJOEI	3
JOENEXTB in $JOE	JOENEXTI	3
JOEPREVB in $JOE	JOEPREVI *	3
JOEJQNXB in $JOE	JOENXJQI	3
JOECHARB in $JOE	JOECHARI	3
JOECHNXB in $JOE	JOECHNXI	3
JOEWKPTB in $JOE	JOEWKPTI	3
JOENETCH in $JOE	JOENETCI	4

Notes:

1. * This field, in **z2 Mode**, is an index unless the high order bit is "on"; then it is an offset.
2. All uses of the old field names must be changed to use the new field names. Also, you must consider whether the checkpoint is **R4 Mode** or **z2 Mode**; this determination tells you whether the field contains an offset or an index, respectively.

A JOE offset is converted to an address by adding the value in $JOTABLE to the offset. A JOE index is converted to an address by multiplying the index by the constant, JOESIZE, and then adding $JOTABLE to the resulting offset.

Note: You should consider using the $#JOE service to process JOE queues. $#JOE is particularly useful since it processes the JOEPREVI anomaly without you having to do so.

Table 13 shows where two byte job number fields have been changed to four bytes. These changes apply whether you operate in **R4 Mode** or **z2 Mode**.

Table 13. Four Byte Job Number Fields

Pre z/OS V1R2 Field Name	z/OS V1R2 Field Name	Control Block
CRXJOBNO	CRXJBNUM	$COMWORK
DCNVJBNO	DCNVJNUM	$DTECNV
FAXBCJP	FAXBJCJP	$FSAXB
GTWJQNUM	GTWJBNUM	$GTW
GTWJQEMX	GTWJQMAX	$GTW

Table 13. Four Byte Job Number Fields (continued)

Pre z/OS V1R2 Field Name	z/OS V1R2 Field Name	Control Block
GTWJQEFR	GTWJQFRE	$GTW
JIBJOBNO	JIBJBNUM	$JIB
JNEWJQE	JNEWJNUM	$JNEW
PSOJOBNO	PSOJBNUM	$PSO
ROTEJBNR	ROTEJNUM	$ROTT
SFRJBNO	SFRJBNUM	$SFRB
SSWJOBNO	SSWJBNUM	$SFSWORK
SJBJOBNO	SJBJBNUM	$SJB
TTEJOBNO	TTEJBNUM	$TTE
Note: You must ensure that any use of these fields refers to <u>four byte</u> job numbers whether you operate in **R4 Mode** or **z2 Mode**.		

Table 14 shows the EQUATES that have changed to four bytes.

Table 14. Four Byte EQUATES

Pre z/OS V1R2 Binary Job Number Equate Name	z/OS V1R2 Equate Name	z/OS V1R2 Value	Control Block
DSIDJBNO	DSIDJNUM	EQU 0,4	$FSIEQU
$MAXJBNO	$MAXJNUM	EQU 999999	$HASPEQU
$MAXJQES	$MAXNJE	EQU 200000	$HASPEQU
COFSEC	COFSEC	COFOPT2+1,4	$COMWORK
You must ensure that any use of these EQUATES refers to <u>four byte</u> values whether you operate in **R4 Mode** or **z2 Mode**.			

$SPID, the prefix area for spool resident control blocks, is changed to contain a four byte job number instead of a two byte job number. Table Table 15 shows the field names and control blocks associated with the $SPID.

Table 15. $SPID Four Byte Fields

Pre z/OS V1R2 Field Name	z/OS V1R2 Field Name	Control Block
HDBJOBNO	HDBJBNUM	$BUFFER
CHKJOBNO	CHKJBNUM	$CHK
IOTJOBNO	IOTJBNUM	$IOT
IOTJBNMB	deleted	$IOT
JCTJOBNO	JCTJBNUMI	$JCT
NHSJOBNO	NHSJBNUM	$NHSB
OCTJOBNO	OCTJBNUM	$OCT
SWBJOBNO	SWBJBNUM	$SWBIT
You must ensure that you use these field to refer to <u>four byte</u> job numbers whether you operate in **R4 Mode** or **z2 Mode**		

$SIG is changed to use the one byte reserved space before the job number field. The new 3 byte job number field has the name, SIGJOBNUM. SIGJOBNO is no

longer valid. You must ensure that you use this field to refer to the 3 byte job number whether you operate in **R4 Mode** or **z2 Mode**.

The formatted command DSECTs and network header/trailer DSECTs (Table 16) have been adjusted to handle job numbers up to 999,999 These DSECTs have both a two byte version and four byte version of the job number. You need not consider control block changes for these DSECTs in z/OS JES2 Version 1 Release 2 because JES2 uses the four byte job number field if available; otherwise JES2 uses the two byte job number field.. Both job number fields are compatible with older releases.

Table 16. Formatted Command DSECTs

Two Byte Field Name	Four Byte Field Name	Control Block
COMFJID	COMFJNO	$COMMWORK
COSIJID	COSIJNO	$COMMWORK
NJHGJID	NJHGJNO	$NHD
NJHOOJNO	NJHOOJBN	$NHD
NMRFJID	NMRFJNO	$NMR
CMBFJID	CMBFJNUM *	$CMB
* CMBFJNUM is new for z/OS JES2 Version 1 Release 2		

Table 17 shows the changes to job number fields or JQE counter fields in control blocks that JES2 writes to the checkpoint data set.. Existing two byte fields are renamed and a new four byte field created to accomodate the larger job number. In **R4 Mode**, JES2 uses the existing (but, renamed) two byte job number field to process the job number (JES2 does <u>NOT</u> examine the four byte field). After you have "activated" to **z2 Mode**, JES2 uses the new four byte job number field. The two byte job number field is discarded.

Table 17. Checkpoint Control Block Fields

Pre z/OS V1R2 Field Name	z/OS V1R2 Field Name (R4 Mode)	Four Byte Field Name (z2 Mode)	Control Block
DASJOBNO	DASJOBNO_R4	DASJBNUM	$DAS
$MAXJOBS	$MAXJOBS_R4	$JQENUM	$HCT
$JQEFREC	$JQEFREC_R4	$JQEFRCN	$HCT
JCTINJO	field deleted	field deleted	$JCT
JQEJOBNO	JQEJOBNO_R4	JQEJBNUM	$JQE
JQEINJNO	JQEINJNO_R4	JQXIJNUM	$JQE
JQEARMID	JQEARMID_R4	JQEARMMI	$JQE
JQEWSLOK	JQEWSLOK_R4	JQEWSLCK	$JQE

Table 17. Checkpoint Control Block Fields (continued)

Pre z/OS V1R2 Field Name	z/OS V1R2 Field Name (R4 Mode)	Four Byte Field Name (z2 Mode)	Control Block
$HCT	Reserved space in the HCT is now used for the two new four byte fields that identify maximum number of jobs ($JQENUM) and JQE free count ($JQEFRCN).		
$JCT	The 2 byte original job number JCTINJNO is deleted from the JCT and <u>NOT</u> replaced with a four byte job number. A down-level JES2 (Pre z2) cannot rely on this field.		
$JQE	The new four byte initial job number field (JQXIJNUM) uses previous reserved space in the JQX. JES2 processes four byte job numbers (JQEJBNUM) in the JQE in the fields once occupied by JQEARMID, JQEWSLOK and 2 reserved bytes. When JES2 runs in **z2 Mode**, the contents of the JQEARMID and JQEWSLOK fields are stored in the 2 bytes used by JQENEWSU. JES2 accessess this data using the names JQEARMMI and JQEWSLCK respectively. Figure 1 shows the changes.		
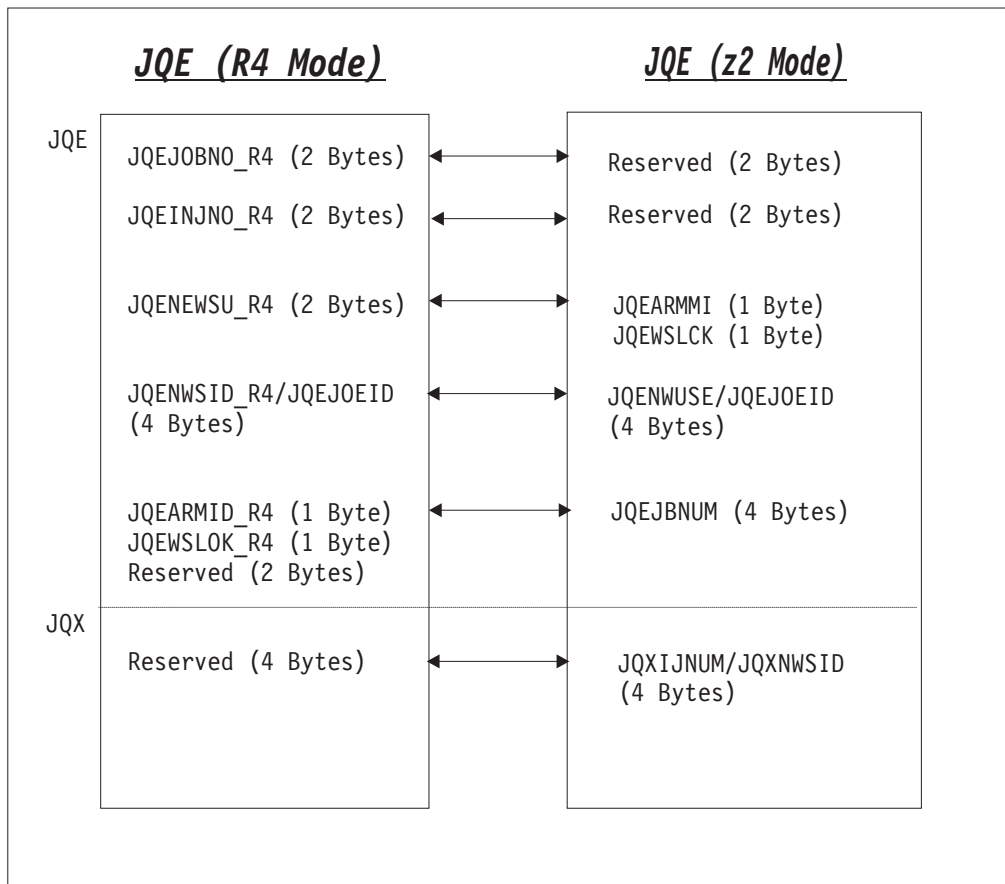
Figure 1. JQE (R4 Mode - vs - z2 Mode)

When examining fields in the real JQE/JQX, the exit must be sensitive to the checkpoint mode. If in **R4 Mode**, you must use the fields with the "_R4" suffix. In **z2 Mode**, you use the new z/OS V1R2 four byte fields. When using a JQA, always use the new z/OS V1R2 fields regardless of whether you are in **R4 Mode** or **z2 Mode**.

Note: When the JQA is returned (or checkpointed), $DOGJQE sets the appropriate fields in the real JQE depending on whether running in **R4 Mode** or **z2**

Mode. $DOGJQE also sets the four byte fields in the JQA and alleviates the need for code that uses the JQA to determine whether JES2 is processing the checkpoint data set in **R4 Mode** or **z2 Mode**.

Macro Updates Relating to JOE and JQE Changes

There are some new and changed macros used to minimize the impact to JES2 exit routines. . By using these macros, your exit routines do not have to determine whether JES2 is processing in **R4 Mode** or **z2 Mode**.

$QJQE and $#JOE

These macros process the JOE and JQE control block chains regardless of checkpoint mode.. $#JOE is changed to process chains (such as the CHAR JOE chain). $QJQE and $#JOE have improvements to their loop control.

$DOGJQE

The JQA returned by $DOGJQE always reflects the **z2 Mode** processing format. Code that processes an artificial JQE (JQA) and examines JQE/JQA fields need not care whether JES2 is running in **z2 Mode** or **R4 Mode**.

$JQEJNUM

$JQEJNUM is a new macro used to obtain the binary job number for a particular JQE. This macro operates in **z2 Mode** and **R4 Mode** and processes the JQA or JQE that is passed.

$JBIDBLD

This macro converts a binary job number to a printable jobid. $JBIDBLD is updated to accept a JQE (or JQA) as input and operates in **z2 Mode** or **R4 Mode**. $JBIDBLD formats the jobid, in the correct way, based on the current job number range.

JESNEWS

In **R4 Mode**, JESNEWS processing is exactly like JES2 z/OS V1R1 except for field name changes. In **z2 Mode** , JESNEWS processing uses a JESNEWS level number to correlate the JOE and the JNEW control blocks. This JNEW control block contains the job number of the JESNEWS JQE.

Checkpoint Improvements

Prior to JES2 z/OS V1R2, switching between checkpoint modes required an all member warm start. Now installations can switch checkpoint modes using the `$T CKPTDEF,MODE=DUAL|DUPLEX` command. An all member warm start is not required. Changing to `DUPLEX` mode is always allowed. Changing to `DUAL` mode is NOT allowed when either CKPTn or NEWCKPTn is on the coupling facility or if DUPLEX=OFF is specified on the CKPTDEF initialization statement and both CKPTs are in use.

Note: You can use the `$T CKPTDEF,MODE=DUAL|DUPLEX` command even though some MAS members are not at the JES2 z/OS V1R2 release.

See *z/OS JES2 Commands* and *z/OS JES2 Initialization and Tuning Reference* for details.

Large Volume Support

Prior to JES2 z/OS V1R2, the SPOOL data set had to be contained within the first 64K tracks of the volume. With the new JES2 z/OS V1R2 addressing scheme, the SPOOL data set can reside anywhere on the volume.

Since SPOOL volumes are read by a number of vendor products (including SDSF), any change to how SPOOL addressing operates affects these products. Specifically, JES2 now supports both a relative and absolute track addressing for SPOOL. The SPOOLDEF initialization statement is updated to allow you to specify whether you "always", "as needed", or "never" want to use relative MTTRs in processing the JES2 SPOOL. You use the new keyword, RELADDR= with a value of ALWAYS , ASNEEDED, or NEVER to specify your choice.

Notes:

1. Relative addressing in processing the JES2 SPOOL is only used if all members of the MAS support relative addressing. Any member that is not supporting relative addressing fails to initialize and a HASP401 message results.

2. The size of a SPOOL data set is still limited to 64K tracks.

The $T SPOOLDEF command is updated to allow you to update the current setting for RELADDR=. If relative addressing is being used for a volume, the $D SPOOL,UNITDATA displays a new keyword, BASETRAK=. The value of BASETRAK= must be added to the TT (part of MTTR) to convert a relative track address to an absolute track address.

A new interface is provided to read records from JES2 SPOOL. This interface is implemented as an option on the SSI 71 call. When you use this interface, you need not know about internal JES2 MTTR processing.

See *z/OS JES2 Initialization and Tuning Reference* for using the new RELADDR= operand on the SPOOLDEF initialization statement and *z/OS JES2 Commands* for using the $T SPOOLDEF command.

Dynamic JES2 PROCLIB and INCLUDE Initialization Statement

Updating the JES2 PROC and concatenating PROCLIB data sets is inconvenient and disruptive to installations. Problems with the JES2 PROC often prevents JES2 from starting. For jobs that require PROCLIB data sets, the data sets have to be defined to the JES2 PROC, which would invariably necessitate a restart to update. JES2 z/OS V1R2 provides a PROCLIB initialization statement to define a logical DD name that the JES2 converter can use. Up to 255 data set names can be associated with the PROCLIB initialization statement. Static PROCLIB data set concatenations, defined in the JES2 PROC can be eliminated.

JES2 z/OS V1R2 provides the INCLUDE initialization statement to help you to reduce the need to update the JES2 PROC. Instead of declaring a concatenation of initialization data set members or data sets in the JES2 PROC, the installation can use the INCLUDE initialization statement that "include" other initialization streams as part of the JES2 initialization process. The INCLUDE initialization statement requires a data set name (and optional member name) where the additional initialization stream can be found. Also, you can include a D INCLUDE statement in your initialization stream to display the current data set that is being processed.

The following commands provide additional flexibility:

- $ADD PROCLIB creates a new dynamic PROCLIB concatenation.
- $DEL PROCLIB deletes an existing dynamic PROCLIB. This command does not affect static PROCLIBs.
- $T PROCLIB updates an existing dynamic PROCLIB
- $D PROCLIB displays the current dynamic PROCLIBs only and does not display the static ones. This command has a DEBUG option that provides additional

information about the dynamic PROCLIB concatenation as well as logically deleted dynamic PROCLIBs that are still in use.

See *z/OS JES2 Initialization and Tuning Reference* and *z/OS JES2 Commands* for more details.

To complement dynamic PROCLIBs, the JES2 PROC should be updated to:

- Minimize the information in the JES2 PROC (thus limiting the chance that some change in the system makes the PROC unusable).
- Provided a generic PROC to start JES2 for those situation where the JES2 PROC becomes unusable. The two main statements in the JES2 PROC identify the PROCLIB data sets and the JES2 initialization data sets. JES2 provides initialization statements for these two data sets.

Spinning JESlog Data Sets for Long Running Jobs

When you have jobs that run for extended periods of time (maybe days, even weeks...), JESlog data needs to be periodically spun as the job runs. JES2 provides for the "spin off" or "suppression" of JESlog data (specifically the data sets, JESMSGLG and JESYSMSG). This periodic ability to spin off JESlog data improves the use of JES resources.

Spinning or suppressing JESlog data is accomplished with the following:

- A JES2 JOBCLASS initialization statement keyword, JESLOG=, where you specify whether to "spin off" or suppress the output of JESlog data sets. The JESLOG= keyword on the JOBCLASS(v) initialization statement along with the JESLOG= keyword on the JOB card makes the specification complete. The specification on the JOB card overrides what is specified on the JOBCLASS(v) initialization statement.
- The SPOOL Data Set Browse interface is updated to allow the fetching of the "logical" JESMSGLG and the logical JESYSMSG data sets.
- The $T job,SPIN command causes immediate spinning of JESlog data sets provided either JESLOG=SPIN or JESLOG=(SPIN,xxx) was specified.
- The $T JOBCLASS(x),JESLOG=(SPIN,yyy) command lets you alter your setting on the JOBCLASS(v) initialization statement. The jobclass you want to alter the SPIN value for is *x* and *yyy* is the SPIN option you specify. This command can be issued at any time and only affects those jobs converted after the change is made.
- The $T REQJOBID,JESLOG=(SPIN,yyy) command lets you set the SPIN value for "Request Jobid" jobs much the same way $T JOBCLASS(x),JESLOG=(SPIN,yyy) does for ordinary jobs.
- The "Request Jobid" SSOB extension, IEFSSRR, allows attributes (including SPIN values) to be set for "Request Jobid" jobs in the same way that JCL statements do for ordinary jobs.

See *z/OS JES2 Initialization and Tuning Reference* and *z/OS JES2 Commands* for more details.

The SPOOL Data Set Browse interface is updated to allow the fetching of the "logical" JESMSGLSG and the "logical" JESYSMSG data sets. Spool Data Set Browse is document in *z/OS JES2 Initialization and Tuning Guide*.

There are new messages that support spinning JESlog data sets for long running jobs:

- HASP138

- HASP860
- HASP130

See *z/OS JES2 Messages* for details.

Attention:

- Jobs that are converted on a JES2 z/OS V1R2 MAS member but run on a pre-JES2 z/OS V1R2 MAS member cannot perform the JESLOG spin off.
- Jobs that are converted on a JES2 z/OS V1R1 (or earlier) MAS member cannot perform JESLOG spin off regardless of which MAS member they run on.
- The converter (and START commands) cannot recognize the JESLOG specification in a JES2 z/OS V1R1 (or earlier) MAS member. This condition produces an error.

Message HASP443 - Updated

The HASP443 message clarifies the reason for a SPOOL allocation failure. The text of the message is:

```
►►──HASP443──vvvvvv──NOT ALLOCATED───────────────────────────────────►

                                                        (1)
►──┬─OBTAIN FAILURE, RC=01──────────────────┬───────────────────────►◄
   ├─I/O ERROR DURING FORMAT, RC=02──────────┤
   ├─I/O ERROR DURING READ VALIDATION, RC=03─┤
   ├─DYNAMIC ALLOCATION FAILURE, RC=04───────┤
   ├─INTERNAL ERROR, RC=05───────────────────┤
   ├─UNABLE TO GET DEVICE INFORMATION, RC=06─┤
   ├─UNUSABLE EXTENT SIZE, RC=07─────────────┤
   └─EXTENT ABOVE 64K TRACK LIMIT, RC=08─────┘
```
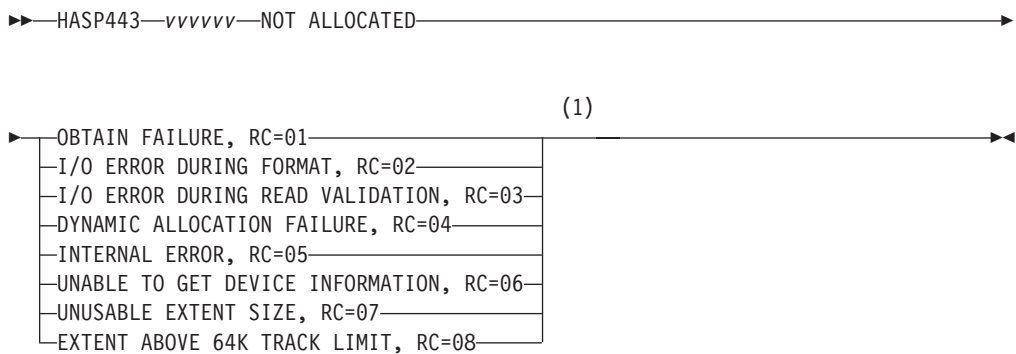
Notes:

1   One or more of these reason code can be displayed with HASP443.

Command Syntax Migration Aid Removal (Exit 5)

In OS/390 JES2 Version 2 Release 4, JES2 provided a command syntax compatibility and migration aid in the form of an automatically installed exit 5 routine. The migration aid was intended as a temporary bridge to give customers time to adapt to the command syntax changes made in OS/390 JES2 Version 2 Release 4. As of z/OS JES2 Version 1 Release 2, the migration aid is no longer automatically installed and enabled. If you are not using the command syntax documented in *z/OS JES2 Commands* and are still relying on the migration aid, you may experience erroneous system operation or receive HASP649, HASP650, HASP003 messages.

HASX05C is still provided as a sample exit in SYS1.SHASSAMP. For more details concerning the operational implications of removing the command syntax migration aid, see the applicable migration chapter of this publication and within it, the section describing "Operations".

Exit 14 and Exit 49

An indicator has been provided that controls class list optimization.

Other New and Changed Macros

- $XMPOST

  The $XMPOST macro is changed. When you specify $XMPOST XMPLIST=YES (or leave XMPLIST off and let it default), the mapping of the input parameter list is changed. The expected parameter list mapping is as follows:
  - +0 Error return address
  - +4 Address of ECB to be POSTed
  - +8 Address of related ASCB
  - +C ECB to be POSTed (if LOSTPOST=CHECK is specified on the macro call)

  For further information see *z/OS JES2 Macros*.

Warm Start Redesign

JES2 z/OS V1R2 improves warm start processing that in the past would cause system malfunctions because of a shortage of BERTs (malfunctions such as, hangs, repeated ABENDs, and other erroneous situations) JES2 z/OS V1R2 warm start code is rewritten to alleviate this shortage problem. JES2 z/OS V1R2 provides this warmstart redesign whether in **R4 Mode** or **z2 Mode**.

APPLCOPY Application

APPLCOPY is no longer supported.

Counting of Null Records

Prior to JES2 z/OS V1R2, null records written to SYSOUT data sets were padded to a single blank record. Because of this padding, these records were counted as records in the job totals. This count is not consistent with SYSOUT received over NJE. Null records from NJE are not padded and not counted in the job totals.

Processing to pad null SYSOUT records was deleted in JES2 z/OS V1R2. This deleted processing makes all the record counting within JES2 consistent. However, this consistent counting may change record counts for existing jobs. This change is most noticeable in a mixed MAS (JES2 z/OS V1R2 running with earlier levels of JES2).

z/OS JES2 Version 1 Release 1 Overview

$ACTIVATE

You must be in "Release 4 mode" to run JES2 z/OS V1R1 since support for "Pre-release 4 mode" has been dropped. Support for the $ACTIVATE command is no longer needed since only one checkpoint mode is currently supported.

Spool Performance Improvements

JES2 SPOOL management was enhanced in this release to improve the performance of placing data on SPOOL. In prior releases, SPOOL management was performed using fencing to control how many volumes a job can use and the SPOOLDEF TGBPERVL= parameter to control the amount of SPOOL space available for use by jobs.

The default fencing option provided by JES2 could impact performance. Implementing a set of exits to allow installations to define a fencing policy was not

a viable option for some installations. These problems were addressed by providing a new fencing option and a new function, SPOOL affinity.

This release also deleted the TGBPERVL= operand on SPOOLDEF. Instead of trying to describe how to set an external to manage availability of SPOOL space, JES2 will now manage this to provide maximum availability of SPOOL space for jobs writing to SPOOL.

Changes to JES2 external and internal processing that enable spool performance improvements as follows:

- SPOOLDEF FENCE= changes
- Setting the BLOB size
- Dropping use of SPOOLDEF TGBPERVL=
- The new $T SPOOL(nnnn) command

SPOOLDEF FENCE= Changes

The FENCE= keyword parameter on the SPOOLDEF initialization statement is changed as follows:

```
►►──SPOOLDEF──FENCE=──(──ACTIVE=──┬─YES─┬──,VOLUMES=nnn──)──────────────────►◄
                                  └─NO──┘
```

ACTIVE=
    specifies if any fencing is active.

VOLUMES=
    specifies the number of volumes a job is fenced to.

When you specify
```
SPOOLDEF....FENCE=(ACTIVE=YES,VOLUMES=1).....
```

you get the same function as in the previous JES2 release when you specified FENCE=YES. FENCE=YES will be permitted for compatibility and converted to ACTIVE=YES and VOLUMES=1. This specification of SPOOLDEF FENCE= is honored on $T SPOOLDEF and on cold start.

SPOOLDEF TGBPERVL= Dropped

Using the keyword parameter, TGBPERVL= on the SPOOLDEF initialization statement has been dropped.

Spool Affinity

New with this release, JES2 implemented a form of spool fencing that associates MAS members with particular spool volumes. Each spool volume has masks of systems that can allocate space on that volume. This function is called SPOOL affinity.

The $T SPOOL(nnnn) command was added to implement spool affinity. This command allows the installation to assign its systems to spool volumes. This command has one valid keyword operand, SYSAFF=. SYSAFF function similar to SYSAFF on the $TINTRDR command. The value you specify for SYSAFF= on the $TSPOOL(nnnn) command is also added to the output produced by the $D SPOOL command.

Note: There is no equivalent initialization statement that corresponds to the $T SPOOL(nnnn) command.

Also, the $S SPOOL command supports the SYSAFF= parameter. The $S SPOOL command is old but specifying the SYSAFF= parameter on this command is new.

BLOB Size

JES2 implements a form of spool fencing that associates MAS members with particular spool volumes. Each spool volume has masks of systems that can allocate space on that volume. This mask limits the entries that can be placed in the BLOB for that system. Since all track group allocations are made from the BLOB, jobs are limited to the spool volumes associated with a system. If all spool volumes that are associated with a system become full, spool space starts to be allocated from any spool volume that has space. (The assignment of systems to a spool volume is accomplished through the $T SPOOL command.)

In previous releases, the size of the BLOB was determined by the TGBPERVL= keyword multiplied by the number of volumes that had space (with a max of 253 entries per system). The introduction of SPOOL affinity made this method of calculating BLOB size more complicated. To simplify setting up JES2, the size of the BLOB will be calculated internally by JES2. The algorithm will fill all 253 entries in the BLOB as long as the total amount of SPOOL space in all the BLOBs is less than 80% of all available SPOOL space. When the size of the BLOB starts to exceed 80% of the available SPOOL space, then the system will reduce the size of the BLOB so that only 80% of the available space is in the BLOB.

Also, as part of this change, the SPOOL status messages (from a $DSPOOL command) will now consider any SPOOL space in a BLOB as not in use. In previous releases, all space that was in a BLOB was considered in use (even though no jobs are using that space). This will be noticeable in small systems with a small amount of space allocated to SPOOL. These systems will see a lower SPOOL utilization with this release of JES2 as compared to earlier releases.

Exit 11 and Exit 12

Even with the new SPOOL management functions, the use of the ″SPOOLs allowed mask″ for a job continues as in previous releases. However, these exits may need to be updated if your installation decides to implement any of the new functions in this release. In particular a new set of SPOOL masks have been created to reflect the new SPOOL affinity functions. JES2 exits 11 and 12 allow an installation to implement their own type of spool fencing. Each bit in the mask continue to represent one Spool volume. A bit being on means that space can be allocated from the corresponding volume. Exits 11 and 12 can still manipulate the spools allowed mask and assign spool space based on their particular criteria.

New and Changed Macros

$#TJEV macro can be used in exits to find out why output is not printing.

Performance Enhancements

Additionally, JES2 is upgrade with the following performance enhancements:
- TPDEF SNABUF=SIZE= changes
- TRACEDEF PAGES= changes
- Improved CKPT scan

TPDEF SNABUF Changes

To improve SNA performance and better align with other networking products, JES2 has increased the SNA buffer size from a maximum of 3840 to 32512. This is a change to the TPDEF SNABUF=SIZE= keywords.

TRACEDEF PAGES= Changes

The default for TRACEDEF PAGE= is changed from 2 to 9. For compatibility with other releases of JES2, specification of 1-8 are converted automatically to 9.

Improved $#GET/$#POST Processing

The $#GET service is used by devices (and SAPI) to select SYSOUT for processing. The service scans the output queues looking for the best output element that matches the criteria specified by the device. This scan can consume a large amount of resources dependent on the amount of SYSOUT queued to a particular class. The $#POST service is used by processes that add work to the the output queues; this wakes up devices that are waiting for work. Once a device is posted that new work exists, the device calls $#GET to find the work in the output queue.

In previous releases, there was no communication between the $#POST and the $#GET (except for the actual posting). New in this release, $#POST will pass to $#GET the address of the new output element. $#GET can keep track of 1 element passed by $#POST. It can also keep track of the fact that there is no output in the queue for a particular device.

With this support, when output is created for a device that is idle (started with nothing to print), it can select that output without ever scanning the output queue. When finished, it can go idle knowing that there is no new work to do. This greatly reduces the overhead associated with sending work to a devices that generally goes idle between jobs.

Other Items

ZAPJOB Initialization Statement

The ZAPJOB initialization statement is used to remove all traces of a job structure from the JES2 job queue. This statement is intended for situations where a job cannot be removed using normal JES2 commands or by a JES2 restart.

$ZAPJOB Command

$ZAPJOB removes all traces of a job structure from the JES2 job queue. You must ensure that the job and any output it may have created is not active in any JES2 process or active on a device. Zapping a job that is active can lead to ABENDs and the loss of a PCE until JES2 is restarted.

$HASP250 has been updated to indicate that a job has been zapped.

$HASP834 Deleted

$HASP834 message has been deleted.

Active DCTs

Active DCTs are now chained out of the HCT ($NJEADCT, $OFFADCT, $LCLADCT).

Spool Data Set Browse Documented

The application interface to use the SPOOL Data Set Browse function has been simplified to not require access to internal JES2 data areas. This will enable more applications to take advantage of this interface. Complete documentation for this interface has been included in an appendix of the *z/OS JES2 Initialization and Tuning Guide*.

Multi-System Dumps

To aid in first failure data capture, OS/390 Version 2 Release 10 JES2 generates JES2 dumps on all members of a MAS when certain JES2 work queue errors occur.

This is referred to as a "multi-system dump." Since multiple dump data sets are created for these errors, procedures need to be put in place to ensure that all dumps for an error (even though they appear to be duplicates) are retained for problem analysis.

OS/390 JES2 Version 2 Release 8 Overview

CF System Managed Rebuild Support

JES2 checkpoint management has been improved with the CF System Managed Rebuild Support. JES2 now supports a SETXCF START REBUILD command for checkpoints that are on a coupling facility. With Release 8, all XES requests are now subtasked to prevent line timeouts and function interruptions.

Message HASP069 - Updated

Two new reasons for issuing this message are the SLIP and Installation requests.

```
HASPO069   JES2 SYSTEM DUMP SUPPRESSED BY
[DUMP ANALYSIS ELIMINATION]
[SLIP]
[INSTALLATION REQUEST]
```

Message HASP263 - Updated

A new form of Message HASP263 has been provided to indicate that a CF auto rebuild is in progress. The new form is:

```
HASP263   WAITING FOR ACCESS TO JES2 CHECKPOINT.
STRUCTURE REBUILD IN PROGRESS
```

The previous forms of this message still exist and have not changed. The reason for issuing this message has not changed (cannot obtain the checkpoint lock and LOCKOUT has expired). The new text indicates the reason the lock cannot be obtained in a system managed CF auto rebuild.

Message HASP292 - Updated

The time interval for this message has changed from 30 seconds to the current value of MASDEF LOCKOUT. The new optional text explains that the delay was caused by a system managed CF auto rebuild.

```
HASP292   MEMBER memb-name  --   JES2 WAITING FOR RESPONSE
FROM COUPLING FACILITY DURING CHECKPOINT
READ/WRT/FORMAT/IO
ON STRUCTURE strname
(STRUCTURE REBUILD IN PROGRESS)
```

Message HASP538 - Updated

The IXCQUERY return code and reason code are new. See *MVS Sysplex Services Reference* for an explanation of these codes.

```
HASP538   THE STRUCTURE SPECIFIED BY STRNAMEstrname FOR
file DOES NOT EXIST. PLEASE VERIFY THAT THE VALUE
IS CORRECT.
IXCQUERY RC=xxxxxxxx   RSN=xxxxxxxx
```

JES2 Dispatcher Rolling Trace - New

A new CTRACE is provided that is similar to the JOE and JQE CTRACSs. This CTRACE has a subtype of DISP and 2 types of entries (dispatch and $WAIT). The dispatch entry contains the PCE address and type, the $POST reason, the address to be given control (R15), ASC mode, PSW key, and the amount of time the PCE $WAITed. The $WAIT entry includes the PCE address and type, $WAIT parm list, ASC mode, PSW key, and run time statistics (CPU time, wall clock time, I/O count, $CKPT count, $QSUSE indicator).

OS/390 JES2 Version 2 Release 7 Overview

This section describes additions, changes, and deletions to JES2 introduced by OS/390 JES2 Release 7.

Changes that Affect JES2 Installation Modifications

The following summarizes additions, changes, and deletions that affect JES2 installation modifications in OS/390 Release 7 JES2.

Changes to Exit 8

Exit 8 can now receive control for control block I/O ($CBIO) calls from the FSS address space.A new bit was added to the XPL to identify that this $CBIO was done for an FSS printer.

Changes to Exit 25

The $BUFIO and $BUFCK services in HASPFSSM have been deleted. The functions are now provided by the $CBIO service. TYPE=WAIT was added to $CBIO to support the functions that were previously provided by $BUFCK.

Changes to Exit 46 and Exit 47 for NJE Network Processing

- RCCS Headers

  Exit 46 and Exit 47 will be invoked for an additional type of header (RCCS header) in OS/390 Release 7. If you have exit 46 and/or exit 47 installed on your system, you should examine your exit(s) to determine if a change is needed. If you modeled your exit 46 after the sample exit 46 shipped with the product (HASX46A), you should take note that the logic in the sample exit has changed to accomodate the additional type of header that can now be processed by this exit.

  Since exit 46 is now invoked to process RCCS headers, you can now remove, modify, or add an RCCS header to the jobstream before it is transmitted into the NJE network. Exit 47 now allows you to process the RCCS header when it is received from the network.

  Compatibility: This support only introduces an incompatible change for those installations that have exit 46 or exit 47 logic that does not accomodate the additional type of header.

- Exit 47 Input Parameter List Changes

  A JQE address is passed to Exit 47 in field X047JQE. However, this address may be the address of a working copy of the JQE. Examine the X047IND field for X047BJQE; if the X047BJQE flag is on, then X047JQE contains the address of a working copy of the JQE. In this case, the address contained in X047JQE **cannot** be used as input to any services that expect the addresss of a real JQE (for example $DOGJQE).

Fiber Channel Support
JES2 I/O processing for spool and checkpoint has been updated to take advantage of the changes made in FICON (Fiber channels).

Checkpoint Performance
JES2 checkpoint performance has been improved by reducing the number of I/O to the checkpoint data sets under certain conditions. These changes include:

- Elimination of a Read to CKPT2 during checkpoint locking operations (READ1) if at least one checkpoint is on a coupling facility.
- Reduction of the number of writes to CKPT2 when checkpoint is in duplex mode.

As a result of these changes, the specifications for MASDEF HOLD= and DORMANCY= may need adjusting. In particular, if you have HOLD=0 specified in duplex mode, then prior to this change, the actual hold time was limited by the time it took to write to CKPT2. Now, a significant number of those writes have been eliminated and your actual hold time may become too short and cause thrashing.

If you are in duplex mode and have some members which specify DUPLEX=OFF, you may want to reconsider this specification. With Release 7, the CKPT2 data set is only written once every 10 writes to the CKPT1 data set. One of the reasons to specify DUPLEX=OFF was to reduce the I/O's to CKPT2. The combination of the two may cause an unacceptable amount of time to elapse before CKPT2 is written. If the only reason DUPLEX=OFF was specified was to reduce the number of I/O's to CKPT2, then with Release 7, all members should specify DUPLEX=ON.

Changes that Affect JES2 Operations

The processing for ended PCEs has been updated. When a non-required PCE abends a dump is taken and JES2 ends the PCE. The operator is no longer queried by the HASP070 message as to whether recovery is to be attempted. In addition, the HASP068 message has been updated to inform the operator of how many PCEs of the specified type remain.

Changes that Affect Problem Determination and Diagnosis

The following summarizes additions, changes, and deletions that affect problem determination and diagnosis in OS/390 Release 7 JES2.

New Record for Trace ID 17
Trace ID 17 has been updated to contain new measurements (total PCE wait time, number of $CKPTs issued, MVS wait time, $QSUSE time) that JES2 gathers during different phases of checkpoint processing. A new record 3 has been added to the READ2, PRIMARY WRITE, IMMEDIATE WRITE and FINAL WRITE checkpoint cycles that contains these measurements. If you use a program that analyses Trace ID 17 records, it may need to be updated to incorporate this support.

$DPCE Command - Additional Diagnostic Information
The $DPCE command is enhanced to provide the following additional diagnostic information:

- Ended PCE counts - $DPCE,ENDED>0 displays the PCE types that have ended.
- Active PCE counts - $PDCE,ACTIVE>0 displays the PCE types that have active work.
- Detailed PCE information - $PCDE (CNVT) displays current detailed information for each converter PCE. This information includes: where the PCE is

waiting and what it is waiting for, active and outstanding I/O counts, the exit the PCE is in, the current job the PCE is processing, the time the PCE waited, and whether the PCE has ended.

Message HASP291 Enhanced

Message HASP291 is enhanced in this release to include information from the MVS IEDB. This information includes the IOS completion code and the full device sense data. Two completion codes are presented because IOS combines a multiple of its completion codes into one for use by EXCP callers. In some cases, the IOS completion code will more accurately describe the error.

Message HASP064 - New Message

The old I/O error message, HASP094, was incomplete and difficult to read. A new message, HASP064, is added to this release to provide additional data in a format that is easier to understand. The HASP064 message displays the same information as the HASP094 message, but in a format similar to the HASP291 message. In addition, data from the MVS IEDB is added to the HASP064 message and includes the IOS completion code and the device sense data. Two completion codes are presented because IOS combines a multiple of its completion codes into one for use by EXCP callers. In some cases, the IOS completion code will more accurately describe the error. Also included is the failing MTTR/BBCCHHR.

$CBIO Services - Additional Error Recording

Additional error recording is added for the $CBIO services in the USER environment. CBIO errors will now generate a SYMREC similar to the one generated in the main task for control block I/O errors. In addition, the JES2 trace ID 3, previously from the FSS environment $BUFIO service, is moved to the USER environment $CBIO service. These changes provide additional data to help diagnose errors with SPOOL control blocks.

OS/390 JES2 Version 2 Release 5 Overview

JES2-Provided Client Print Services

JES2 element of OS/390 Version 2 Release 5 provides client print services by allowing applications such as the IBM OS/390 Print Server to allocate space on JES2 spool for print jobs submitted from workstations in the TCP/IP network.

Print jobs that use print services can take advantage of traditional JES2 print services. These include work selection and work balancing across logical printer groups, use of JES2 default SYSOUT attributes, SYSOUT processing progress and monitoring, and SYSOUT cancelling, holding, and releasing.

JES2 provides these client print services in a manner compatible with its standard operational characteristcs. For example, client print services can:

- Query their jobs to determine status, as does the JES2 operator
- Set jobs priorities consistent with JES2 priority values, and validate that a job can print on JES2 queues
- Use the SYSOUT application programming interface (SAPI) added to the JES2 element of OS/390 Version 1 Release 3.

Chapter 4. Migration Actions: JES2 z/OS V1R2 to JES2 z/OS V1R4

Customization

New and Changed Installation Exits

Exit 4

In z4, JES2 now supports the // XMIT JCL statement as a means of transmitting jobs to another node. This is in addition to the older /*XMIT , /*ROUTE XEQ and /*XEQ JECL statements

Migration Actions: Examine any exit 4 code that might be scanning for the older JECL statements that affect job execution routing to see if they should be updated to also scan for the JCL statement.

Exit 9

In z4, JES2 performs excession processing for every record. In prior releases this was done only when a spool buffer filled. As a consequence, in z4, exit 9 will be given control immediately when the limit is exceeded, not just when the spool buffer filled.

Migration Actions: Exits have been known to give a low increment (for example a value of 1) knowing that the exit will not be driven until the next spool buffer is filled. That is no longer the case.

Exit 22

In z4, the processing for STATUS/CANCEL has been changed to use data residing in a data space. This resulted in:

- A new parameter list for the exit.
- The passing of a address (and ALET) of a block ($STAC) residing in a data space.

Migration Actions: Exit 22 must be modified to use the new parameter list and to use the mechanism for accessing data space areas. See *z/OS JES2 Installation Exits* for further information.

Exit 44

In z4, JES2 no longer polices duplicate logons due to the changes in End of Memory processing. This was done in order to make TSO userids capable of logging on even though JES2 has not yet completed processing the (JES2) resources owned by the address space.

Migration Actions: Use sample exit HASX44SB in module HASX44A to reinstate the functionality of prior releases if that is needed.

Control Block Changes

$PIT

In z4, installations have the option of specifying zero JES2 initiators (i.e. use just WLM managed initiators). This is done by specifying INITDEF PARTNUM=0. The result is that $PITABLE and CCTPIT can have a value of zero since no PITs exist.

Migration Actions: Examine your exits for references to $PITABLE and CCTPIT and ensure that a value of zero is being handled.

$PSO

In z4, the $PSO has been moved to a data space

Migration Actions: Examine your exits for references to the PSO data area and update appropriately.

Executable Macro Updates

$#BUSY

In z4, the $#BUSY service was enhanced to perform $#POST when BUSY=OFF is specified and to perform $CKPT as appropriate.

Migration Actions: Examine all $#BUSY calls and remove unnecessary uses of $#POST or $CKPT. Failure to remove unnecessary $#POSTs will result in performance degradation.

Initialization

INCLUDE Statement

In z4, the INCLUDE statement has been enhanced to allow initialization statements to be read from the MVS logical PARMLIB concatenation. A second enhancement is that the DSN need not be supplied on an INCLUDE statement if the member being requested is in the same data set as the INCLUDEing member.

Migration Actions

Examine the initialization streams and update appropriately.

PARM operand on START command

In z4, JES2 start processing was enhanced to support a MEMBER= value specified via PARM= on the START command or within the PROC. This operand specifies a member of the MVS logical PARMLIB concatenation that contains the JES2 initialization stream.

If no source of initialization statements is specified either via MEMBER= nor via a successful OPEN of the HASPPARM DD data set, then the default is the HASxxxx (xxxx = the subsystem name, e.g. JES2) member in the MVS logical PARMLIB concatenation.

Migration Actions

If you have a member in the MVS logical PARMLIB with the default name mentioned above, take the appropriate action.

PARTNUM=0 on INITDEF statement is Now Supported

In z4, PARTNUM=0 is supported. This means that no JES2 initiators will be used. This in turn means that there is no PITs and $PITABLE and CCTPIT will be zero.

Migration Actions

If you decide to specify INITNUM=0, verify that no PITs is an acceptable condition.

New Monitor Address Space

In z4, JES2 starts a new (non-optional) address space for each JES2 instance called xxxxMON (xxxx = subsystem name e.g. JES2) to monitor the corresponding J3S2 address space.

Migration Actions

Be aware of the existence of a new system address space.

Security Considerations

New Profiles for monitor commands

In z4, JES2 monitor was provided. The monitor has a set of commands used to give information about the status of the JES2 address space. The entity name of these monitor commands in the CMDAUTH class is of the form: xxxxMON.action.object where xxxx is the subsystem name (e.g. JES2). The complete list of entity names for the JES2 monitor commands can be found in *z/OS JES2 Initialization and Tuning Guide*.

Migration Actions

New profiles may be required for RACF in order to allow monitor commands to be issued.

Operations

TSO GR, TSO SEND Commands and TSO Logons

In z4, JES2 discontinued policing TSO logons. Duplicate instances of a given userid can be logged on at the same time. It is highly recommended that the GRS RNL list be updated to convert SYSIKJUA to a SYSTEMS ENQ in order to prevent duplicates. If the RNL is not updated, and duplicate instances of a userid are logged on, then TSO GR and SEND messages to the duplicately logged on userids will not function.

Migration Actions

Ensure that the procedure outlined in the *z/OS TSO/E Customization* is followed in order to block duplicate logons.

PARM operand on START command

In z4, JES2 start processing was enhanced to support a MEMBER= value specified via PARM= on the START command or within the PROC. This operand specifies a member of the MVS logical PARMLIB concatenation that contains the JES2 initialization stream.

If no source of initialization statements is specified either via MEMBER= nor via a successful OPEN of the HASPPARM DD data set, then the default is the HASxxxx (xxxx = the subsystem name, e.g. JES2) member in the MVS logical PARMLIB concatenation.

Migration Actions

If the systems programmer elects to use the MVS logical PARMLIB concatenation for JES2 parameter statements, you will need to update operations procedures. Local procedures for recovering from errors such as "PARMLIB not found" or JES2 PROC is bad can be updated this function to simplify recovery.

Monitor Commands

In z4, a new monitor has been created to monitor JES2 functions. This monitor has several commands, one in particular used to display information about JES2 when JES2 is non-responsive.

Migration Actions

Update local procedures and educate operators on the use of the $J D STATUS command when JES2 appears to be malfunctioning.

Monitor Alerts

In z4, the JES2 monitor can highlighted messages to report on abnormal conditions in the JES2 address space. These highlighted messages are called "alerts".

Migration Actions

Educate operators on the new alerts and review the documentation for HASP92xx messages and corresponding actions.

JES2 Checkpoint Corruption Message

In z4, JES2 warmstart processing tracks the number of errors encountered during initialization. If the number of errors exceeds a threshold, a HASP863/HASP272 message pair is issued to query whether initialization should be allowed to complete. This query is made before committing changes to the JES2 checkpoint.

Migration Actions

Update local procedures to describe what should be done if this message occurs. The procedure should state that initialization should not continue unless the operator is certain that the correct checkpoint and SPOOL volumes are in use.

JES2 Use of 4 digit message numbers

Starting in z4, JES2 uses message ids with 4 digit numbers (HASPnnnn).

Migration Actions

Ensure that automation takes into account that JES2 message ids can now have 4 digits.

Application Development

// XMIT JCL Statement

In z4, JES2 provides support for the // XMIT JCL statement to route jobs for execution at other nodes. Older methods of routing jobs are still supported.

Migration Actions

JCL can be updated that will allow the same stream to be used in JES2 as can be used in JES3. You may need to update applications that scan JCL to account for the presence and function of the new JCL statement.

Original Record Length passed via FSI

In prior releases, original line length was not passed to FSS applications via the FSI. This causes a problem when blank truncation was used for a SYSOUT class and the data being processed was not EBCDIC (binary streams). Beginning in z4, the original record length is made available for each record passed via the FSI. The length is returned in the IDX (mapping macro IAZIDX).

Migration Actions
Any FSS application which can process binary data streams should be updated to take advantage of the new data.

SAPI JMR

In z4, JES2 has been updated to pass the JMR field JMRUSEID to the application via the SSS2USID field of the IAZSSS2.

Migration Actions
SAPI applications can use the field when generating SMF records.

SPOOL Read SSI

In z4, JES2 has been enhanced to provide an interface to the SPOOL SSI that will allow the return of in storage buffers.

Migration Actions
Use the facility if needed.

Problem Determination and Diagnosis

Changes in System Trace Contents

In z4, many uses of SVC 111 have been converted to stacking PC instructions. In addition JES2 uses SVC 92 instead of SVC 0 to write data to spool for executing jobs.

Migration Actions
Be aware of these changes when debugging problems via the system trace.

Provide for JES2 IPCS Support

Attention:

- You must be proactive to install JES2 IPCS.
- Make sure JES2 IPCS support works before you experience an emergency.
- Make sure you have JES2 IPCS support for all combinations of JES2 and MVS releases in production. (*z/OS JES2 Migration* and *z/OS JES2 Diagnosis*provide information about JES2 IPCS support.)
- When JES2 dumps are being diagnosed on the same level of the system as the system on which the dumps were taken, do the following:
  - Ensure that SHASPARM is specified in the PARMLIB concatenation.
  - Ensure that SHASMIG is specified in the STEPLIB concatenation.
  - Ensure that SHASPNL0 is specified in the ISPPLIB concatenation.

  This ensures that IPCS can find the SMP/E-maintained copy of JES2 data.
- If the JES2 levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, but the level of MVS is the same, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also specify your PARMLIB data set, plus any other data sets containing IPCS parameters that you normally need when using IPCS.
  - Specify the SHASMIG library that corresponds to the JES2 level in the dump in the first position in the STEPLIB concatenation.
  - Specify the SHASPNL0 library that corresponds to the JES2 level in the dump in the first position in the ISPPLIB concatenation.
- If both the JES2 and system levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also, specify the PARMLIB data set, plus any other data sets containing IPCS parameters that correspond to the system level in the dump.
  - Specify the SHASMIG and MIGLIB libraries that correspond to the system in the dump in the first and second positions in the STEPLIB concatenation.
  - Specify the SHASPNL0 and SBLSPNL0 libraries that correspond to the system in the dump in the first position in the ISPPLIB concatenation.

Notes:

1. IPCS requires libraries in addition to the libraries that JES2 | uses. You may need to make other concatenations. For more information see *z/OS TSO/E Customization*.
2. If dump analysis leads into data structures for other elements or products, you might need to add more data sets to the concatenations to complete the diagnosis. Refer to the appropriate product publications for information about establishing the correct IPCS environments for diagnosing other elements and products.

Support for JES2 Trace IDs 32 and 33

Trace IDs 32 and 33 have been added in z4. Trace id 32 is created whenever a JOE is removed ($#REM). The trace has the work JOE and the exit number in control.

Trace ID 33 is created whenever an NJE header or trailer or NMR is transmitted or received over NJE or SPOOL offload. TRACE=YES has to be specified on the line or node for which trace is desired. The data included in the trace is the header/trailer/NMR in an uncompressed non compacted format.

Migration Actions

Be aware of the additional diagnostic information provided by the new trace IDs.

JES2 Installation Modifications

None

Chapter 5. Migration Actions: JES2 z/OS V1R1 to JES2 z/OS V1R4

Customization

New and Changed Installation Exits

If your exits use JQE or JOE chaining fields or reference job number fields, then **z2 Mode** changes require you to update your exits. Field names have been changed to help identify where logic changes are needed.

Exit 5

In z/OS V1R2 (HJE7705), the command syntax migration aid exit 5 routine, HASX5CTR in module HASX05C, is no longer automatically installed and enabled. As of HJE7705, the exit is now shipped as a sample in SYS1.SHASSAMP and is no longer supported. The function of this module was intended only as a migration aid to translate pre-OS/390 JES2 Version 2 Release 4 command syntax to OS/390 JES2 Version 2 Release 4 command syntax.

Migration Actions: IBM recommends you use the JES2 command syntax as documented in *z/OS JES2 Commands*. If you still need command translation, you need to install and enable EXIT 5 (HASX05C).

Note: The function of HASX05C is not altered from JES2 OS/390 V2R4. See Command Syntax Migration Aid Removal in the "Operations" section of this chapter for more information.

Exit 14 and Exit 49

PITCLASS and QGTCLST are changed.

In z/OS V1R2, QGTCLST now points to a work area, which contains the class list for this selection; this class list is a subset of the classes listed in PITCLASS. PITCLASS contains the class list for the initiator as defined by the INIT(nnnn) initialization statement; QGTCLST contains the class list for this selection and can contain different values than PITCLASS.

Migration Actions: If exit 14 and exit 49 in your installation reference the fields PITCLASS or QGTCLST, you need to examine these references and make appropriate changes.

Exit 4

In z4, JES2 now supports the // XMIT JCL statement as a means of transmitting jobs to another node. This is in addition to the older /*XMIT , /*ROUTE XEQ and /*XEQ JECL statements

Migration Actions: Examine any exit 4 code that might be scanning for the older JECL statements that affect job execution routing to see if they should be updated to also scan for the JCL statement.

Exit 9

In z4, JES2 performs excession processing for every record. In prior releases this was done only when a spool buffer filled. As a consequence, in z4, exit 9 will be given control immediately when the limit is exceeded, not just when the spool buffer filled.

Migration Actions: Exits have been known to give a low increment (for example a value of 1) knowing that the exit will not be driven until the next spool buffer is filled. That is no longer the case.

Exit 22

In z4, the processing for STATUS/CANCEL has been changed to use data residing in a data space. This resulted in:

- A new parameter list for the exit.
- The passing of a address (and ALET) of a block ($STAC) residing in a data space.

Migration Actions: Exit 22 must be modified to use the new parameter list and to use the mechanism for accessing data space areas. See *z/OS JES2 Installation Exits* for further information.

Exit 44

In z4, JES2 no longer polices duplicate logons due to the changes in End of Memory processing. This was done in order to make TSO userids capable of logging on even though JES2 has not yet completed processing the (JES2) resources owned by the address space.

Migration Actions: Use sample exit HASX44SB in module HASX44A to reinstate the functionality of prior releases if that is needed.

Field, Field Name, and Control Block Changes

For JES2 to use up to 999,999 job numbers requires that you activate your JES2 z/OS V1R2 MAS to **z2 Mode** . Unless this activation is performed, JES2 operates in **R4 Mode**, and JES2 z/OS V1R2 behaves with the job number limitations to 65534.

Migration Actions

> **Attention !**
>
> In migrating to JES2 z/OS V1R2, it is important to refer to "General Field and Control Block Changes" on page 19 for the needed details about all specific fields, field names, and control block changes that take place both when migrating to JES2 z/OS V1R2 and when you activate.

In preparing for your migration to JES2 z/OS V1R2 you need to analyze:

- All your exit routines and "unique" code that would be affected by activating JES2 z/OS V1R2 in **z2 Mode**
- How your JES2 system processess the JOE, JQE, JQA, and JOT control blocks that are significantly changed in **z2 Mode**.
- Control blocks and fields used to store job numbers (such as the $SJB,$ JIB, $FSAXB, and $COMWORK).
- How to use JES2 macro services, such as $QJQE, $#JOE, $DOGJQE, $JBIDBLD.
- Your assembler errors that result from not thoroughly finding all places where fields, field names, and control block changes affect JES2.
- Your JES2 exits and how they are affected, making necessary changes, re-assembling, and integrating them.
- If your exits use JQE or JOE chaining fields or reference job number fields, then **z2 Mode** changes require updates to your exit routines. Field names have been changed to help identify where logic changes are needed.

Macro Updates Relating to JOE and JQE Changes

Some macros are created and updated to allow JES2 exit routines to be mode insensitive.

$QJQE and $#JOE

These macros process the JQE and JOE control block chains using proper chaining fields based on the current checkpoint processing mode (either **R4 Mode** or **z2 Mode**. Therefore, users of these macros do not have to be sensitive to which JES2 processing mode is currently being used.

$#JOE is changed to process chains (such as the CHAR JOE chain). Both $QJQE and $#JOE have improved their loop control and have usability enhancements.

$DOGJQE

The JQA returned by $DOGJQE always reflects the **z2 Mode** processing format. Code that processes an artificial JQE (JQA) need not examine the JES2 processing mode (**z2 Mode** or **R4 Mode**) to examine JQA fields.

$JQEJNUM

$JQEJNUM is a new macro used to obtain the binary job number for a particular JQE. This macro operates in **z2 Mode** and **R4 Mode** and processes the JQA or JQE that is passed.

$JBIDBLD

This macro converts a binary job number to a printable jobid. $JBIDBLD is updated to accept a JQE (or JQA) as input and operates in **z2 Mode** or **R4 Mode**. $JBIDBLD formats the jobid, in the correct way, based on the current job number range.

Migration Actions

- Review your installation exit routines and thoroughly understand what changes you might need to make because of the various field, field names, and control block changes.
- Convert any "unique" installation code you use to use the new fields and field names.
- Process JOEs and JQEs as much as possible with the macros provided ($QJQE, $#JOE, $DOGJQE, $JQEJNUM, $JBIDBLD) instead of formulating your own unique code.
- Ensure that all the field and control block changes identified with JES2 z/OS V1R2 are incorporated, including updates to the surrounding logic.

$XMPOST Macro

- $XMPOST

The $XMPOST macro is changed. When you specify $XMPOST XMPLIST=YES (or leave XMPLIST off and let it default), the mapping of the input parameter list is changed. The expected parameter list mapping is as follows:

- +0 Error return address
- +4 Address of ECB to be POSTed
- +8 Address of related ASCB
- +C ECB to be POSTed (if LOSTPOST=CHECK is specified on the macro call)

$QLOCNXT Macro

$QLOCNXT finds the next allocated job number after a specified job number.

Control Block Changes

$PIT

In z4, installations have the option of specifying zero JES2 initiators (i.e. use just WLM managed initiators). This is done by specifying INITDEF PARTNUM=0. The result is that $PITABLE and CCTPIT can have a value of zero since no PITs exist.

Migration Actions: Examine your exits for references to $PITABLE and CCTPIT and ensure that a value of zero is being handled.

$PSO

In z4, the $PSO has been moved to a data space

Migration Actions: Examine your exits for references to the PSO data area and update appropriately.

Executable Macro Updates

$#BUSY

In z4, the $#BUSY service was enhanced to perform $#POST when BUSY=OFF is specified and to perform $CKPT as appropriate.

Migration Actions: Examine all $#BUSY calls and remove unnecessary uses of $#POST or $CKPT. Failure to remove unnecessary $#POSTs will result in performance degradation.

Initialization

Migrating to z2 Mode

Several installation actions are required to migrate to **z2 Mode**:

- Stabilize all MAS members running JES2 z/OS V1R2 in **R4 Mode**
- $ACTIVATE LEVEL=z2 without changing any limits
- Only when your installation is stable, running in **z2 Mode**, consider increasing limits.

Large Volume Support

JES2 supports relative and absolute track addressing for processing the SPOOL. An initialization option indicates whether relative addressing is used, never used, or only used when needed. This option ensures that relative addressing is not used before any vendor product that looks at SPOOL is ready to use relative addressing.

The SPOOLDEF initialization statement is updated to allow you to specify whether you "always", "as needed", or "never" want to use relative MTTRs in processing the JES2 SPOOL. You use the new keyword, RELADDR= with a value of ALWAYS , ASNEEDED, or NEVER to specify your choice.

Notes:

1. Relative addressing in processing the JES2 SPOOL is only used if all members of the MAS support relative addressing. Any member that is not supporting relative addressing fails to initialize and a HASP401 message results.
2. The size of a SPOOL data set is still limited to a size of 64K tracks.

The $T SPOOLDEF command is updated to allow you to update the current setting for RELADDR=. If relative addressing is being used for a volume, the $D

SPOOL,UNITDATA displays a new keyword, BASETRAK=. The value of BASETRAK= must be added to the TT (part of MTTR) to convert a relative track address to an absolute track address.

A new interface is provided to read records from JES2 SPOOL. This interface is implemented as an option on the SSI 71 call. When you use this interface, you need not know about internal JES2 MTTR processing.

Migration Actions
- Understand how to use the SPOOLDEF initialization keyword, RELADDR= to specify the absolute or relative track addressing for the SPOOL.
- Understand how to use the new interface to read records from the JES2 SPOOL.
- Examine your code to ensure that the code the reads the JES2 SPOOL is consistent with this function.

Dynamic PROCLIBs Support

Customer IPL outages are aggravated by a defective JES2 PROC. JES2 z/OS V1R2 addresses this problem in "3" ways:
- Minimizes the information in the JES2 PROC (thus limiting the chance that some change in the system makes the PROC unusable).
- Provides the ability to start JES2 with a generic PROC, such as IEASYSAS for those cases where the JES2 PROC does become unusable. The two most frequently updated statements in the JES2 PROC are the PROCLIB data set concatenations and the JES2 initialization data sets. Now, JES2 provides initialization statements for these two data sets.
- Provides PROCLIB and INCLUDE initialization statements. The PROCLIB initialization statement allows you to define a logical DD name that the converter can use. Up to 255 data set names can be associated with each PROCLIB initialization statement. The INCLUDE initialization statement gives you the ability to "include" additional initialization streams into your initialization process.

Additionally, the following commands provide additional flexibility in specifying PROCLIB data sets:
- $ADD PROCLIB creates a new dynamic PROCLIB concatenation.
- $DEL PROCLIB deletes an existing dynamic PROCLIB. This command does not affect the static PROCLIBs defined in the JES2 PROC.
- $T PROCLIB updates an existing dynamic PROCLIB
- $D PROCLIB displays the current dynamic PROCLIBs only and does not display the static ones. This command has a DEBUG option that provides additional information about the dynamic PROCLIB concatenation as well as logically deleted dynamic PROCLIBs that are still in use.

Migration Actions
- Review your JES2 PROC.
- Review/Update your JES2 initialization statements appropriately.
- Take advantage of the PROCLIB and INCLUDE initialization statements to create dynamic PROCLIB concatenations and "include" initialization streams into your JES2 startup.

JES2 Support of a Maximum of 999,999 Job Numbers

With **z2 Mode**, you will be able to define up to 999,999 job numbers. Currently, the limit is just 65,534. In order to support the larger job numbers, the jobid format is changed. The format used depends on the upper limit on the RANGE operand of the JOBDEF initialization statement.

- If the upper limit is > 99,999, then the jobid is Jxxxxxxx
- If the upper limit is <= 99,999 then the jobid is JOBxxxxx
- If the upper limit is > 99,999, then the jobid is Txxxxxxx
- If the upper limit is <= 99,999 then the jobid is TSUxxxxx
- If the upper limit is > 99,999, then the jobid is Sxxxxxxx
- If the upper limit is <= 99,999 then the jobid is STCxxxxx

Note: If you are in **z2 Mode** and RASSIGN=YES is specified (or defaulted) on the JOBDEF initialization statement, the NJE jobs could use the Jxxxxxxx form of jobid if the job number is > 99,999; this occurs regardless of the RANGE setting.

Migration Actions

ATTENTION:

You should review the general description of activating your JES2 MAS to **z2 Mode** as well as the fields, field names, and control block changes that occur when running in **z2 Mode**. Reread "z2 Mode for $ACTIVATE" on page 17 and "General Field and Control Block Changes" on page 19 for the details.

- Review how your installation processes jobs/job ids. Ensure your automation adapts to potential jobids up to 999,999.
- Have ready back out procedures.
- Adjust your JOBDEF initialization parameters accordingly.
- Consider setting RASSIGN=NO.

Spinning JESlog Data Sets for Long Running Jobs

You initialize JES2 to allow the installation to "spin" or "suppress" JESlog data sets as the job runs by using the JESLOG= keyword on the JOBCLASS initialization statement.

Migration Actions

- Examine your usage of JESMSGLG and JESYSMSG data sets.
- Review initialization parameters (JOBCLASS) and JOB JCL.
- Makes changes you desire for "spinning" or "suppressing" JESlog message output.
- Review/Update automation to decide whether to use $T Jxxxx,SPIN.
- If you use Spool Data Set Browse, examine your use and be aware of new function.

Checkpoint Improvements

Prior to JES2 z/OS V1R2, switching between checkpoint modes required an all member warm start. Now installations can switch checkpoint modes using the $T CKPTDEF,MODE=DUAL|DUPLEX command. An all member warm start is not required. Changing to DUPLEX mode is always allowed. Changing to DUAL mode is NOT

allowed when either CKPTn or NEWCKPTn is on the coupling facility or if DUPLEX=OFF is specified on the CKPTDEF initialization statement and both CKPTs are in use.

Note: You can use the $T CKPTDEF,MODE=DUAL|DUPLEX command even though some MAS members are not at the JES2 z/OS V1R2 release.

Migration Actions
- Update your procedures that change the MODE.

INCLUDE Statement

In z4, the INCLUDE statement has been enhanced to allow initialization statements to be read from the MVS logical PARMLIB concatenation. A second enhancement is that the DSN need not be supplied on an INCLUDE statement if the member being requested is in the same data set as the INCLUDEing member.

Migration Actions
Examine the initialization streams and update appropriately.

PARM operand on START command

In z4, JES2 start processing was enhanced to support a MEMBER= value specified via PARM= on the START command or within the PROC. This operand specifies a member of the MVS logical PARMLIB concatenation that contains the JES2 initialization stream.

If no source of initialization statements is specified either via MEMBER= nor via a successful OPEN of the HASPPARM DD data set, then the default is the HASxxxx (xxxx = the subsystem name, e.g. JES2) member in the MVS logical PARMLIB concatenation.

Migration Actions
If you have a member in the MVS logical PARMLIB with the default name mentioned above, take the appropriate action.

PARTNUM=0 on INITDEF statement is Now Supported

In z4, PARTNUM=0 is supported. This means that no JES2 initiators will be used. This in turn means that there is no PITs and $PITABLE and CCTPIT will be zero.

Migration Actions
If you decide to specify INITNUM=0, verify that no PITs is an acceptable condition.

New Monitor Address Space

In z4, JES2 starts a new (non-optional) address space for each JES2 instance called xxxxMON (xxxx = subsystem name e.g. JES2) to monitor the corresponding J3S2 address space.

Migration Actions
Be aware of the existence of a new system address space.

Security Considerations

New Profiles for monitor commands

In z4, JES2 monitor was provided. The monitor has a set of commands used to give information about the status of the JES2 address space. The entity name of these monitor commands in the CMDAUTH class is of the form: xxxxMON.action.object where xxxx is the subsystem name (e.g. JES2). The complete list of entity names for the JES2 monitor commands can be found in *z/OS JES2 Initialization and Tuning Guide*.

Migration Actions

New profiles may be required for RACF in order to allow monitor commands to be issued.

Operations

Z2 Mode

Use the JOBDEF initialization statement to increase the amount of job numbers you can have in your installation. In **z2 Mode**, JES2 allows you to define up to 999,999 job numbers (expanding the limit from 65534).

With **z2 Mode**, you will be able to define up to 999,999 job numbers. Currently, the limit is just 65,534. In order to support the larger job numbers, the jobid format is changed. The format used depends on the upper limit on the RANGE operand of the JOBDEF initialization statement.

- If the upper limit is > 99,999, then the jobid is Jxxxxxxx
- If the upper limit is <= 99,999 then the jobid is JOBxxxxx
- If the upper limit is > 99,999, then the jobid is Txxxxxxx
- If the upper limit is <= 99,999 then the jobid is TSUxxxxx
- If the upper limit is > 99,999, then the jobid is Sxxxxxxx
- If the upper limit is <= 99,999 then the jobid is STCxxxxx

Note:

> JES2 uses the appropriate jobid format in displays of job information. There is a transition period after JOBDEF RANGE= is increased above 99,999 where the "old" format jobids may still be displayed. This can occur for jobs that were created when the old range was being used. Until these jobs are purged from the system, the old format for jobids may be displayed as part of informational messages about those specific jobs.

See *z/OS JES2 Initialization and Tuning Reference* for the specific changes to the JOBDEF initialization statement

Operator commands that process jobs by job number are improved to handle the increase of job numbers.

$T JOBDEF,RANGE= is updated to allow you to set a job number limit up to a maximum of 999,999. Also, the following $T command changes allow you to take advantage of new maximum for the number of jobs, number of JOEs and number of BERTs as well as to decrease their settings:

- $T JOBDEF JOBNUM= for the number of jobs

- $T OUTDEF JOENUM= for the number of JOEs
- $T CKPTSPACE BERTNUM= for the number of BERTs

JOBID Formats

The format of job numbers being displayed as part of command responses or messages has changed in z2 mode. Currently job numbers are displayed as JOBnnnnn. Five characters is not large enough to display job numbers greater than 99,999. Therefore, job numbers will be displayed differently when job numbers greater than 99,999 are used or potentially used.

The "new" job number format will be as follows: If the maximum allowed job number (high value of the JOBDEF RANGE= statement) is above 99,999 then the job number format will be J0nnnnnn. This format will always be used unless the job number range is decreased below 100,000. Similarly, STCnnnnn will become S0nnnnnn and TSUnnnnn will become T0nnnnnn.

If the job number range is increased above 99,999 or decreased below 100,000 with the $T command, a transition period will exist where the old job number format and the new job number format will be displayed as part of commands, messages, and SMF records.

Rules for determining job number format:
- If the local maximum is > 99,999 then use the new format.
- If the local maximum is <= 99,999 then use the old format as long as the job number being displayed is<= 99,999.

Note: If you are in **z2 Mode** and RASSIGN=YES is specified (or defaulted) on the JOBDEF initialization statement, the NJE jobs could use the Jxxxxxxx form of jobid if the job number is > 99,999; this occurs regardless of the RANGE setting.

Migration Actions
- Reread "z2 Mode for $ACTIVATE" on page 17 to understand how to use the $ACTIVATE command to bring your JES2 to **z2 Mode** and what are the changes to JES2 job number processing.
- Examine and change your automation with respect to commands, messages, and job numbers.
- Implement appropriate procedures to return to **R4 Mode**.

Spinning JESlog Data Sets for Long Running Jobs

Spinning or suppressing JESlog data described in "Spinning JESlog Data Sets for Long Running Jobs" on page 27. Refer to this section for an explanation of this function. A description of migration actions follow.

Migration Actions
- Examine your JOBCLASS(v) initialization statements and change as appropriate to use this function.
- Your installation workload of jobs might require the use of JESLOG.

Attention:

- Jobs that are converted on a JES2 z/OS V1R2 MAS member but run on a pre-JES2 z/OS V1R2 MAS member cannot perform the JESLOG spin off.
- Jobs that are converted on a JES2 z/OS V1R1 (or earlier) MAS member cannot perform JESLOG spin off regardless of which MAS member they run on.
- The converter (and START commands) cannot recognize the JESLOG in a JES2 z/OS V1R1 (or earlier) MAS member. This condition produces an error.

Examine how your installation responds to the following messages:
- HASP138
- HASP860
- HASP130

See *z/OS JES2 Messages* for details.

Dynamic JES2 PROCLIB Commands

JES2 z/OS V1R2 provides a PROCLIB initialization statement you use to define a logical DD name that the converter can use. Up to 255 data set names can be associated with the PROCLIB initialization statement. Dedicated PROCLIB data set concatenations can be eliminated.

Migration Actions

- Use the PROCLIB initialization statements in your initialization stream.
- Understand the following commands:.
  - $ADD PROCLIB creates a new dynamic PROCLIB concatenation.
  - $DEL PROCLIB deletes an existing dynamic PROCLIB. This command does not affect static PROCLIBs defined in the JES2 PROC.
  - $T PROCLIB updates an existing dynamic PROCLIB
  - $D PROCLIB displays the current dynamic PROCLIBs only and does not display the static ones. This command has a DEBUG option that provides additional information about the dynamic PROCLIB concatenation as well as logically deleted dynamic PROCLIBs that are still in use.
- This function can operate independent of mode (**z2 Mode** or **R4 Mode**).

See *z/OS JES2 Initialization and Tuning Reference* and *z/OS JES2 Commands* for more details.

Large Volume Support

JES2 supports both a relative and absolute track addressing for SPOOL. The SPOOLDEF initialization statement is updated to allow you to specify whether you **always**, **as needed**, or **never** want to use relative MTTRs in processing the JES2 SPOOL. You use the new keyword, RELADDR= with a value of ALWAYS , ASNEEDED, or NEVER to specify your choice.

Notes:

1. Relative addressing in processing the JES2 SPOOL is only used if all members of the MAS support relative addressing. Any member that is not supporting relative addressing fails to initialize and a HASP401 message results.
2. The size of a SPOOL data set is still limited to 64K tracks.

Migration Actions

- Examine the `RELADDR=` on the SPOOLDEF initialization statement to ensure it is appropriate for the volume. Make adjustments as necessary.
- Ensure that all members of the MAS use relative addressing.
- This support works in **z2 Mode** or **R4 Mode**.

Checkpoint Improvements

Installations can switch checkpoint modes using the `$T CKPTDEF,MODE=DUAL|DUPLEX` command. An all member warm start is not required. Changing to `DUPLEX` mode is always allowed. Changing to `DUAL` mode is NOT allowed when either CKPTn or NEWCKPTn is on the coupling facility or if DUPLEX=OFF is specified on the CKPTDEF initialization statement and both CKPTs are in use.

Note: You can use the `$T CKPTDEF,MODE=DUAL|DUPLEX` command even through some MAS members are not at the JES2 z/OS V1R2 level.

Migration Actions

- Update your procedures that change the `MODE`.

HASP443 Message

The HASP443 message clarifies the reason for a SPOOL allocation failure.

Migration Actions

- Examine any automation that processes this message.

Command Syntax Migration Aid Removal

In JES2 OS/390 V2R4, JES2 provided a command syntax compatibility and migration aid in the form of an automatically installed exit 5 routine. The migration aid was intended as a temporary bridge to give customers time to adapt to the command syntax changes made in OS/390 JES2 Version 2 Release 4. As of JES2 z/OS V1R2, the migration aid is no longer automatically installed and enabled. If you are not using the command syntax documented in *z/OS JES2 Commands* and are still relying on the migration aid, you can experience erroneous system operation or receive HASP649, HASP650, HASP003 messages.

The following table lists the commands translated and the results of that translation.

Table 18. Command Translation Formats

JES2 OS/390 V2R4 Format	JES2 z/OS V1R2 Format
`$D'name',...`	`$DJOBQ'name',CMDAUTH=*,...`
`$T'name',...`	`$TJOBQ'name',...` **(See NOTE 1)**
`$DJ1,2,...`	`$DJ(1,2),...` **(See NOTES 2,3)**
`$DJ1-2,J3-4,...`	`$DJ(1-2,3-4),...` **(See NOTE 3)**
`$LJnnn,ALL`	`$DOJnnn`
`$LJnnn,H`	`$DOJnnn,HELD`
`$LJnnn,READY`	`$DOJnnn,READY`
`$LJnnn,OUTGRP=xxx`	`$DOJnnn,OUTGRP=xxx`
`$CJnnn,OUTGRP=xxx`	`$COJnn,OUTGRP=xxx`
`$PJnnn,OUTGRP=xxx`	`$POJnnn,OUTGRP=xxx` **(See NOTE 4)**

Table 18. Command Translation Formats (continued)

JES2 OS/390 V2R4 Format	JES2 z/OS V1R2 Format
`$PJnnn,Q=x`	`$POJnnn,Q=x`
`$vJnnn,A=|Days=|Hours=`	`$vJnnn,A>|Days>|Hours>`
`$TJnnn,S=sid1,sid2,...`	`$TJnnn,S=(sid1,sid2,...)`
`$DSPL,JOBS=nn`	`$DJOBQ,SPOOL=(PERCENT>=nn)`
`$SSPL,V=xxxxxx,...`	`$SSPL(xxxxxx),...`
`$vIxx`	`$vI(xx)`
`$vIxx-yy`	`$vI(xx)` **(See NOTE 5)**
`$vJxxxx,Txxxx...`	`$vJOBQ(xxxx,xxxx)` **(See NOTE 6)**
`$TIxx,class-list`	`$TI(xx),C=class-list`
`$HQ,ALL`	`$TJOBCLASS(*),QHELD=Y`
`$HQ,C=xyz`	`$TJOBCLASS(x,y,z),QHEL=Y`
`$AQ,ALL`	`$TJOBLCASS(*),QHELD=N`
`$AQ,C=xyz`	`$TJOBCLASS(x,y,z),QHELD=N`
`$PQ,ALL,...`	`$POJOBQ,READY,...`
`$PQ,Q=xyz`	`$POJOBQ,READY,Q=xyz`
`$OQ,ALL,...`	`$OJOBQ,/R=LOCAL.*,...`
`$OQ,Q=xyz,...`	`$OJOBQ,/R=LOCAL.*,/Q=xyz,...`
`$TALL,sid1,sid2,...`	`$TJOBQ(*),/S=(sid1,sid2,...)`
`$LSYS`	`$DMEMBER`
`$ESYS,sid`	`$EMEMBER(sid)`
`$ESYS,RESET=sid`	`$ECKPTLOCK,HELDBY=sid`
`$TSYS,IND=Y|N`	`$TMEMBER(local),IND=Y|N`

Notes:

1. Changes are similar for $A, $C, $E, $H, $L, $O, $P, $T, $TO
2. J can be J, JOB, S, STC, T, TSU
3. Changes are similar for $A, $C, $E, $H, $L, $O, $P, $TO
4. Unless Q= is a valid job queue (XEQ, PPU, etc.)
5. Where v is one of the command verbs D, P, S, T, or Z. This command processed a named range of initiators. You must now issue these commands as separate commands.
6. Where v is one of the commands verbs A, C, E, H, L, O, P, T, TO. This command processed a list of JOBs, TSO users, and STCs. You must now issue these commands either as multiple commands ($v Jxxxx... and $v Txxxx...) or use $v JOBQ(xxxx,xxxx). The JOBQ version has slightly different function (does not check job type).

Migration Actions

If you are exclusively using the command syntax documented in the *z/OS JES2 Commands* publication and are not relying on the command migration aid exit, then no action is required. If you do not know whether or not you are currently relying on the command migration aid exit, you need to do the following:

1. First determine if the migration exit is installed. Issue $DEXIT5 on your current production systems and if the command response does not include routine name HASX5CTR, then no further action is required.
2. If the exit is installed, look through SYSLOG for $HASP006 messages which display translated commands. The fact that commands are translated does not

necessarily indicate dependence on the migration aid exit because it conservatively translates several command formats that already have valid syntax. Commands with verbs with a single subscript or subscript range that are enclosed in parentheses are frequently unnecessarily translated as shown in the following examples:

```
$HASP006 $DJ5   TRANSLATED BY EXIT HASX05C TO $DJ(5)
$HASP006 $DJ5-6 TRANSLATED BY EXIT HASX05C TO $DJ(5-6)
```

If however, you find a $HASP006 message in SYSLOG with any other form, or in the above form but having additional operands that are altered, then you are currently dependent on the migration aid exit. If you find you are currently dependent on the migration aid exit, then determine the origin of the obsolete command syntax and take corrective action. Either:

- Have your operators begin using only the supported command syntax and convert your automation scripts to use valid syntax. Refer to the *z/OS JES2 Commands* for correct syntax of JES2 commands. IBM recommends you begin this task immediately on your current production systems and analyze $HASP006 messages in SYSLOG to measure your progress.

  OR

- Although IBM strongly recommends against it, install the command migration aid exit yourself and assume the risks described in the **CAUTION** below. As of HJE7705, the command migration aid exit is now shipped as member HASX05C in SYS1.SHASSAMP.

> **CAUTION !**
> **The JES2 command migration aid, exit 5 routine HASX5CTR, is now shipped in member HASX05C of SYS1.SHASSAMP and is no longer supported. This exit is not warranted and is functionally stabilized. You are strongly urged to start eliminating your dependency on this exit now and stop using the exit as soon as possible. Failure to make an orderly transition now could result in a prematurely rushed transition later.**

TSO GR, TSO SEND Commands and TSO Logons

In z4, JES2 discontinued policing TSO logons. Duplicate instances of a given userid can be logged on at the same time. It is highly recommended that the GRS RNL list be updated to convert SYSIKJUA to a SYSTEMS ENQ in order to prevent duplicates. If the RNL is not updated, and duplicate instances of a userid are logged on, then TSO GR and SEND messages to the duplicately logged on userids will not function.

Migration Actions

Ensure that the procedure outlined in the *z/OS TSO/E Customization* is followed in order to block duplicate logons.

PARM operand on START command

In z4, JES2 start processing was enhanced to support a MEMBER= value specified via PARM= on the START command or within the PROC. This operand specifies a member of the MVS logical PARMLIB concatenation that contains the JES2 initialization stream.

If no source of initialization statements is specified either via MEMBER= nor via a successful OPEN of the HASPPARM DD data set, then the default is the HASxxxx (xxxx = the subsystem name, e.g. JES2) member in the MVS logical PARMLIB concatenation.

Migration Actions

If the systems programmer elects to use the MVS logical PARMLIB concatenation for JES2 parameter statements, you will need to update operations procedures. Local procedures for recovering from errors such as "PARMLIB not found" or JES2 PROC is bad can be updated this function to simplify recovery.

Monitor Commands

In z4, a new monitor has been created to monitor JES2 functions. This monitor has several commands, one in particular used to display information about JES2 when JES2 is non-responsive.

Migration Actions

Update local procedures and educate operators on the use of the $J D STATUS command when JES2 appears to be malfunctioning.

Monitor Alerts

In z4, the JES2 monitor can highlighted messages to report on abnormal conditions in the JES2 address space. These highlighted messages are called "alerts".

Migration Actions

Educate operators on the new alerts and review the documentation for HASP92xx messages and corresponding actions.

JES2 Checkpoint Corruption Message

In z4, JES2 warmstart processing tracks the number of errors encountered during initialization. If the number of errors exceeds a threshold, a HASP863/HASP272 message pair is issued to query whether initialization should be allowed to complete. This query is made before committing changes to the JES2 checkpoint.

Migration Actions

Update local procedures to describe what should be done if this message occurs. The procedure should state that initialization should not continue unless the operator is certain that the correct checkpoint and SPOOL volumes are in use.

JES2 Use of 4 digit message numbers

Starting in z4, JES2 uses message ids with 4 digit numbers (HASPnnnn).

Migration Actions

Ensure that automation takes into account that JES2 message ids can now have 4 digits.

Application Development

APPLCOPY No Longer Supported

APPLCOPY is no longer supported.

APPLCOPY is a way to access data in the checkpoint, but is replaced with checkpoint versions. (SSI 71) applications that used APPLCOPY (such as SDSF) have migrated to now use checkpoint versions. Becuase of installation overhead

concerns, especially with very large checkpoint data sets, support for APPLCOPY has been discontinued. The CKPTDEF APPLCOPY= keyword is dropped and using it will generate an error during initialization (causing other operands on CKPTDEF to be ignored).

Migration Actions

- Installations should delete references to APPLCOPY in their initialization decks.
- Installations that have an application that still uses APPLCOPY, need to have that application migrate to the use of checkpoint versions before installing JES2 z/OS V1R2.

// XMIT JCL Statement

In z4, JES2 provides support for the // XMIT JCL statement to route jobs for execution at other nodes. Older methods of routing jobs are still supported.

Migration Actions

JCL can be updated that will allow the same stream to be used in JES2 as can be used in JES3. You may need to update applications that scan JCL to account for the presence and function of the new JCL statement.

Original Record Length passed via FSI

In prior releases, original line length was not passed to FSS applications via the FSI. This causes a problem when blank truncation was used for a SYSOUT class and the data being processed was not EBCDIC (binary streams). Beginning in z4, the original record length is made available for each record passed via the FSI. The length is returned in the IDX (mapping macro IAZIDX).

Migration Actions

Any FSS application which can process binary data streams should be updated to take advantage of the new data.

SAPI JMR

In z4, JES2 has been updated to pass the JMR field JMRUSEID to the application via the SSS2USID field of the IAZSSS2.

Migration Actions

SAPI applications can use the field when generating SMF records.

SPOOL Read SSI

In z4, JES2 has been enhanced to provide an interface to the SPOOL SSI that will allow the return of in storage buffers.

Migration Actions

Use the facility if needed.

Problem Determination and Diagnosis

Provide for JES2 IPCS Support

> **Attention!**
> - You must be proactive to install JES2 IPCS.
> - Make sure JES2 IPCS support works before you experience an emergency.
> - Make sure you have JES2 IPCS support for all combinations of JES2 and MVS releases in production. (*z/OS JES2 Migration* and *z/OS JES2 Diagnosis* provide information about JES2 IPCS support.)

- When JES2 dumps are being diagnosed on the same level of the system as the system on which the dumps were taken, do the following:
  - Ensure that SHASPARM is specified in the PARMLIB concatenation.
  - Ensure that SHASMIG is specified in the STEPLIB concatenation.
  - Ensure that SHASPNL0 is specified in the ISPPLIB concatenation.

  This ensures that IPCS can find the SMP/E-maintained copy of JES2 data.
- If the JES2 levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, but the level of MVS is the same, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also specify your PARMLIB data set, plus any other data sets containing IPCS parameters that you normally need when using IPCS.
  - Specify the SHASMIG library that corresponds to the JES2 level in the dump in the first position in the STEPLIB concatenation.
  - Specify the SHASPNL0 library that corresponds to the JES2 level in the dump in the first position in the ISPPLIB concatenation.
- If both the JES2 and system levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also, specify the PARMLIB data set, plus any other data sets containing IPCS parameters that correspond to the system level in the dump.
  - Specify the SHASMIG and MIGLIB libraries that correspond to the system in the dump in the first and second positions in the STEPLIB concatenation.
  - Specify the SHASPNL0 and SBLSPNL0 libraries that correspond to the system in the dump in the first position in the ISPPLIB concatenation.

Note: IPCS requires libraries in addition to the libraries that JES2 uses. You may need to make other concatenations. For more information, see *z/OS MVS IPCS Customization*.

If dump analysis leads into data structures for other elements or products, you might need to add more data sets to the concatenations to complete the diagnosis. Refer to the appropriate product publications for information about establishing the correct IPCS environments for diagnosing other elements and products.

HASP443 Message

The HASP443 message is updated to better clarify the reason a SPOOL allocation failed. One or more "reason code" can be displayed with this message to further clarify this conditon.

Migration Actions

- Review how you examine the HASP443 message.
- If you have automation that interprets HASP443, then you might need to make changes.

Auto Reply to Termination WTORs

Automatic responses to HASP070, HASP089, and HASP098 occur if there is no operator response for 10 minutes. See *z/OS JES2 Messages* for message, HASP065 for more detail.

Changes in System Trace Contents

In z4, many uses of SVC 111 have been converted to stacking PC instructions. In addition JES2 uses SVC 92 instead of SVC 0 to write data to spool for executing jobs.

Migration Actions

Be aware of these changes when debugging problems via the system trace.

Provide for JES2 IPCS Support

Attention:

- You must be proactive to install JES2 IPCS.
- Make sure JES2 IPCS support works before you experience an emergency.
- Make sure you have JES2 IPCS support for all combinations of JES2 and MVS releases in production. (*z/OS JES2 Migration* and *z/OS JES2 Diagnosis*provide information about JES2 IPCS support.)
- When JES2 dumps are being diagnosed on the same level of the system as the system on which the dumps were taken, do the following:
  - Ensure that SHASPARM is specified in the PARMLIB concatenation.
  - Ensure that SHASMIG is specified in the STEPLIB concatenation.
  - Ensure that SHASPNL0 is specified in the ISPPLIB concatenation.

  This ensures that IPCS can find the SMP/E-maintained copy of JES2 data.
- If the JES2 levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, but the level of MVS is the same, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also specify your PARMLIB data set, plus any other data sets containing IPCS parameters that you normally need when using IPCS.
  - Specify the SHASMIG library that corresponds to the JES2 level in the dump in the first position in the STEPLIB concatenation.
  - Specify the SHASPNL0 library that corresponds to the JES2 level in the dump in the first position in the ISPPLIB concatenation.
- If both the JES2 and system levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also, specify the PARMLIB data set, plus any other data sets containing IPCS parameters that correspond to the system level in the dump.
  - Specify the SHASMIG and MIGLIB libraries that correspond to the system in the dump in the first and second positions in the STEPLIB concatenation.
  - Specify the SHASPNL0 and SBLSPNL0 libraries that correspond to the system in the dump in the first position in the ISPPLIB concatenation.

Notes:

1. IPCS requires libraries in addition to the libraries that JES2 | uses. You may need to make other concatenations. For more information see *z/OS TSO/E Customization*.
2. If dump analysis leads into data structures for other elements or products, you might need to add more data sets to the concatenations to complete the diagnosis. Refer to the appropriate product publications for information about establishing the correct IPCS environments for diagnosing other elements and products.

Support for JES2 Trace IDs 32 and 33

Trace IDs 32 and 33 have been added in z4. Trace id 32 is created whenever a JOE is removed ($#REM). The trace has the work JOE and the exit number in control.

Trace ID 33 is created whenever an NJE header or trailer or NMR is transmitted or received over NJE or SPOOL offload. TRACE=YES has to be specified on the line or node for which trace is desired. The data included in the trace is the header/trailer/NMR in an uncompressed non compacted format.

Migration Actions
Be aware of the additional diagnostic information provided by the new trace IDs.

JES2 Installation Modifications

Modification Considerations

If you plan modifications to JES2 beyond using exit routines, your migration actions will be similar to those for exit routines. See "z/OS JES2 Version 1 Release 2 Overview" on page 17 and review each item in JES2 z/OS V1R2. The migration actions enumerated for these items will apply to your own special modifications.

Distribution Library Name Changes

Previous library data set names included version, modification, and release values (for example,SYS1.V2R10M0.SHASLINK. In JES2 z/OS V1R2, the *"vrm"* qualifier (which in the example is V2R10M0) is eliminated (For example, SYS1.SHASLINK is the proper name.).

Migration Actions
Change your installations JCL appropriately, including TSO and LOGON PROCs for IPCS.

Counting Null Records

Prior to JES2 z/OS V1R2, null records written to SYSOUT data sets were padded to a single blank record. Because of this padding, these records were counted as records in the job totals. This count is not consistent with SYSOUT received over NJE. Null records from NJE are not padded and not counted in the job totals.

Migration Actions
Since the processing to pad null SYSOUT records was deleted in JES2 z/OS V1R2, counting may change record counts for existing jobs. You would notice this change in a mixed MAS (JES2 z/OS V1R2 running with earlier levels of JES2).

Chapter 6. Migration Actions: JES2 OS/390 V2R8 to JES2 z/OS V1R4

Customization

New and Changed Installation Exits

Installation Exits 11 and 12

Installation exits 11 and 12 are unaffected if you use no new functions in z/OS V1R1 JES2. However, if you turn on spool affinity or set fencing to more than one volume, then your current exits 11 and 12 are affected.

Migration Actions: Review the documentation for exits 11 and 12 in *z/OS JES2 Installation Exits*. Before activating SPOOL affinity, ensure that the correct available SPOOL masks are used to determine what volumes to add to a job in these exits.

Your installation uses spool partitioning if `FENCE=ACTIVE=YES` is specified on the SPOOLDEF initialization statement. Spool affinity is used if any spool volume has something other than SYSAFF=ANY displayed for it.

Installation Exits 0, 19, 24

The HASPINIT load module and the initialization PCE have been moved above the 16M line. This may impact installation exits called during initialization processing (such as exits 0, 19 and 24). If your exit switches to AMODE 24 to invoke DFP modules, then this code will no longer be able to access the initialization PCE (and in particular the save area in the PCE).

Migration Actions: Review all exits for code that switches to AMODE 24 ($AMODE 24 macro invocations). Verify that the switch to 24 bit mode is still required. If it is required and you are in an exit that gets control during initialization, ensure that the code running in AMODE 24 does not access the initialization PCE.

Also, examine all uses of DFP macros, such as GET and PUT. If these macros were hand coded (instead of using the IBM-supplied macro) changes might be necessary. The DFP macros were changed to support 31 bit addressable callers.

Exit 5

In z/OS V1R2 (HJE7705), the command syntax migration aid exit 5 routine, HASX5CTR in module HASX05C, is no longer automatically installed and enabled. As of HJE7705, the exit is now shipped as a sample in SYS1.SHASSAMP and is no longer supported. The function of this module was intended only as a migration aid to translate pre-OS/390 JES2 Version 2 Release 4 command syntax to OS/390 JES2 Version 2 Release 4 command syntax.

Migration Actions: IBM recommends you use the JES2 command syntax as documented in *z/OS JES2 Commands*. If you still need command translation, you need to install and enable EXIT 5 (HASX05C).

Note: The function of HASX05C is not altered from JES2 OS/390 V2R4. See Command Syntax Migration Aid Removal in the "Operations" section of this chapter for more information.

Exit 14 and Exit 49

PITCLASS and QGTCLST are changed.

In z/OS V1R2, QGTCLST now points to a work area, which contains the class list for this selection; this class list is a subset of the classes listed in PITCLASS. PITCLASS contains the class list for the initiator as defined by the INIT(nnnn) initialization statement; QGTCLST contains the class list for this selection and can contain different values than PITCLASS.

Migration Actions: If exit 14 and exit 49 in your installation reference the fields PITCLASS or QGTCLST, you need to examine these references and make appropriate changes.

Exit 4

In z4, JES2 now supports the // XMIT JCL statement as a means of transmitting jobs to another node. This is in addition to the older /*XMIT , /*ROUTE XEQ and /*XEQ JECL statements

Migration Actions: Examine any exit 4 code that might be scanning for the older JECL statements that affect job execution routing to see if they should be updated to also scan for the JCL statement.

Exit 9

In z4, JES2 performs excession processing for every record. In prior releases this was done only when a spool buffer filled. As a consequence, in z4, exit 9 will be given control immediately when the limit is exceeded, not just when the spool buffer filled.

Migration Actions: Exits have been known to give a low increment (for example a value of 1) knowing that the exit will not be driven until the next spool buffer is filled. That is no longer the case.

Exit 22

In z4, the processing for STATUS/CANCEL has been changed to use data residing in a data space. This resulted in:
- A new parameter list for the exit.
- The passing of a address (and ALET) of a block ($STAC) residing in a data space.

Migration Actions: Exit 22 must be modified to use the new parameter list and to use the mechanism for accessing data space areas. See *z/OS JES2 Installation Exits* for further information.

Exit 44

In z4, JES2 no longer polices duplicate logons due to the changes in End of Memory processing. This was done in order to make TSO userids capable of logging on even though JES2 has not yet completed processing the (JES2) resources owned by the address space.

Migration Actions: Use sample exit HASX44SB in module HASX44A to reinstate the functionality of prior releases if that is needed.

Field, Field Name, and Control Block Changes

For JES2 to use up to 999,999 job numbers requires that you activate your JES2 z/OS V1R2 MAS to **z2 Mode** . Unless this activation is performed, JES2 operates in **R4 Mode**, and JES2 z/OS V1R2 behaves with the job number limitations to 65534.

Migration Actions

> **Attention !**
>
> In migrating to JES2 z/OS V1R2, it is important to refer to "General Field and Control Block Changes" on page 19 for the needed details about all specific fields, field names, and control block changes that take place both when migrating to JES2 z/OS V1R2 and when you activate.

In preparing for your migration to JES2 z/OS V1R2 you need to analyze:

- All your exit routines and "unique" code that would be affected by activating JES2 z/OS V1R2 in **z2 Mode**
- How your JES2 system processess the JOE, JQE, JQA, and JOT control blocks that are significantly changed in **z2 Mode**.
- Control blocks and fields used to store job numbers (such as the $SJB,$ JIB, $FSAXB, and $COMWORK).
- How to use JES2 macro services, such as $QJQE, $#JOE, $DOGJQE, $JBIDBLD.
- Your assembler errors that result from not thoroughly finding all places where fields, field names, and control block changes affect JES2.
- Your JES2 exits and how they are affected, making necessary changes, re-assembling, and integrating them.
- If your exits use JQE or JOE chaining fields or reference job number fields, then **z2 Mode** changes require updates to your exit routines. Field names have been changed to help identify where logic changes are needed.

Macro Updates Relating to JOE and JQE Changes

Some macros are created and updated to allow JES2 exit routines to be mode insensitive.

$QJQE and $#JOE
These macros process the JQE and JOE control block chains using proper chaining fields based on the current checkpoint processing mode (either **R4 Mode** or **z2 Mode**. Therefore, users of these macros do not have to be sensitive to which JES2 processing mode is currently being used.

$#JOE is changed to process chains (such as the CHAR JOE chain). Both $QJQE and $#JOE have improved their loop control and have usability enhancements.

$DOGJQE
The JQA returned by $DOGJQE always reflects the **z2 Mode** processing format. Code that processes an artificial JQE (JQA) need not examine the JES2 processing mode (**z2 Mode** or **R4 Mode**) to examine JQA fields.

$JQEJNUM
$JQEJNUM is a new macro used to obtain the binary job number for a particular JQE. This macro operates in **z2 Mode** and **R4 Mode** and processes the JQA or JQE that is passed.

$JBIDBLD
This macro converts a binary job number to a printable jobid. $JBIDBLD is updated to accept a JQE (or JQA) as input and operates in **z2 Mode** or **R4 Mode**. $JBIDBLD formats the jobid, in the correct way, based on the current job number range.

Migration Actions

- Review your installation exit routines and thoroughly understand what changes you might need to make because of the various field, field names, and control block changes.
- Convert any ″unique″ installation code you use to use the new fields and field names.
- Process JOEs and JQEs as much as possible with the macros provided ($QJQE, $#JOE, $DOGJQE, $JQEJNUM, $JBIDBLD) instead of formulating your own unique code.
- Ensure that all the field and control block changes identified with JES2 z/OS V1R2 are incorporated, including updates to the surrounding logic.

$XMPOST Macro

- $XMPOST

  The $XMPOST macro is changed. When you specify $XMPOST XMPLIST=YES (or leave XMPLIST off and let it default), the mapping of the input parameter list is changed. The expected parameter list mapping is as follows:
  - +0 Error return address
  - +4 Address of ECB to be POSTed
  - +8 Address of related ASCB
  - +C ECB to be POSTed (if LOSTPOST=CHECK is specified on the macro call)

$QLOCNXT Macro

$QLOCNXT finds the next allocated job number after a specified job number.

Control Block Changes

$PIT

In z4, installations have the option of specifying zero JES2 initiators (i.e. use just WLM managed initiators). This is done by specifying INITDEF PARTNUM=0. The result is that $PITABLE and CCTPIT can have a value of zero since no PITs exist.

Migration Actions: Examine your exits for references to $PITABLE and CCTPIT and ensure that a value of zero is being handled.

$PSO

In z4, the $PSO has been moved to a data space

Migration Actions: Examine your exits for references to the PSO data area and update appropriately.

Executable Macro Updates

$#BUSY

In z4, the $#BUSY service was enhanced to perform $#POST when BUSY=OFF is specified and to perform $CKPT as appropriate.

Migration Actions: Examine all $#BUSY calls and remove unnecessary uses of $#POST or $CKPT. Failure to remove unnecessary $#POSTs will result in performance degradation.

Initialization

Checkpoint Data Sets

As of z/OS V1R1, JES2 no longer supports pre-release 4 checkpoints. JES2 can only read checkpoints that are in Release 4 mode. A Release 4 checkpoint is one that has been written by JES2 Release 4 or later after a $ACTIVATE command has been executed. To determine the checkpoint mode of your current checkpoint data set issue a $D ACTIVATE command. IF the checkpoint mode is not Release 4, you must activate to the Release 4 Mode before migrating to JES2 z/OS V1R2.

Migration Actions

- If your checkpoint is **R4 Mode** (You have used $ACTIVATE.), then no additional actions are needed. JES2 z/OS V1R2 can coexist with your existing systems or warm start using your existing checkpoint data set.
- If your checkpoint is **Pre-Release 4 Mode** (You have not used $ACTIVATE.), you must do so before you can add a JES2 z/OS V1R2 system to your MAS or before performing a warm start of a JES2 z/OS V1R2 system using your existing checkpoint data set.

SPOOLDEF FENCE= Changes

The FENCE= keyword parameter on the SPOOLDEF initialization statement is changed to allow you to specify fencing along with the number of spool volumes to be used.

When you specify

```
SPOOLDEF....FENCE=(ACTIVE=YES,VOLUMES=1).....
```

you get the same function as in the previous JES2 release when you specified FENCE=YES. . This specification of SPOOLDEF FENCE= is honored on $T SPOOLDEF and on cold start.

Migration Actions

SPOOLDEF FENCE=YES is still permitted for compatibility and is converted to ACTIVE=YES and VOLUMES=1.

SPOOLDEF TGBPERVL= Dropped

JES2 processing now controls the amount of SPOOL space cached for job writing data to SPOOL (the BLOB). In previous releases, this was controlled by the TGBPERVL parameter on SPOOLDEF. TGBPERVL= is no longer supported. In the initialization stream, this keyword is ignored (no error is returned). If used in a $TSPOOLDEF command, it will generate an error. In addition the HASP834 message that indicated TGBPERVL was set too high, has been deleted.

Migration Actions

Delete references to TGBPERVL from your initialization decks and any operator commands or automation scripts. Any automated action associated with the HASP834 message should be examined and potentially eliminated.

SPOOL Utilization

As part of automatically setting the size of the SPOOL cache (the BLOB), JES2 processing increased the amount of SPOOL space that traditionally was marked as allocated. In most installations, this change would not be noticeable. However, in certain smaller installations (such as test systems) the total SPOOL space that is

available may be allocated to the BLOB. This makes determining the true SPOOL utilization much more difficult. To compensate for this, JES2 now considers SPOOL space in the BLOB as available rather than as allocated (as it had in previous releases). In small installations, with limited SPOOL space, this could decrease total SPOOL utilization. Also, multiple MAS members at different JES2 levels may report different SPOOL utilizations.

Migration Actions

If you have a limited amount of SPOOL space, you may have set your SPOOLDEF TGSPACE=WARN= value high knowing that when that level was reached, you still have the space in the BLOB that can be used before SPOOL volumes are full. You should re-evaluate your settings of TGSPACE=WARN= to account for the changes in how SPOOL utilization has changes.

Also, be aware that SPOOL utilization may differ between MAS members of different release levels. This is normal. The smaller the number of track groups in the system, the greater the difference. However, the situation where the HASP355 message (SPOOL VOLUMES ARE FULL) is issued remains unchanged even in z/OS V1R1 JES2.

TPDEF SNABUF Changes

To improve SNA performance and better align with other networking products, JES2 has increased the SNA buffer size from a maximum of 3840 to 32512. This is a change to the TPDEF SNABUF=SIZE= keywords.

Migration Actions

Your VTAM network definitions should be evaluated.

TRACEDEF PAGES= Changes

The default for TRACEDEF PAGE= is changed from 2 to 9. For compatibility with other releases of JES2, specification of 1-8 are converted automatically to 9.

Migration Actions

For compatibility with other releases of JES2, specification of 1-8 for TRACEDEF PAGES, are converted automatically to 9. No migration actions are necessary, but you should evaluate your use of extended CSA space. If your current TRACEDEF initialization statements accepts the old default of PAGES=2, this will be converted to 9. For TABLES=100, you'll get 900 pages of buffers. Your setting for TABWARN= on the TRACEDEF initialization statements should also be evaluated.

ZAPJOB Initialization Statement

The ZAPJOB initialization statement is new and is used to remove all traces of a job structure from the JES2 job queue. This statement is intended for situations where a job cannot be removed using normal JES2 commands or by a JES2 restart.

Migration Actions

You should be careful when using this statement. Zapping a job that is active can lead to ABENDs and the loss of a PCE that might require a JES2 restart.

Migrating to z2 Mode

Several installation actions are required to migrate to **z2 Mode**:

- Stabilize all MAS members running JES2 z/OS V1R2 in **R4 Mode**
- `$ACTIVATE LEVEL=z2` without changing any limits

- Only when your installation is stable, running in **z2 Mode**, consider increasing limits.

Large Volume Support

JES2 supports relative and absolute track addressing for processing the SPOOL. An initialization option indicates whether relative addressing is used, never used, or only used when needed. This option ensures that relative addressing is not used before any vendor product that looks at SPOOL is ready to use relative addressing.

The SPOOLDEF initialization statement is updated to allow you to specify whether you "always", "as needed", or "never" want to use relative MTTRs in processing the JES2 SPOOL. You use the new keyword, RELADDR= with a value of ALWAYS , ASNEEDED, or NEVER to specify your choice.

Notes:

1. Relative addressing in processing the JES2 SPOOL is only used if all members of the MAS support relative addressing. Any member that is not supporting relative addressing fails to initialize and a HASP401 message results.
2. The size of a SPOOL data set is still limited to a size of 64K tracks.

The $T SPOOLDEF command is updated to allow you to update the current setting for RELADDR=. If relative addressing is being used for a volume, the $D SPOOL,UNITDATA displays a new keyword, BASETRAK=. The value of BASETRAK= must be added to the TT (part of MTTR) to convert a relative track address to an absolute track address.

A new interface is provided to read records from JES2 SPOOL. This interface is implemented as an option on the SSI 71 call. When you use this interface, you need not know about internal JES2 MTTR processing.

Migration Actions

- Understand how to use the SPOOLDEF initialization keyword, RELADDR= to specify the absolute or relative track addressing for the SPOOL.
- Understand how to use the new interface to read records from the JES2 SPOOL.
- Examine your code to ensure that the code the reads the JES2 SPOOL is consistent with this function.

Dynamic PROCLIBs Support

Customer IPL outages are aggravated by a defective JES2 PROC. JES2 z/OS V1R2 addresses this problem in "3" ways:

- Minimizes the information in the JES2 PROC (thus limiting the chance that some change in the system makes the PROC unusable).
- Provides the ability to start JES2 with a generic PROC, such as IEASYSAS for those cases where the JES2 PROC does become unusable. The two most frequently updated statements in the JES2 PROC are the PROCLIB data set concatenations and the JES2 initialization data sets. Now, JES2 provides initialization statements for these two data sets.
- Provides PROCLIB and INCLUDE initialization statements. The PROCLIB initialization statement allows you to define a logical DD name that the converter can use. Up to 255 data set names can be associated with each PROCLIB initialization statement. The INCLUDE initialization statement gives you the ability to "include" additional initialization streams into your initialization process.

Additionally, the following commands provide additional flexibility in specifying PROCLIB data sets:

- $ADD PROCLIB creates a new dynamic PROCLIB concatenation.
- $DEL PROCLIB deletes an existing dynamic PROCLIB. This command does not affect the static PROCLIBs defined in the JES2 PROC.
- $T PROCLIB updates an existing dynamic PROCLIB
- $D PROCLIB displays the current dynamic PROCLIBs only and does not display the static ones. This command has a DEBUG option that provides additional information about the dynamic PROCLIB concatenation as well as logically deleted dynamic PROCLIBs that are still in use.

Migration Actions

- Review your JES2 PROC.
- Review/Update your JES2 initialization statements appropriately.
- Take advantage of the PROCLIB and INCLUDE initialization statements to create dynamic PROCLIB concatenations and "include" initialization streams into your JES2 startup.

JES2 Support of a Maximum of 999,999 Job Numbers

With **z2 Mode**, you will be able to define up to 999,999 job numbers. Currently, the limit is just 65,534. In order to support the larger job numbers, the jobid format is changed. The format used depends on the upper limit on the RANGE operand of the JOBDEF initialization statement.

- If the upper limit is > 99,999, then the jobid is Jxxxxxxx
- If the upper limit is <= 99,999 then the jobid is JOBxxxxx
- If the upper limit is > 99,999, then the jobid is Txxxxxxx
- If the upper limit is <= 99,999 then the jobid is TSUxxxxx
- If the upper limit is > 99,999, then the jobid is Sxxxxxxx
- If the upper limit is <= 99,999 then the jobid is STCxxxxx

Note: If you are in **z2 Mode** and RASSIGN=YES is specified (or defaulted) on the JOBDEF initialization statement, the NJE jobs could use the Jxxxxxxx form of jobid if the job number is > 99,999; this occurs regardless of the RANGE setting.

Migration Actions

ATTENTION:

You should review the general description of activating your JES2 MAS to **z2 Mode** as well as the fields, field names, and control block changes that occur when running in **z2 Mode**. Reread "z2 Mode for $ACTIVATE" on page 17 and "General Field and Control Block Changes" on page 19 for the details.

- Review how your installation processes jobs/job ids. Ensure your automation adapts to potential jobids up to 999,999.
- Have ready back out procedures.
- Adjust your JOBDEF initialization parameters accordingly.
- Consider setting RASSIGN=NO.

Spinning JESlog Data Sets for Long Running Jobs

You initialize JES2 to allow the installation to "spin" or "suppress" JESlog data sets as the job runs by using the JESLOG= keyword on the JOBCLASS initialization statement.

Migration Actions

* Examine your usage of JESMSGLG and JESYSMSG data sets.
* Review initialization parameters (JOBCLASS) and JOB JCL.
* Makes changes you desire for "spinning" or "suppressing" JESlog message output.
* Review/Update automation to decide whether to use $T Jxxxx,SPIN.
* If you use Spool Data Set Browse, examine your use and be aware of new function.

Checkpoint Improvements

Prior to JES2 z/OS V1R2, switching between checkpoint modes required an all member warm start. Now installations can switch checkpoint modes using the $T CKPTDEF,MODE=DUAL|DUPLEX command. An all member warm start is not required. Changing to DUPLEX mode is always allowed. Changing to DUAL mode is NOT allowed when either CKPTn or NEWCKPTn is on the coupling facility or if DUPLEX=OFF is specified on the CKPTDEF initialization statement and both CKPTs are in use.

Note: You can use the $T CKPTDEF,MODE=DUAL|DUPLEX command even though some MAS members are not at the JES2 z/OS V1R2 release.

Migration Actions

* Update your procedures that change the MODE.

INCLUDE Statement

In z4, the INCLUDE statement has been enhanced to allow initialization statements to be read from the MVS logical PARMLIB concatenation. A second enhancement is that the DSN need not be supplied on an INCLUDE statement if the member being requested is in the same data set as the INCLUDEing member.

Migration Actions
Examine the initialization streams and update appropriately.

PARM operand on START command

In z4, JES2 start processing was enhanced to support a MEMBER= value specified via PARM= on the START command or within the PROC. This operand specifies a member of the MVS logical PARMLIB concatenation that contains the JES2 initialization stream.

If no source of initialization statements is specified either via MEMBER= nor via a successful OPEN of the HASPPARM DD data set, then the default is the HASxxxx (xxxx = the subsystem name, e.g. JES2) member in the MVS logical PARMLIB concatenation.

Migration Actions
If you have a member in the MVS logical PARMLIB with the default name mentioned above, take the appropriate action.

PARTNUM=0 on INITDEF statement is Now Supported

In z4, PARTNUM=0 is supported. This means that no JES2 initiators will be used. This in turn means that there is no PITs and $PITABLE and CCTPIT will be zero.

Migration Actions

If you decide to specify INITNUM=0, verify that no PITs is an acceptable condition.

New Monitor Address Space

In z4, JES2 starts a new (non-optional) address space for each JES2 instance called xxxxMON (xxxx = subsystem name e.g. JES2) to monitor the corresponding J3S2 address space.

Migration Actions

Be aware of the existence of a new system address space.

Security Considerations

New Profiles for monitor commands

In z4, JES2 monitor was provided. The monitor has a set of commands used to give information about the status of the JES2 address space. The entity name of these monitor commands in the CMDAUTH class is of the form: xxxxMON.action.object where xxxx is the subsystem name (e.g. JES2). The complete list of entity names for the JES2 monitor commands can be found in *z/OS JES2 Initialization and Tuning Guide*.

Migration Actions

New profiles may be required for RACF in order to allow monitor commands to be issued.

Operations

$ACTIVATE

Before warm starting JES2 z/OS V1R2, ensure a $ACTIVATE has been performed on JES2 OS/390 V2R8. To do this, issue a $D ACTIVATE command.

$SJ Command

In migrating from JES2 OS/390 V2R4 to JES2 z/OS V1R1, the behavior of the $SJ command changes if you have not installed the PTF to fix APAR OW38962.

- Prior to OW38962, the $SJ command only considered members of the JES2 MAS that were eligible to run a job, but only if (1) the member of the MAS did NOT have a member draining situation (such as, when a $P is entered) and (2) the member did NOT have an initiator draining situation (such as, when a $P XEQ is entered).

- After the fix to APAR OW38962 is installed, members of the MAS that are in a member draining situation are still NOT considered. JES2 ignores the initiator draining situation, but only on the MAS member where the $SJ command is entered.

To illustrate this change in processing, suppose there are three members active in the JES2 MAS: SY1, SY2, and SY3. Furthermore, suppose that a $P XEQ is issued on SY1 and SY2, but NOT on SY3. In this situation, when a $SJ command is issued

on SY3, SY3 is the only member that the job can run on, and then only if the affinity and SCHENV status permits it to run. If the $SJ command is issued on SY2, then both SY2 and SY3 are considered for running the job.

$T SPOOL(nnnn) Command

$T SPOOL(nnnn) is a new JES2 command. This commands allows the installation to assign its systems to spool volumes. This command has one valid keyword operand, SYSAFF=. The SYSAFF= keyword has a similar function SYSAFF on the $TINTRDR command. The value you specify for SYSAFF= on the $TSPOOL(nnnn) command is also added to the output produced by the $D SPOOL command.

Migration Actions
A new RACF authorization profile is required for the $T SPOOL command. The profile is JESx.MODIFY.SPOOL with ACCESS=CONTROL.

$ZAPJOB Command

$ZAPJOB removes all traces of a job structure from the JES2 job queue. You must ensure that the job and any output it may have created is not active in any JES2 process or active on a device. Zapping a job that is active can lead to ABENDs and the loss of a PCE until JES2 is restarted.

There is nothing to stop entering a $ZAPJOB command in the JES2 initialization stream, however, IBM recommends that the ZAPJOB initialization statement be used instead. A ZAPJOB command in the initialization stream will run after warm start processing completes whereas a ZAPJOB initialization statement will be processed before warm start processing begins, before verifying the job queue.

$HASP250 now indicates when a job has been zapped.

Migration Actions
A new RACF authorization profile is required for the $ZAPJOB command. The profile is JESx.ZAP.JOB with ACCESS=CONTROL.

$HASP250 has been changed to add that a job has been zapped.

$HASP834 Deleted

$HASP834 message has been deleted.

Migration Actions
Examine your installations message automation. There is no replacement for $HASP834.

Z2 Mode

Use the JOBDEF initialization statement to increase the amount of job numbers you can have in your installation. In **z2 Mode**, JES2 allows you to define up to 999,999 job numbers (expanding the limit from 65534).

With **z2 Mode**, you will be able to define up to 999,999 job numbers. Currently, the limit is just 65,534. In order to support the larger job numbers, the jobid format is changed. The format used depends on the upper limit on the RANGE operand of the JOBDEF initialization statement.

- If the upper limit is > 99,999, then the jobid is Jxxxxxxx
- If the upper limit is <= 99,999 then the jobid is JOBxxxxx
- If the upper limit is > 99,999, then the jobid is Txxxxxxx

- If the upper limit is <= 99,999 then the jobid is TSUxxxxx
- If the upper limit is > 99,999, then the jobid is Sxxxxxxx
- If the upper limit is <= 99,999 then the jobid is STCxxxxx

Note:

> JES2 uses the appropriate jobid format in displays of job information. There is a transition period after JOBDEF RANGE= is increased above 99,999 where the "old" format jobids may still be displayed. This can occur for jobs that were created when the old range was being used. Until these jobs are purged from the system, the old format for jobids may be displayed as part of informational messages about those specific jobs.

See *z/OS JES2 Initialization and Tuning Reference* for the specific changes to the JOBDEF initialization statement

Operator commands that process jobs by job number are improved to handle the increase of job numbers.

$T JOBDEF,RANGE= is updated to allow you to set a job number limit up to a maximum of 999,999. Also, the following $T command changes allow you to take advantage of new maximum for the number of jobs, number of JOEs and number of BERTs as well as to decrease their settings:

- $T JOBDEF JOBNUM= for the number of jobs
- $T OUTDEF JOENUM= for the number of JOEs
- $T CKPTSPACE BERTNUM= for the number of BERTs

JOBID Formats

The format of job numbers being displayed as part of command responses or messages has changed in z2 mode. Currently job numbers are displayed as JOBnnnnn. Five characters is not large enough to display job numbers greater than 99,999. Therefore, job numbers will be displayed differently when job numbers greater than 99,999 are used or potentially used.

The "new" job number format will be as follows: If the maximum allowed job number (high value of the JOBDEF RANGE= statement) is above 99,999 then the job number format will be J0nnnnnn. This format will always be used unless the job number range is decreased below 100,000. Similarly, STCnnnnn will become S0nnnnnn and TSUnnnnn will become T0nnnnnn.

If the job number range is increased above 99,999 or decreased below 100,000 with the $T command, a transition period will exist where the old job number format and the new job number format will be displayed as part of commands, messages, and SMF records.

Rules for determining job number format:

- If the local maximum is > 99,999 then use the new format.
- If the local maximum is <= 99,999 then use the old format as long as the job number being displayed is<= 99,999.

Note: If you are in **z2 Mode** and `RASSIGN=YES` is specified (or defaulted) on the JOBDEF initialization statement, the NJE jobs could use the Jxxxxxxx form of jobid if the job number is > 99,999; this occurs regardless of the `RANGE` setting.

Migration Actions

- Reread "z2 Mode for $ACTIVATE" on page 17 to understand how to use the $ACTIVATE command to bring your JES2 to **z2 Mode** and what are the changes to JES2 job number processing.
- Examine and change your automation with respect to commands, messages, and job numbers.
- Implement appropriate procedures to return to **R4 Mode**.

Spinning JESlog Data Sets for Long Running Jobs

Spinning or suppressing JESlog data described in "Spinning JESlog Data Sets for Long Running Jobs" on page 27. Refer to this section for an explanation of this function. A description of migration actions follow.

Migration Actions

- Examine your JOBCLASS(v) initialization statements and change as appropriate to use this function.
- Your installation workload of jobs might require the use of JESLOG.

Attention:

- Jobs that are converted on a JES2 z/OS V1R2 MAS member but run on a pre-JES2 z/OS V1R2 MAS member cannot perform the JESLOG spin off.
- Jobs that are converted on a JES2 z/OS V1R1 (or earlier) MAS member cannot perform JESLOG spin off regardless of which MAS member they run on.
- The converter (and START commands) cannot recognize the JESLOG in a JES2 z/OS V1R1 (or earlier) MAS member. This condition produces an error.

Examine how your installation responds to the following messages:
- HASP138
- HASP860
- HASP130

See *z/OS JES2 Messages* for details.

Dynamic JES2 PROCLIB Commands

JES2 z/OS V1R2 provides a PROCLIB initialization statement you use to define a logical DD name that the converter can use. Up to 255 data set names can be associated with the PROCLIB initialization statement. Dedicated PROCLIB data set concatenations can be eliminated.

Migration Actions

- Use the PROCLIB initialization statements in your initialization stream.
- Understand the following commands:.
  - $ADD PROCLIB creates a new dynamic PROCLIB concatenation.
  - $DEL PROCLIB deletes an existing dynamic PROCLIB. This command does not affect static PROCLIBs defined in the JES2 PROC.
  - $T PROCLIB updates an existing dynamic PROCLIB
  - $D PROCLIB displays the current dynamic PROCLIBs only and does not display the static ones. This command has a DEBUG option that provides additional information about the dynamic PROCLIB concatenation as well as logically deleted dynamic PROCLIBs that are still in use.
- This function can operate independent of mode (**z2 Mode** or **R4 Mode**).

See *z/OS JES2 Initialization and Tuning Reference* and *z/OS JES2 Commands* for more details.

Large Volume Support

JES2 supports both a relative and absolute track addressing for SPOOL. The SPOOLDEF initialization statement is updated to allow you to specify whether you **always**, **as needed**, or **never** want to use relative MTTRs in processing the JES2 SPOOL. You use the new keyword, RELADDR= with a value of ALWAYS , ASNEEDED, or NEVER to specify your choice.

Notes:

1. Relative addressing in processing the JES2 SPOOL is only used if all members of the MAS support relative addressing. Any member that is not supporting relative addressing fails to initialize and a HASP401 message results.

2. The size of a SPOOL data set is still limited to 64K tracks.

Migration Actions

- Examine the RELADDR= on the SPOOLDEF initialization statement to ensure it is appropriate for the volume. Make adjustments as necessary.
- Ensure that all members of the MAS use relative addressing.
- This support works in **z2 Mode** or **R4 Mode**.

Checkpoint Improvements

Installations can switch checkpoint modes using the $T CKPTDEF,MODE=DUAL|DUPLEX command. An all member warm start is not required. Changing to DUPLEX mode is always allowed. Changing to DUAL mode is NOT allowed when either CKPTn or NEWCKPTn is on the coupling facility or if DUPLEX=OFF is specified on the CKPTDEF initialization statement and both CKPTs are in use.

Note: You can use the $T CKPTDEF,MODE=DUAL|DUPLEX command even through some MAS members are not at the JES2 z/OS V1R2 level.

Migration Actions

- Update your procedures that change the MODE.

HASP443 Message

The HASP443 message clarifies the reason for a SPOOL allocation failure.

Migration Actions

- Examine any automation that processes this message.

Command Syntax Migration Aid Removal

In JES2 OS/390 V2R4, JES2 provided a command syntax compatibility and migration aid in the form of an automatically installed exit 5 routine. The migration aid was intended as a temporary bridge to give customers time to adapt to the command syntax changes made in OS/390 JES2 Version 2 Release 4. As of JES2 z/OS V1R2, the migration aid is no longer automatically installed and enabled. If you are not using the command syntax documented in *z/OS JES2 Commands* and are still relying on the migration aid, you can experience erroneous system operation or receive HASP649, HASP650, HASP003 messages.

The following table lists the commands translated and the results of that translation.

Table 19. Command Translation Formats

JES2 OS/390 V2R4 Format	JES2 z/OS V1R2 Format
$D'name',...	$DJOBQ'name',CMDAUTH=*,...
$T'name',...	$TJOBQ'name',... **(See NOTE 1)**
$DJ1,2,...	$DJ(1,2),... **(See NOTES 2,3)**
$DJ1-2,J3-4,...	$DJ(1-2,3-4),... **(See NOTE 3)**
$LJnnn,ALL	$DOJnnn
$LJnnn,H	$DOJnnn,HELD
$LJnnn,READY	$DOJnnn,READY
$LJnnn,OUTGRP=xxx	$DOJnnn,OUTGRP=xxx
$CJnnn,OUTGRP=xxx	$COJnn,OUTGRP=xxx
$PJnnn,OUTGRP=xxx	$POJnnn,OUTGRP=xxx **(See NOTE 4)**
$PJnnn,Q=x	$POJnnn,Q=x
$vJnnn,A=|Days=|Hours=	$vJnnn,A>|Days>|Hours>
$TJnnn,S=sid1,sid2,...	$TJnnn,S=(sid1,sid2,...)
$DSPL,JOBS=nn	$DJOBQ,SPOOL=(PERCENT>=nn)
$SSPL,V=xxxxxx,...	$SSPL(xxxxxx),...
$vIxx	$vI(xx)
$vIxx-yy	$vI(xx) **(See NOTE 5)**
$vJxxxx,Txxxx...	$vJOBQ(xxxx,xxxx) **(See NOTE 6)**
$TIxx,class-list	$TI(xx),C=class-list
$HQ,ALL	$TJOBCLASS(*),QHELD=Y
$HQ,C=xyz	$TJOBCLASS(x,y,z),QHEL=Y
$AQ,ALL	$TJOBLCASS(*),QHELD=N
$AQ,C=xyz	$TJOBCLASS(x,y,z),QHELD=N
$PQ,ALL,...	$POJOBQ,READY,...
$PQ,Q=xyz	$POJOBQ,READY,Q=xyz
$OQ,ALL,...	$OJOBQ,/R=LOCAL.*,...
$OQ,Q=xyz,...	$OJOBQ,/R=LOCAL.*,/Q=xyz,...
$TALL,sid1,sid2,...	$TJOBQ(*),/S=(sid1,sid2,...)
$LSYS	$DMEMBER
$ESYS,sid	$EMEMBER(sid)
$ESYS,RESET=sid	$ECKPTLOCK,HELDBY=sid
$TSYS,IND=Y|N	$TMEMBER(local),IND=Y|N

Table 19. Command Translation Formats (continued)

JES2 OS/390 V2R4 Format	JES2 z/OS V1R2 Format
Notes:	
1. Changes are similar for $A, $C, $E, $H, $L, $O, $P, $T, $TO	
2. J can be J, JOB, S, STC, T, TSU	
3. Changes are similar for $A, $C, $E, $H, $L, $O, $P, $TO	
4. Unless Q= is a valid job queue (XEQ, PPU, etc.)	
5. Where v is one of the command verbs D, P, S, T, or Z. This command processed a named range of initiators. You must now issue these commands as separate commands.	
6. Where v is one of the commands verbs A, C, E, H, L, O, P, T, TO. This command processed a list of JOBs, TSO users, and STCs. You must now issue these commands either as multiple commands ($v Jxxxx... and $v Txxxx...) or use $v JOBQ(xxxx,xxxx). The JOBQ version has slightly different function (does not check job type).	

Migration Actions

If you are exclusively using the command syntax documented in the *z/OS JES2 Commands* publication and are not relying on the command migration aid exit, then no action is required. If you do not know whether or not you are currently relying on the command migration aid exit, you need to do the following:

1. First determine if the migration exit is installed. Issue $DEXIT5 on your current production systems and if the command response does not include routine name HASX5CTR, then no further action is required.

2. If the exit is installed, look through SYSLOG for $HASP006 messages which display translated commands. The fact that commands are translated does not necessarily indicate dependence on the migration aid exit because it conservatively translates several command formats that already have valid syntax. Commands with verbs with a single subscript or subscript range that are enclosed in parentheses are frequently unnecessarily translated as shown in the following examples:

   ```
   $HASP006 $DJ5   TRANSLATED BY EXIT HASX05C TO $DJ(5)
   $HASP006 $DJ5-6 TRANSLATED BY EXIT HASX05C TO $DJ(5-6)
   ```

   If however, you find a $HASP006 message in SYSLOG with any other form, or in the above form but having additional operands that are altered, then you are currently dependent on the migration aid exit. If you find you are currently dependent on the migration aid exit, then determine the origin of the obsolete command syntax and take corrective action. Either:

   - Have your operators begin using only the supported command syntax and convert your automation scripts to use valid syntax. Refer to the *z/OS JES2 Commands* for correct syntax of JES2 commands. IBM recommends you begin this task immediately on your current production systems and analyze $HASP006 messages in SYSLOG to measure your progress.

   OR

   - Although IBM strongly recommends against it, install the command migration aid exit yourself and assume the risks described in the **CAUTION** below. As of HJE7705, the command migration aid exit is now shipped as member HASX05C in SYS1.SHASSAMP.

> **CAUTION !**
> The JES2 command migration aid, exit 5 routine HASX5CTR, is now shipped in member HASX05C of SYS1.SHASSAMP and is no longer supported. This exit is not warranted and is functionally stabilized. You are strongly urged to start eliminating your dependency on this exit now and stop using the exit as soon as possible. Failure to make an orderly transition now could result in a prematurely rushed transition later.

TSO GR, TSO SEND Commands and TSO Logons

In z4, JES2 discontinued policing TSO logons. Duplicate instances of a given userid can be logged on at the same time. It is highly recommended that the GRS RNL list be updated to convert SYSIKJUA to a SYSTEMS ENQ in order to prevent duplicates. If the RNL is not updated, and duplicate instances of a userid are logged on, then TSO GR and SEND messages to the duplicately logged on userids will not function.

Migration Actions
Ensure that the procedure outlined in the *z/OS TSO/E Customization* is followed in order to block duplicate logons.

PARM operand on START command

In z4, JES2 start processing was enhanced to support a MEMBER= value specified via PARM= on the START command or within the PROC. This operand specifies a member of the MVS logical PARMLIB concatenation that contains the JES2 initialization stream.

If no source of initialization statements is specified either via MEMBER= nor via a successful OPEN of the HASPPARM DD data set, then the default is the HASxxxx (xxxx = the subsystem name, e.g. JES2) member in the MVS logical PARMLIB concatenation.

Migration Actions
If the systems programmer elects to use the MVS logical PARMLIB concatenation for JES2 parameter statements, you will need to update operations procedures. Local procedures for recovering from errors such as "PARMLIB not found" or JES2 PROC is bad can be updated this function to simplify recovery.

Monitor Commands

In z4, a new monitor has been created to monitor JES2 functions. This monitor has several commands, one in particular used to display information about JES2 when JES2 is non-responsive.

Migration Actions
Update local procedures and educate operators on the use of the $J D STATUS command when JES2 appears to be malfunctioning.

Monitor Alerts

In z4, the JES2 monitor can highlighted messages to report on abnormal conditions in the JES2 address space. These highlighted messages are called "alerts".

Migration Actions

Educate operators on the new alerts and review the documentation for HASP92xx messages and corresponding actions.

JES2 Checkpoint Corruption Message

In z4, JES2 warmstart processing tracks the number of errors encountered during initialization. If the number of errors exceeds a threshold, a HASP863/HASP272 message pair is issued to query whether initialization should be allowed to complete. This query is made before committing changes to the JES2 checkpoint.

Migration Actions

Update local procedures to describe what should be done if this message occurs. The procedure should state that initialization should not continue unless the operator is certain that the correct checkpoint and SPOOL volumes are in use.

JES2 Use of 4 digit message numbers

Starting in z4, JES2 uses message ids with 4 digit numbers (HASPnnnn).

Migration Actions

Ensure that automation takes into account that JES2 message ids can now have 4 digits.

Application Development

APPLCOPY No Longer Supported

APPLCOPY is no longer supported.

APPLCOPY is a way to access data in the checkpoint, but is replaced with checkpoint versions. (SSI 71) applications that used APPLCOPY (such as SDSF) have migrated to now use checkpoint versions. Becuase of installation overhead concerns, especially with very large checkpoint data sets, support for APPLCOPY has been discontinued. The CKPTDEF APPLCOPY= keyword is dropped and using it will generate an error during initialization (causing other operands on CKPTDEF to be ignored).

Migration Actions

- Installations should delete references to APPLCOPY in their initialization decks.
- Installations that have an application that still uses APPLCOPY, need to have that application migrate to the use of checkpoint versions before installing JES2 z/OS V1R2.

// XMIT JCL Statement

In z4, JES2 provides support for the // XMIT JCL statement to route jobs for execution at other nodes. Older methods of routing jobs are still supported.

Migration Actions

JCL can be updated that will allow the same stream to be used in JES2 as can be used in JES3. You may need to update applications that scan JCL to account for the presence and function of the new JCL statement.

Original Record Length passed via FSI

In prior releases, original line length was not passed to FSS applications via the FSI. This causes a problem when blank truncation was used for a SYSOUT class

and the data being processed was not EBCDIC (binary streams). Beginning in z4, the original record length is made available for each record passed via the FSI. The length is returned in the IDX (mapping macro IAZIDX).

Migration Actions
Any FSS application which can process binary data streams should be updated to take advantage of the new data.

SAPI JMR

In z4, JES2 has been updated to pass the JMR field JMRUSEID to the application via the SSS2USID field of the IAZSSS2.

Migration Actions
SAPI applications can use the field when generating SMF records.

SPOOL Read SSI

In z4, JES2 has been enhanced to provide an interface to the SPOOL SSI that will allow the return of in storage buffers.

Migration Actions
Use the facility if needed.

Problem Determination and Diagnosis

Provide for JES2 IPCS Support

> **Attention!**
> - You must be proactive to install JES2 IPCS.
> - Make sure JES2 IPCS support works before you experience an emergency.
> - Make sure you have JES2 IPCS support for all combinations of JES2 and MVS releases in production. (*z/OS JES2 Migration* and *z/OS JES2 Diagnosis* provide information about JES2 IPCS support.)

- When JES2 dumps are being diagnosed on the same level of the system as the system on which the dumps were taken, do the following:
  - Ensure that SHASPARM is specified in the PARMLIB concatenation.
  - Ensure that SHASMIG is specified in the STEPLIB concatenation.
  - Ensure that SHASPNL0 is specified in the ISPPLIB concatenation.

  This ensures that IPCS can find the SMP/E-maintained copy of JES2 data.
- If the JES2 levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, but the level of MVS is the same, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also specify your PARMLIB data set, plus any other data sets containing IPCS parameters that you normally need when using IPCS.
  - Specify the SHASMIG library that corresponds to the JES2 level in the dump in the first position in the STEPLIB concatenation.
  - Specify the SHASPNL0 library that corresponds to the JES2 level in the dump in the first position in the ISPPLIB concatenation.

- If both the JES2 and system levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also, specify the PARMLIB data set, plus any other data sets containing IPCS parameters that correspond to the system level in the dump.
  - Specify the SHASMIG and MIGLIB libraries that correspond to the system in the dump in the first and second positions in the STEPLIB concatenation.
  - Specify the SHASPNL0 and SBLSPNL0 libraries that correspond to the system in the dump in the first position in the ISPPLIB concatenation.

Note: IPCS requires libraries in addition to the libraries that JES2 uses. You may need to make other concatenations. For more information, see *z/OS MVS IPCS Customization*.

If dump analysis leads into data structures for other elements or products, you might need to add more data sets to the concatenations to complete the diagnosis. Refer to the appropriate product publications for information about establishing the correct IPCS environments for diagnosing other elements and products.

Dumping all Members of the MAS

z/OS V1R1 JES2 dumps all members of the MAS when certain JES2 work queue errors occur. This is referred to as "multi-system dumps".

Migration Actions
You need to make sure you have the procedures in place to collect all the dumps from all the systems.

HASP443 Message

The HASP443 message is updated to better clarify the reason a SPOOL allocation failed. One or more "reason code" can be displayed with this message to further clarify this conditon.

Migration Actions
- Review how you examine the HASP443 message.
- If you have automation that interprets HASP443, then you might need to make changes.

Auto Reply to Termination WTORs

Automatic responses to HASP070, HASP089, and HASP098 occur if there is no operator response for 10 minutes. See *z/OS JES2 Messages* for message, HASP065 for more detail.

Changes in System Trace Contents

In z4, many uses of SVC 111 have been converted to stacking PC instructions. In addition JES2 uses SVC 92 instead of SVC 0 to write data to spool for executing jobs.

Migration Actions
Be aware of these changes when debugging problems via the system trace.

Provide for JES2 IPCS Support

Attention:

- You must be proactive to install JES2 IPCS.
- Make sure JES2 IPCS support works before you experience an emergency.
- Make sure you have JES2 IPCS support for all combinations of JES2 and MVS releases in production. (*z/OS JES2 Migration* and *z/OS JES2 Diagnosis*provide information about JES2 IPCS support.)
- When JES2 dumps are being diagnosed on the same level of the system as the system on which the dumps were taken, do the following:
  - Ensure that SHASPARM is specified in the PARMLIB concatenation.
  - Ensure that SHASMIG is specified in the STEPLIB concatenation.
  - Ensure that SHASPNL0 is specified in the ISPPLIB concatenation.

  This ensures that IPCS can find the SMP/E-maintained copy of JES2 data.
- If the JES2 levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, but the level of MVS is the same, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also specify your PARMLIB data set, plus any other data sets containing IPCS parameters that you normally need when using IPCS.
  - Specify the SHASMIG library that corresponds to the JES2 level in the dump in the first position in the STEPLIB concatenation.
  - Specify the SHASPNL0 library that corresponds to the JES2 level in the dump in the first position in the ISPPLIB concatenation.
- If both the JES2 and system levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also, specify the PARMLIB data set, plus any other data sets containing IPCS parameters that correspond to the system level in the dump.
  - Specify the SHASMIG and MIGLIB libraries that correspond to the system in the dump in the first and second positions in the STEPLIB concatenation.
  - Specify the SHASPNL0 and SBLSPNL0 libraries that correspond to the system in the dump in the first position in the ISPPLIB concatenation.

Notes:

1. IPCS requires libraries in addition to the libraries that JES2 | uses. You may need to make other concatenations. For more information see *z/OS TSO/E Customization*.
2. If dump analysis leads into data structures for other elements or products, you might need to add more data sets to the concatenations to complete the diagnosis. Refer to the appropriate product publications for information about establishing the correct IPCS environments for diagnosing other elements and products.

Support for JES2 Trace IDs 32 and 33

Trace IDs 32 and 33 have been added in z4. Trace id 32 is created whenever a JOE is removed ($#REM). The trace has the work JOE and the exit number in control.

Trace ID 33 is created whenever an NJE header or trailer or NMR is transmitted or received over NJE or SPOOL offload. TRACE=YES has to be specified on the line or node for which trace is desired. The data included in the trace is the header/trailer/NMR in an uncompressed non compacted format.

Migration Actions

Be aware of the additional diagnostic information provided by the new trace IDs.

JES2 Installation Modifications

Active DCTs

Active DCTs are now chained out of the HCT ($NJEADCT, $OFFADCT, $LCLADCT).

Note: If an error occurs in the DCT chain, a SYSOUT device may not be selected for work. Such might be the case when $S command is issued. JES2 rebuilds this chain should errors occur.

Migration Actions

Installations that manipulate DCTs need to be aware of these changes.

Modification Considerations

If you plan modifications to JES2 beyond using exit routines, your migration actions will be similar to those for exit routines. See "z/OS JES2 Version 1 Release 2 Overview" on page 17 and review each item in JES2 z/OS V1R2. The migration actions enumerated for these items will apply to your own special modifications.

Distribution Library Name Changes

Previous library data set names included version, modification, and release values (for example,SYS1.V2R10M0.SHASLINK. In JES2 z/OS V1R2, the *"vrm"* qualifier (which in the example is V2R10M0) is eliminated (For example, SYS1.SHASLINK is the proper name.).

Migration Actions

Change your installations JCL appropriately, including TSO and LOGON PROCs for IPCS.

Counting Null Records

Prior to JES2 z/OS V1R2, null records written to SYSOUT data sets were padded to a single blank record. Because of this padding, these records were counted as records in the job totals. This count is not consistent with SYSOUT received over NJE. Null records from NJE are not padded and not counted in the job totals.

Migration Actions

Since the processing to pad null SYSOUT records was deleted in JES2 z/OS V1R2, counting may change record counts for existing jobs. You would notice this change in a mixed MAS (JES2 z/OS V1R2 running with earlier levels of JES2).

Chapter 7. Migration Actions: JES2 OS/390 V2R7 to JES2 z/OS V1R4

Customization

New and Changed Installation Exits

Installation Exits 11 and 12

Installation exits 11 and 12 are unaffected if you use no new functions in z/OS V1R1 JES2. However, if you turn on spool affinity or set fencing to more than one volume, then your current exits 11 and 12 are affected.

Migration Actions: Review the documentation for exits 11 and 12 in *z/OS JES2 Installation Exits*. Before activating SPOOL affinity, ensure that the correct available SPOOL masks are used to determine what volumes to add to a job in these exits.

Your installation uses spool partitioning if `FENCE=ACTIVE=YES` is specified on the SPOOLDEF initialization statement. Spool affinity is used if any spool volume has something other than SYSAFF=ANY displayed for it.

Installation Exits 0, 19, 24

The HASPINIT load module and the initialization PCE have been moved above the 16M line. This may impact installation exits called during initialization processing (such as exits 0, 19 and 24). If your exit switches to AMODE 24 to invoke DFP modules, then this code will no longer be able to access the initialization PCE (and in particular the save area in the PCE).

Migration Actions: Review all exits for code that switches to AMODE 24 ($AMODE 24 macro invocations). Verify that the switch to 24 bit mode is still required. If it is required and you are in an exit that gets control during initialization, ensure that the code running in AMODE 24 does not access the initialization PCE.

Also, examine all uses of DFP macros, such as GET and PUT. If these macros were hand coded (instead of using the IBM-supplied macro) changes might be necessary. The DFP macros were changed to support 31 bit addressable callers.

Exit 5

In z/OS V1R2 (HJE7705), the command syntax migration aid exit 5 routine, HASX5CTR in module HASX05C, is no longer automatically installed and enabled. As of HJE7705, the exit is now shipped as a sample in SYS1.SHASSAMP and is no longer supported. The function of this module was intended only as a migration aid to translate pre-OS/390 JES2 Version 2 Release 4 command syntax to OS/390 JES2 Version 2 Release 4 command syntax.

Migration Actions: IBM recommends you use the JES2 command syntax as documented in *z/OS JES2 Commands*. If you still need command translation, you need to install and enable EXIT 5 (HASX05C).

Note: The function of HASX05C is not altered from JES2 OS/390 V2R4. See Command Syntax Migration Aid Removal in the "Operations" section of this chapter for more information.

Exit 14 and Exit 49

PITCLASS and QGTCLST are changed.

In z/OS V1R2, QGTCLST now points to a work area, which contains the class list for this selection; this class list is a subset of the classes listed in PITCLASS. PITCLASS contains the class list for the initiator as defined by the INIT(nnnn) initialization statement; QGTCLST contains the class list for this selection and can contain different values than PITCLASS.

Migration Actions: If exit 14 and exit 49 in your installation reference the fields PITCLASS or QGTCLST, you need to examine these references and make appropriate changes.

Field, Field Name, and Control Block Changes

For JES2 to use up to 999,999 job numbers requires that you activate your JES2 z/OS V1R2 MAS to **z2 Mode** . Unless this activation is performed, JES2 operates in **R4 Mode**, and JES2 z/OS V1R2 behaves with the job number limitations to 65534.

Migration Actions

> **Attention !**
>
> In migrating to JES2 z/OS V1R2, it is important to refer to "General Field and Control Block Changes" on page 19 for the needed details about all specific fields, field names, and control block changes that take place both when migrating to JES2 z/OS V1R2 and when you activate.

In preparing for your migration to JES2 z/OS V1R2 you need to analyze:

*   All your exit routines and "unique" code that would be affected by activating JES2 z/OS V1R2 in **z2 Mode**
*   How your JES2 system processess the JOE, JQE, JQA, and JOT control blocks that are significantly changed in **z2 Mode**.
*   Control blocks and fields used to store job numbers (such as the $SJB,$ JIB, $FSAXB, and $COMWORK).
*   How to use JES2 macro services, such as $QJQE, $#JOE, $DOGJQE, $JBIDBLD.
*   Your assembler errors that result from not thoroughly finding all places where fields, field names, and control block changes affect JES2.
*   Your JES2 exits and how they are affected, making necessary changes, re-assembling, and integrating them.
*   If your exits use JQE or JOE chaining fields or reference job number fields, then **z2 Mode** changes require updates to your exit routines. Field names have been changed to help identify where logic changes are needed.

Macro Updates Relating to JOE and JQE Changes

Some macros are created and updated to allow JES2 exit routines to be mode insensitive.

$QJQE and $#JOE

These macros process the JQE and JOE control block chains using proper chaining fields based on the current checkpoint processing mode (either **R4 Mode** or **z2 Mode**. Therefore, users of these macros do not have to be sensitive to which JES2 processing mode is currently being used.

$#JOE is changed to process chains (such as the CHAR JOE chain). Both $QJQE and $#JOE have improved their loop control and have usability enhancements.

$DOGJQE

The JQA returned by $DOGJQE always reflects the **z2 Mode** processing format. Code that processes an artificial JQE (JQA) need not examine the JES2 processing mode (**z2 Mode** or **R4 Mode**) to examine JQA fields.

$JQEJNUM

$JQEJNUM is a new macro used to obtain the binary job number for a particular JQE. This macro operates in **z2 Mode** and **R4 Mode** and processes the JQA or JQE that is passed.

$JBIDBLD

This macro converts a binary job number to a printable jobid. $JBIDBLD is updated to accept a JQE (or JQA) as input and operates in **z2 Mode** or **R4 Mode**. $JBIDBLD formats the jobid, in the correct way, based on the current job number range.

Migration Actions

- Review your installation exit routines and thoroughly understand what changes you might need to make because of the various field, field names, and control block changes.
- Convert any "unique" installation code you use to use the new fields and field names.
- Process JOEs and JQEs as much as possible with the macros provided ($QJQE, $#JOE, $DOGJQE, $JQEJNUM, $JBIDBLD) instead of formulating your own unique code.
- Ensure that all the field and control block changes identified with JES2 z/OS V1R2 are incorporated, including updates to the surrounding logic.

$XMPOST Macro

- $XMPOST

The $XMPOST macro is changed. When you specify $XMPOST XMPLIST=YES (or leave XMPLIST off and let it default), the mapping of the input parameter list is changed. The expected parameter list mapping is as follows:

- +0 Error return address
- +4 Address of ECB to be POSTed
- +8 Address of related ASCB
- +C ECB to be POSTed (if LOSTPOST=CHECK is specified on the macro call)

$QLOCNXT Macro

$QLOCNXT finds the next allocated job number after a specified job number.

Initialization

PTF for APAR OW36796

If you are migrating from OS/390 JES2 Release 3, 4, 5, or 7, without PTF OW36796 applied, then ensure that you have not tried to change NJEDEF OWNNODE on a warmstart and replied "Y" to message HASP441 to continue initialization. If you have, then contact IBM service prior to installing z/OS V1R1 JES2.

To determine if you have attempted to change OWNNODE on a warm start, examine the syslogs from the most recent restart of JES2 for each member of your MAS. Look for the following sequence of messages:

```
HASP442  INITIALIZATION STATEMENTS CONFLICTING WITH
               SAVED VALUES

HASP496  NJEDEF OWNNODE=xx CANNOT BE CHANGED FROM SAVED
               VALUE OF yy

HASP441  REPLY 'Y' TO TO CONTINUE INITIALIZATION OR
               'N' TO TERMINATE IN RESPONSE TOOOO HASP442
```

You should contact IBM service if you find the above messages in your log and a reply of "Y" was given to message HASP441. If you do not find these messages with a reply of "Y", then no migration action is required.

Checkpoint Data Sets

As of z/OS V1R1, JES2 no longer supports pre-release 4 checkpoints. JES2 can only read checkpoints that are in Release 4 mode. A Release 4 checkpoint is one that has been written by JES2 Release 4 or later after a $ACTIVATE command has been executed. To determine the checkpoint mode of your current checkpoint data set issue a $D ACTIVATE command. IF the checkpoint mode is not Release 4, you must activate to the Release 4 Mode <u>before</u> migrating to JES2 z/OS V1R2.

Migration Actions

- If your checkpoint is **R4 Mode** (You have used $ACTIVATE.), then no additional actions are needed. JES2 z/OS V1R2 can coexist with your existing systems or warm start using your existing checkpoint data set.
- If your checkpoint is **Pre-Release 4 Mode** (You have not used $ACTIVATE.), you must do so before you can add a JES2 z/OS V1R2 system to your MAS or before performing a warm start of a JES2 z/OS V1R2 system using your existing checkpoint data set.

SPOOLDEF FENCE= Changes

The FENCE= keyword parameter on the SPOOLDEF initialization statement is changed to allow you to specify fencing along with the number of spool volumes to be used.

When you specify

```
SPOOLDEF....FENCE=(ACTIVE=YES,VOLUMES=1).....
```

you get the same function as in the previous JES2 release when you specified FENCE=YES. . This specification of SPOOLDEF FENCE= is honored on $T SPOOLDEF and on cold start.

Migration Actions

SPOOLDEF FENCE=YES is still permitted for compatibility and is converted to ACTIVE=YES and VOLUMES=1.

SPOOLDEF TGBPERVL= Dropped

JES2 processing now controls the amount of SPOOL space cached for job writing data to SPOOL (the BLOB). In previous releases, this was controlled by the TGBPERVL parameter on SPOOLDEF. TGBPERVL= is no longer supported. In the initialization stream, this keyword is ignored (no error is returned). If used in a

$TSPOOLDEF command, it will generate an error. In addition the HASP834 message that indicated TGBPERVL was set too high, has been deleted.

Migration Actions
Delete references to TGBPERVL from your initialization decks and any operator commands or automation scripts. Any automated action associated with the HASP834 message should be examined and potentially eliminated.

SPOOL Utilization

As part of automatically setting the size of the SPOOL cache (the BLOB), JES2 processing increased the amount of SPOOL space that traditionally was marked as allocated. In most installations, this change would not be noticeable. However, in certain smaller installations (such as test systems) the total SPOOL space that is available may be allocated to the BLOB. This makes determining the true SPOOL utilization much more difficult. To compensate for this, JES2 now considers SPOOL space in the BLOB as available rather than as allocated (as it had in previous releases). In small installations, with limited SPOOL space, this could decrease total SPOOL utilization. Also, multiple MAS members at different JES2 levels may report different SPOOL utilizations.

Migration Actions
If you have a limited amount of SPOOL space, you may have set your SPOOLDEF TGSPACE=WARN= value high knowing that when that level was reached, you still have the space in the BLOB that can be used before SPOOL volumes are full. You should re-evaluate your settings of TGSPACE=WARN= to account for the changes in how SPOOL utilization has changes.

Also, be aware that SPOOL utilization may differ between MAS members of different release levels. This is normal. The smaller the number of track groups in the system, the greater the difference. However, the situation where the HASP355 message (SPOOL VOLUMES ARE FULL) is issued remains unchanged even in z/OS V1R1 JES2.

TPDEF SNABUF Changes

To improve SNA performance and better align with other networking products, JES2 has increased the SNA buffer size from a maximum of 3840 to 32512. This is a change to the TPDEF SNABUF=SIZE= keywords.

Migration Actions
Your VTAM network definitions should be evaluated.

TRACEDEF PAGES= Changes

The default for TRACEDEF PAGE= is changed from 2 to 9. For compatibility with other releases of JES2, specification of 1-8 are converted automatically to 9.

Migration Actions
For compatibility with other releases of JES2, specification of 1-8 for TRACEDEF PAGES, are converted automatically to 9. No migration actions are necessary, but you should evaluate your use of extended CSA space. If your current TRACEDEF initialization statements accepts the old default of PAGES=2, this will be converted to 9. For TABLES=100, you'll get 900 pages of buffers. Your setting for TABWARN= on the TRACEDEF initialization statements should also be evaluated.

ZAPJOB Initialization Statement

The ZAPJOB initialization statement is new and is used to remove all traces of a job structure from the JES2 job queue. This statement is intended for situations where a job cannot be removed using normal JES2 commands or by a JES2 restart.

Migration Actions

You should be careful when using this statement. Zapping a job that is active can lead to ABENDs and the loss of a PCE that might require a JES2 restart.

Migrating to z2 Mode

Several installation actions are required to migrate to **z2 Mode**:

- Stabilize all MAS members running JES2 z/OS V1R2 in **R4 Mode**
- `$ACTIVATE LEVEL=z2` without changing any limits
- Only when your installation is stable, running in **z2 Mode**, consider increasing limits.

Large Volume Support

JES2 supports relative and absolute track addressing for processing the SPOOL. An initialization option indicates whether relative addressing is used, never used, or only used when needed. This option ensures that relative addressing is not used before any vendor product that looks at SPOOL is ready to use relative addressing.

The SPOOLDEF initialization statement is updated to allow you to specify whether you "always", "as needed", or "never" want to use relative MTTRs in processing the JES2 SPOOL. You use the new keyword, `RELADDR=` with a value of `ALWAYS` , `ASNEEDED`, or `NEVER` to specify your choice.

Notes:

1. Relative addressing in processing the JES2 SPOOL is only used if all members of the MAS support relative addressing. Any member that is not supporting relative addressing fails to initialize and a HASP401 message results.
2. The size of a SPOOL data set is still limited to a size of 64K tracks.

The $T SPOOLDEF command is updated to allow you to update the current setting for `RELADDR=`. If relative addressing is being used for a volume, the $D SPOOL,UNITDATA displays a new keyword, BASETRAK=. The value of BASETRAK= must be added to the TT (part of MTTR) to convert a relative track address to an absolute track address.

A new interface is provided to read records from JES2 SPOOL. This interface is implemented as an option on the SSI 71 call. When you use this interface, you need not know about internal JES2 MTTR processing.

Migration Actions

- Understand how to use the SPOOLDEF initialization keyword, `RELADDR=` to specify the absolute or relative track addressing for the SPOOL.
- Understand how to use the new interface to read records from the JES2 SPOOL.
- Examine your code to ensure that the code the reads the JES2 SPOOL is consistent with this function.

Dynamic PROCLIBs Support

Customer IPL outages are aggravated by a defective JES2 PROC. JES2 z/OS V1R2 addresses this problem in "3" ways:

- Minimizes the information in the JES2 PROC (thus limiting the chance that some change in the system makes the PROC unusable).
- Provides the ability to start JES2 with a generic PROC, such as IEASYSAS for those cases where the JES2 PROC does become unusable. The two most frequently updated statements in the JES2 PROC are the PROCLIB data set concatenations and the JES2 initialization data sets. Now, JES2 provides initialization statements for these two data sets.
- Provides PROCLIB and INCLUDE initialization statements. The PROCLIB initialization statement allows you to define a logical DD name that the converter can use. Up to 255 data set names can be associated with each PROCLIB initialization statement. The INCLUDE initialization statement gives you the ability to "include" additional initialization streams into your initialization process.

Additionally, the following commands provide additional flexibility in specifying PROCLIB data sets:
- $ADD PROCLIB creates a new dynamic PROCLIB concatenation.
- $DEL PROCLIB deletes an existing dynamic PROCLIB. This command does not affect the static PROCLIBs defined in the JES2 PROC.
- $T PROCLIB updates an existing dynamic PROCLIB
- $D PROCLIB displays the current dynamic PROCLIBs only and does not display the static ones. This command has a DEBUG option that provides additional information about the dynamic PROCLIB concatenation as well as logically deleted dynamic PROCLIBs that are still in use.

Migration Actions
- Review your JES2 PROC.
- Review/Update your JES2 initialization statements appropriately.
- Take advantage of the PROCLIB and INCLUDE initialization statements to create dynamic PROCLIB concatenations and "include" initialization streams into your JES2 startup.

JES2 Support of a Maximum of 999,999 Job Numbers

With **z2 Mode**, you will be able to define up to 999,999 job numbers. Currently, the limit is just 65,534. In order to support the larger job numbers, the jobid format is changed. The format used depends on the upper limit on the RANGE operand of the JOBDEF initialization statement.
- If the upper limit is > 99,999, then the jobid is Jxxxxxxx
- If the upper limit is <= 99,999 then the jobid is JOBxxxxx
- If the upper limit is > 99,999, then the jobid is Txxxxxxx
- If the upper limit is <= 99,999 then the jobid is TSUxxxxx
- If the upper limit is > 99,999, then the jobid is Sxxxxxxx
- If the upper limit is <= 99,999 then the jobid is STCxxxxx

Note: If you are in **z2 Mode** and RASSIGN=YES is specified (or defaulted) on the JOBDEF initialization statement, the NJE jobs could use the Jxxxxxxx form of jobid if the job number is > 99,999; this occurs regardless of the RANGE setting.

Migration Actions

ATTENTION:

JES2 OS/390 V2R7 to JES2 z/OS V1R4

You should review the general description of activating your JES2 MAS to **z2 Mode** as well as the fields, field names, and control block changes that occur when running in **z2 Mode**. Reread "z2 Mode for $ACTIVATE" on page 17 and "General Field and Control Block Changes" on page 19 for the details.

- Review how your installation processes jobs/job ids. Ensure your automation adapts to potential jobids up to 999,999.
- Have ready back out procedures.
- Adjust your JOBDEF initialization parameters accordingly.
- Consider setting RASSIGN=NO.

Spinning JESlog Data Sets for Long Running Jobs

You initialize JES2 to allow the installation to "spin" or "suppress" JESlog data sets as the job runs by using the JESLOG= keyword on the JOBCLASS initialization statement.

Migration Actions
- Examine your usage of JESMSGLG and JESYSMSG data sets.
- Review initialization parameters (JOBCLASS) and JOB JCL.
- Makes changes you desire for "spinning" or "suppressing" JESlog message output.
- Review/Update automation to decide whether to use $T Jxxxx,SPIN.
- If you use Spool Data Set Browse, examine your use and be aware of new function.

Checkpoint Improvements

Prior to JES2 z/OS V1R2, switching between checkpoint modes required an all member warm start. Now installations can switch checkpoint modes using the $T CKPTDEF,MODE=DUAL|DUPLEX command. An all member warm start is not required. Changing to DUPLEX mode is always allowed. Changing to DUAL mode is NOT allowed when either CKPTn or NEWCKPTn is on the coupling facility or if DUPLEX=OFF is specified on the CKPTDEF initialization statement and both CKPTs are in use.

Note: You can use the $T CKPTDEF,MODE=DUAL|DUPLEX command even though some MAS members are not at the JES2 z/OS V1R2 release.

Migration Actions
- Update your procedures that change the MODE.

Operations

$ACTIVATE

Before warm starting JES2 z/OS V1R2, ensure a $ACTIVATE has been performed on JES2 OS/390 V2R8. To do this, issue a $D ACTIVATE command.

$SJ Command

In migrating from JES2 OS/390 V2R4 to JES2 z/OS V1R1, the behavior of the $SJ command changes if you have not installed the PTF to fix APAR OW38962.

- Prior to OW38962, the $SJ command only considered members of the JES2 MAS that were eligible to run a job, but only if (1) the member of the MAS did NOT

have a member draining situation (such as, when a $P is entered) and (2) the member did NOT have an initiator draining situation (such as, when a $P XEQ is entered).

- After the fix to APAR OW38962 is installed, members of the MAS that are in a member draining situation are still NOT considered. JES2 ignores the initiator draining situation, but only on the MAS member where the $SJ command is entered.

To illustrate this change in processing, suppose there are three members active in the JES2 MAS: SY1, SY2, and SY3. Furthermore, suppose that a $P XEQ is issued on SY1 and SY2, but NOT on SY3. In this situation, when a $SJ command is issued on SY3, SY3 is the only member that the job can run on, and then only if the affinity and SCHENV status permits it to run. If the $SJ command is issued on SY2, then both SY2 and SY3 are considered for running the job.

$T SPOOL(nnnn) Command

$T SPOOL(nnnn) is a new JES2 command. This commands allows the installation to assign its systems to spool volumes. This command has one valid keyword operand, SYSAFF=. The SYSAFF= keyword has a similar function SYSAFF on the $TINTRDR command. The value you specify for SYSAFF= on the $TSPOOL(nnnn) command is also added to the output produced by the $D SPOOL command.

Migration Actions
A new RACF authorization profile is required for the $T SPOOL command. The profile is JESx.MODIFY.SPOOL with ACCESS=CONTROL.

$ZAPJOB Command

$ZAPJOB removes all traces of a job structure from the JES2 job queue. You must ensure that the job and any output it may have created is not active in any JES2 process or active on a device. Zapping a job that is active can lead to ABENDs and the loss of a PCE until JES2 is restarted.

There is nothing to stop entering a $ZAPJOB command in the JES2 initialization stream, however, IBM recommends that the ZAPJOB initialization statement be used instead. A ZAPJOB command in the initialization stream will run after warm start processing completes whereas a ZAPJOB initialization statement will be processed before warm start processing begins, before verifying the job queue.

$HASP250 now indicates when a job has been zapped.

Migration Actions
A new RACF authorization profile is required for the $ZAPJOB command. The profile is JESx.ZAP.JOB with ACCESS=CONTROL.

$HASP250 has been changed to add that a job has been zapped.

$HASP834 Deleted

$HASP834 message has been deleted.

Migration Actions
Examine your installations message automation. There is no replacement for $HASP834.

Z2 Mode

Use the JOBDEF initialization statement to increase the amount of job numbers you can have in your installation. In **z2 Mode**, JES2 allows you to define up to 999,999 job numbers (expanding the limit from 65534).

With **z2 Mode**, you will be able to define up to 999,999 job numbers. Currently, the limit is just 65,534. In order to support the larger job numbers, the jobid format is changed. The format used depends on the upper limit on the RANGE operand of the JOBDEF initialization statement.
- If the upper limit is > 99,999, then the jobid is Jxxxxxxx
- If the upper limit is <= 99,999 then the jobid is JOBxxxxx
- If the upper limit is > 99,999, then the jobid is Txxxxxxx
- If the upper limit is <= 99,999 then the jobid is TSUxxxxx
- If the upper limit is > 99,999, then the jobid is Sxxxxxxx
- If the upper limit is <= 99,999 then the jobid is STCxxxxx

Note:

> JES2 uses the appropriate jobid format in displays of job information. There is a transition period after JOBDEF RANGE= is increased above 99,999 where the "old" format jobids may still be displayed. This can occur for jobs that were created when the old range was being used. Until these jobs are purged from the system, the old format for jobids may be displayed as part of informational messages about those specific jobs.

See *z/OS JES2 Initialization and Tuning Reference* for the specific changes to the JOBDEF initialization statement

Operator commands that process jobs by job number are improved to handle the increase of job numbers.

$T JOBDEF,RANGE= is updated to allow you to set a job number limit up to a maximum of 999,999. Also, the following $T command changes allow you to take advantage of new maximum for the number of jobs, number of JOEs and number of BERTs as well as to decrease their settings:
- $T JOBDEF JOBNUM= for the number of jobs
- $T OUTDEF JOENUM= for the number of JOEs
- $T CKPTSPACE BERTNUM= for the number of BERTs

JOBID Formats

The format of job numbers being displayed as part of command responses or messages has changed in z2 mode. Currently job numbers are displayed as JOBnnnnn. Five characters is not large enough to display job numbers greater than 99,999. Therefore, job numbers will be displayed differently when job numbers greater than 99,999 are used or potentially used.

The "new" job number format will be as follows: If the maximum allowed job number (high value of the JOBDEF RANGE= statement) is above 99,999 then the job number format will be J0nnnnnn. This format will always be used unless the job number range is decreased below 100,000. Similarly, STCnnnnn will become S0nnnnnn and TSUnnnnn will become T0nnnnnn.

If the job number range is increased above 99,999 or decreased below 100,000 with the $T command, a transition period will exist where the old job number format and the new job number format will be displayed as part of commands, messages, and SMF records.

Rules for determining job number format:
- If the local maximum is > 99,999 then use the new format.
- If the local maximum is <= 99,999 then use the old format as long as the job number being displayed is<= 99,999.

Note: If you are in **z2 Mode** and RASSIGN=YES is specified (or defaulted) on the JOBDEF initialization statement, the NJE jobs could use the Jxxxxxxx form of jobid if the job number is > 99,999; this occurs regardless of the RANGE setting.

Migration Actions
- Reread "z2 Mode for $ACTIVATE" on page 17 to understand how to use the $ACTIVATE command to bring your JES2 to **z2 Mode** and what are the changes to JES2 job number processing.
- Examine and change your automation with respect to commands, messages, and job numbers.
- Implement appropriate procedures to return to **R4 Mode**.

Spinning JESlog Data Sets for Long Running Jobs

Spinning or suppressing JESlog data described in "Spinning JESlog Data Sets for Long Running Jobs" on page 27. Refer to this section for an explanation of this function. A description of migration actions follow.

Migration Actions
- Examine your JOBCLASS(v) initialization statements and change as appropriate to use this function.
- Your installation workload of jobs might require the use of JESLOG.

Attention:
- Jobs that are converted on a JES2 z/OS V1R2 MAS member but run on a pre-JES2 z/OS V1R2 MAS member cannot perform the JESLOG spin off.
- Jobs that are converted on a JES2 z/OS V1R1 (or earlier) MAS member cannot perform JESLOG spin off regardless of which MAS member they run on.
- The converter (and START commands) cannot recognize the JESLOG in a JES2 z/OS V1R1 (or earlier) MAS member. This condition produces an error.

Examine how your installation responds to the following messages:
- HASP138
- HASP860
- HASP130

See *z/OS JES2 Messages* for details.

Dynamic JES2 PROCLIB Commands

JES2 z/OS V1R2 provides a PROCLIB initialization statement you use to define a logical DD name that the converter can use. Up to 255 data set names can be associated with the PROCLIB initialization statement. Dedicated PROCLIB data set concatenations can be eliminated.

Migration Actions

- Use the PROCLIB initialization statements in your initialization stream.
- Understand the following commands:.
  - $ADD PROCLIB creates a new dynamic PROCLIB concatenation.
  - $DEL PROCLIB deletes an existing dynamic PROCLIB. This command does not affect static PROCLIBs defined in the JES2 PROC.
  - $T PROCLIB updates an existing dynamic PROCLIB
  - $D PROCLIB displays the current dynamic PROCLIBs only and does not display the static ones. This command has a DEBUG option that provides additional information about the dynamic PROCLIB concatenation as well as logically deleted dynamic PROCLIBs that are still in use.
- This function can operate independent of mode (**z2 Mode** or **R4 Mode**).

See *z/OS JES2 Initialization and Tuning Reference* and *z/OS JES2 Commands* for more details.

Large Volume Support

JES2 supports both a relative and absolute track addressing for SPOOL. The SPOOLDEF initialization statement is updated to allow you to specify whether you **always**, **as needed**, or **never** want to use relative MTTRs in processing the JES2 SPOOL. You use the new keyword, RELADDR= with a value of ALWAYS , ASNEEDED, or NEVER to specify your choice.

Notes:

1. Relative addressing in processing the JES2 SPOOL is only used if all members of the MAS support relative addressing. Any member that is not supporting relative addressing fails to initialize and a HASP401 message results.
2. The size of a SPOOL data set is still limited to 64K tracks.

Migration Actions

- Examine the RELADDR= on the SPOOLDEF initialization statement to ensure it is appropriate for the volume. Make adjustments as necessary.
- Ensure that all members of the MAS use relative addressing.
- This support works in **z2 Mode** or **R4 Mode**.

Checkpoint Improvements

Installations can switch checkpoint modes using the $T CKPTDEF,MODE=DUAL|DUPLEX command. An all member warm start is not required. Changing to DUPLEX mode is always allowed. Changing to DUAL mode is NOT allowed when either CKPTn or NEWCKPTn is on the coupling facility or if DUPLEX=OFF is specified on the CKPTDEF initialization statement and both CKPTs are in use.

Note: You can use the $T CKPTDEF,MODE=DUAL|DUPLEX command even through some MAS members are not at the JES2 z/OS V1R2 level.

Migration Actions

- Update your procedures that change the MODE.

HASP443 Message

The HASP443 message clarifies the reason for a SPOOL allocation failure.

Migration Actions
- Examine any automation that processes this message.

Command Syntax Migration Aid Removal

In JES2 OS/390 V2R4, JES2 provided a command syntax compatibility and migration aid in the form of an automatically installed exit 5 routine. The migration aid was intended as a temporary bridge to give customers time to adapt to the command syntax changes made in OS/390 JES2 Version 2 Release 4. As of JES2 z/OS V1R2, the migration aid is no longer automatically installed and enabled. If you are not using the command syntax documented in *z/OS JES2 Commands* and are still relying on the migration aid, you can experience erroneous system operation or receive HASP649, HASP650, HASP003 messages.

The following table lists the commands translated and the results of that translation.

Table 20. Command Translation Formats

JES2 OS/390 V2R4 Format	JES2 z/OS V1R2 Format
`$D'name',...`	`$DJOBQ'name',CMDAUTH=*,...`
`$T'name',...`	`$TJOBQ'name',...` **(See NOTE 1)**
`$DJ1,2,...`	`$DJ(1,2),...` **(See NOTES 2,3)**
`$DJ1-2,J3-4,...`	`$DJ(1-2,3-4),...` **(See NOTE 3)**
`$LJnnn,ALL`	`$DOJnnn`
`$LJnnn,H`	`$DOJnnn,HELD`
`$LJnnn,READY`	`$DOJnnn,READY`
`$LJnnn,OUTGRP=xxx`	`$DOJnnn,OUTGRP=xxx`
`$CJnnn,OUTGRP=xxx`	`$COJnn,OUTGRP=xxx`
`$PJnnn,OUTGRP=xxx`	`$POJnnn,OUTGRP=xxx` **(See NOTE 4)**
`$PJnnn,Q=x`	`$POJnnn,Q=x`
`$vJnnn,A=|Days=|Hours=`	`$vJnnn,A>|Days>|Hours>`
`$TJnnn,S=sid1,sid2,...`	`$TJnnn,S=(sid1,sid2,...)`
`$DSPL,JOBS=nn`	`$DJOBQ,SPOOL=(PERCENT>=nn)`
`$SSPL,V=xxxxxx,...`	`$SSPL(xxxxxx),...`
`$vIxx`	`$vI(xx)`
`$vIxx-yy`	`$vI(xx)` **(See NOTE 5)**
`$vJxxxx,Txxxx...`	`$vJOBQ(xxxx,xxxx)` **(See NOTE 6)**
`$TIxx,class-list`	`$TI(xx),C=class-list`
`$HQ,ALL`	`$TJOBCLASS(*),QHELD=Y`
`$HQ,C=xyz`	`$TJOBCLASS(x,y,z),QHEL=Y`
`$AQ,ALL`	`$TJOBLCASS(*),QHELD=N`
`$AQ,C=xyz`	`$TJOBCLASS(x,y,z),QHELD=N`
`$PQ,ALL,...`	`$POJOBQ,READY,...`
`$PQ,Q=xyz`	`$POJOBQ,READY,Q=xyz`
`$OQ,ALL,...`	`$OJOBQ,/R=LOCAL.*,...`
`$OQ,Q=xyz,...`	`$OJOBQ,/R=LOCAL.*,/Q=xyz,...`
`$TALL,sid1,sid2,...`	`$TJOBQ(*),/S=(sid1,sid2,...)`

Table 20. Command Translation Formats (continued)

JES2 OS/390 V2R4 Format	JES2 z/OS V1R2 Format
`$LSYS`	`$DMEMBER`
`$ESYS,sid`	`$EMEMBER(sid)`
`$ESYS,RESET=sid`	`$ECKPTLOCK,HELDBY=sid`
`$TSYS,IND=Y|N`	`$TMEMBER(local),IND=Y|N`

Notes:

1. Changes are similar for $A, $C, $E, $H, $L, $O, $P, $T, $TO
2. J can be J, JOB, S, STC, T, TSU
3. Changes are similar for $A, $C, $E, $H, $L, $O, $P, $TO
4. Unless Q= is a valid job queue (XEQ, PPU, etc.)
5. Where v is one of the command verbs D, P, S, T, or Z. This command processed a named range of initiators. You must now issue these commands as separate commands.
6. Where v is one of the commands verbs A, C, E, H, L, O, P, T, TO. This command processed a list of JOBs, TSO users, and STCs. You must now issue these commands either as multiple commands ($v Jxxxx... and $v Txxxx...) or use $v JOBQ(xxxx,xxxx). The JOBQ version has slightly different function (does not check job type).

Migration Actions

If you are exclusively using the command syntax documented in the *z/OS JES2 Commands* publication and are not relying on the command migration aid exit, then no action is required. If you do not know whether or not you are currently relying on the command migration aid exit, you need to do the following:

1. First determine if the migration exit is installed. Issue $DEXIT5 on your current production systems and if the command response does not include routine name HASX5CTR, then no further action is required.

2. If the exit is installed, look through SYSLOG for $HASP006 messages which display translated commands. The fact that commands are translated does not necessarily indicate dependence on the migration aid exit because it conservatively translates several command formats that already have valid syntax. Commands with verbs with a single subscript or subscript range that are enclosed in parentheses are frequently unnecessarily translated as shown in the following examples:

   ```
   $HASP006 $DJ5   TRANSLATED BY EXIT HASX05C TO $DJ(5)
   $HASP006 $DJ5-6 TRANSLATED BY EXIT HASX05C TO $DJ(5-6)
   ```

   If however, you find a $HASP006 message in SYSLOG with any other form, or in the above form but having additional operands that are altered, then you are currently dependent on the migration aid exit. If you find you are currently dependent on the migration aid exit, then determine the origin of the obsolete command syntax and take corrective action. Either:

   - Have your operators begin using only the supported command syntax and convert your automation scripts to use valid syntax. Refer to the *z/OS JES2 Commands* for correct syntax of JES2 commands. IBM recommends you begin this task immediately on your current production systems and analyze $HASP006 messages in SYSLOG to measure your progress.

   OR

   - Although IBM strongly recommends against it, install the command migration aid exit yourself and assume the risks described in the **CAUTION**

below. As of HJE7705, the command migration aid exit is now shipped as member HASX05C in SYS1.SHASSAMP.

> **CAUTION !**
>
> **The JES2 command migration aid, exit 5 routine HASX5CTR, is now shipped in member HASX05C of SYS1.SHASSAMP and is no longer supported. This exit is not warranted and is functionally stabilized. You are strongly urged to start eliminating your dependency on this exit now and stop using the exit as soon as possible. Failure to make an orderly transition now could result in a prematurely rushed transition later.**

Application Development

APPLCOPY No Longer Supported

APPLCOPY is no longer supported.

APPLCOPY is a way to access data in the checkpoint, but is replaced with checkpoint versions. (SSI 71) applications that used APPLCOPY (such as SDSF) have migrated to now use checkpoint versions. Becuase of installation overhead concerns, especially with very large checkpoint data sets, support for APPLCOPY has been discontinued. The CKPTDEF APPLCOPY= keyword is dropped and using it will generate an error during initialization (causing other operands on CKPTDEF to be ignored).

Migration Actions
- Installations should delete references to APPLCOPY in their initialization decks.
- Installations that have an application that still uses APPLCOPY, need to have that application migrate to the use of checkpoint versions before installing JES2 z/OS V1R2.

Problem Determination and Diagnosis

Provide for JES2 IPCS Support

> **Attention!**
>
> - You must be proactive to install JES2 IPCS.
> - Make sure JES2 IPCS support works before you experience an emergency.
> - Make sure you have JES2 IPCS support for all combinations of JES2 and MVS releases in production. (*z/OS JES2 Migration* and *z/OS JES2 Diagnosis* provide information about JES2 IPCS support.)

- When JES2 dumps are being diagnosed on the same level of the system as the system on which the dumps were taken, do the following:
  - Ensure that SHASPARM is specified in the PARMLIB concatenation.
  - Ensure that SHASMIG is specified in the STEPLIB concatenation.
  - Ensure that SHASPNL0 is specified in the ISPPLIB concatenation.

This ensures that IPCS can find the SMP/E-maintained copy of JES2 data.

- If the JES2 levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, but the level of MVS is the same, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also specify your PARMLIB data set, plus any other data sets containing IPCS parameters that you normally need when using IPCS.
  - Specify the SHASMIG library that corresponds to the JES2 level in the dump in the first position in the STEPLIB concatenation.
  - Specify the SHASPNL0 library that corresponds to the JES2 level in the dump in the first position in the ISPPLIB concatenation.
- If both the JES2 and system levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also, specify the PARMLIB data set, plus any other data sets containing IPCS parameters that correspond to the system level in the dump.
  - Specify the SHASMIG and MIGLIB libraries that correspond to the system in the dump in the first and second positions in the STEPLIB concatenation.
  - Specify the SHASPNL0 and SBLSPNL0 libraries that correspond to the system in the dump in the first position in the ISPPLIB concatenation.

Note: IPCS requires libraries in addition to the libraries that JES2 uses. You may need to make other concatenations. For more information, see *z/OS MVS IPCS Customization*.

If dump analysis leads into data structures for other elements or products, you might need to add more data sets to the concatenations to complete the diagnosis. Refer to the appropriate product publications for information about establishing the correct IPCS environments for diagnosing other elements and products.

Dumping all Members of the MAS

z/OS V1R1 JES2 dumps all members of the MAS when certain JES2 work queue errors occur. This is referred to as "multi-system dumps".

Migration Actions
You need to make sure you have the procedures in place to collect all the dumps from all the systems.

HASP443 Message

The HASP443 message is updated to better clarify the reason a SPOOL allocation failed. One or more "reason code" can be displayed with this message to further clarify this conditon.

Migration Actions
- Review how you examine the HASP443 message.
- If you have automation that interprets HASP443, then you might need to make changes.

Auto Reply to Termination WTORs

Automatic responses to HASP070, HASP089, and HASP098 occur if there is no operator response for 10 minutes. See *z/OS JES2 Messages* for message, HASP065 for more detail.

JES2 Installation Modifications

Active DCTs

Active DCTs are now chained out of the HCT ($NJEADCT, $OFFADCT, $LCLADCT).

Note: If an error occurs in the DCT chain, a SYSOUT device may not be selected for work. Such might be the case when $S command is issued. JES2 rebuilds this chain should errors occur.

Migration Actions
Installations that manipulate DCTs need to be aware of these changes.

Modification Considerations

If you plan modifications to JES2 beyond using exit routines, your migration actions will be similar to those for exit routines. See "z/OS JES2 Version 1 Release 2 Overview" on page 17 and review each item in JES2 z/OS V1R2. The migration actions enumerated for these items will apply to your own special modifications.

Distribution Library Name Changes

Previous library data set names included version, modification, and release values (for example,SYS1.V2R10M0.SHASLINK. In JES2 z/OS V1R2, the *"vrm"* qualifier (which in the example is V2R10M0) is eliminated (For example, SYS1.SHASLINK is the proper name.).

Migration Actions
Change your installations JCL appropriately, including TSO and LOGON PROCs for IPCS.

Counting Null Records

Prior to JES2 z/OS V1R2, null records written to SYSOUT data sets were padded to a single blank record. Because of this padding, these records were counted as records in the job totals. This count is not consistent with SYSOUT received over NJE. Null records from NJE are not padded and not counted in the job totals.

Migration Actions
Since the processing to pad null SYSOUT records was deleted in JES2 z/OS V1R2, counting may change record counts for existing jobs. You would notice this change in a mixed MAS (JES2 z/OS V1R2 running with earlier levels of JES2).

Chapter 8. Migration Actions: JES2 OS/390 V2R5 to JES2 z/OS V1R4

Customization

New and Changed Installation Exits

Exit	Explanation	Migration Action
Exit 8	This exit can now receive control for control block I/O (CBIO) calls from the FSS address space. A new bit was added to the XPL to identify that this $CBIO was done for an FSS printer.	Check new bit if you want to ignore calls for the FSS printer.
Exit 25	The $BUFIO and $BUFCK services in HASPFSSM have been deleted. These services are now provided by the $CBIO service. TYPE=WAIT was added to $CBIO to support the functions that were previously provided by $BUFCK.	If you were using $BUFIO or $BUFCK services, now use $CBIO.
Exit 46	Exit 46 will now be invoked to process RCCS headers.	Insure that your exit 46 routine is coded to accomodate RCCS headers, espically if you have patterned your exit 46 routine after the sample exit 46 routine (HASX46A) shipped with the product. See the sample exit 46 routine shipped with OS/390 Rlease 7 for an example of how to write code to accomodate RCCS headers.
Exit 47	Exit 47 will now be invoked to process RCCS headers.	Insure that your exit 47 routine is coded to accomodate RCCS headers.
Exit 47	Exit 47 will not always be passed the address of a real JQE.	Test indicator X047IND for setting X047BJQE. If on, X047JQE contains the address of a working copy of the JQE, not of a real JQE. Do not use this address as input to any services that expect the address of a real JQE (for example, $DOGJQE).
Exit 70	Exit 70 function (added by APAR OW32040) is now moved to Exit 46.	If the NJE receivers in your network are still down-level, then either move your exit 70 function to exit 46, or change the statements in your INIT deck to invoke your exit 70 routine as exit 46.

Installation Exits 11 and 12

Installation exits 11 and 12 are unaffected if you use no new functions in z/OS V1R1 JES2. However, if you turn on spool affinity or set fencing to more than one volume, then your current exits 11 and 12 are affected.

Migration Actions: Review the documentation for exits 11 and 12 in *z/OS JES2 Installation Exits*. Before activating SPOOL affinity, ensure that the correct available SPOOL masks are used to determine what volumes to add to a job in these exits.

Your installation uses spool partitioning if `FENCE=ACTIVE=YES` is specified on the SPOOLDEF initialization statement. Spool affinity is used if any spool volume has something other than SYSAFF=ANY displayed for it.

Installation Exits 0, 19, 24

The HASPINIT load module and the initialization PCE have been moved above the 16M line. This may impact installation exits called during initialization processing (such as exits 0, 19 and 24). If your exit switches to AMODE 24 to invoke DFP modules, then this code will no longer be able to access the initialization PCE (and in particular the save area in the PCE).

Migration Actions: Review all exits for code that switches to AMODE 24 ($AMODE 24 macro invocations). Verify that the switch to 24 bit mode is still required. If it is required and you are in an exit that gets control during initialization, ensure that the code running in AMODE 24 does not access the initialization PCE.

Also, examine all uses of DFP macros, such as GET and PUT. If these macros were hand coded (instead of using the IBM-supplied macro) changes might be necessary. The DFP macros were changed to support 31 bit addressable callers.

Exit 5

In z/OS V1R2 (HJE7705), the command syntax migration aid exit 5 routine, HASX5CTR in module HASX05C, is no longer automatically installed and enabled. As of HJE7705, the exit is now shipped as a sample in SYS1.SHASSAMP and is no longer supported. The function of this module was intended only as a migration aid to translate pre-OS/390 JES2 Version 2 Release 4 command syntax to OS/390 JES2 Version 2 Release 4 command syntax.

Migration Actions: IBM recommends you use the JES2 command syntax as documented in *z/OS JES2 Commands*. If you still need command translation, you need to install and enable EXIT 5 (HASX05C).

Note: The function of HASX05C is not altered from JES2 OS/390 V2R4. See Command Syntax Migration Aid Removal in the ″Operations″ section of this chapter for more information.

Exit 14 and Exit 49

PITCLASS and QGTCLST are changed.

In z/OS V1R2, QGTCLST now points to a work area, which contains the class list for this selection; this class list is a subset of the classes listed in PITCLASS. PITCLASS contains the class list for the initiator as defined by the INIT(nnnn) initialization statement; QGTCLST contains the class list for this selection and can contain different values than PITCLASS.

Migration Actions: If exit 14 and exit 49 in your installation reference the fields PITCLASS or QGTCLST, you need to examine these references and make appropriate changes.

Field, Field Name, and Control Block Changes

For JES2 to use up to 999,999 job numbers requires that you activate your JES2 z/OS V1R2 MAS to **z2 Mode** . Unless this activation is performed, JES2 operates in **R4 Mode**, and JES2 z/OS V1R2 behaves with the job number limitations to 65534.

Migration Actions:

> **Attention !**
>
> In migrating to JES2 z/OS V1R2, it is important to refer to "General Field and Control Block Changes" on page 19 for the needed details about all specific fields, field names, and control block changes that take place both when migrating to JES2 z/OS V1R2 and when you activate.

In preparing for your migration to JES2 z/OS V1R2 you need to analyze:

- All your exit routines and "unique" code that would be affected by activating JES2 z/OS V1R2 in **z2 Mode**
- How your JES2 system processess the JOE, JQE, JQA, and JOT control blocks that are significantly changed in **z2 Mode**.
- Control blocks and fields used to store job numbers (such as the $SJB,$ JIB, $FSAXB, and $COMWORK).
- How to use JES2 macro services, such as $QJQE, $#JOE, $DOGJQE, $JBIDBLD.
- Your assembler errors that result from not thoroughly finding all places where fields, field names, and control block changes affect JES2.
- Your JES2 exits and how they are affected, making necessary changes, re-assembling, and integrating them.
- If your exits use JQE or JOE chaining fields or reference job number fields, then **z2 Mode** changes require updates to your exit routines. Field names have been changed to help identify where logic changes are needed.

Macro Updates Relating to JOE and JQE Changes

Some macros are created and updated to allow JES2 exit routines to be mode insensitive.

$QJQE and $#JOE

These macros process the JQE and JOE control block chains using proper chaining fields based on the current checkpoint processing mode (either **R4 Mode** or **z2 Mode**. Therefore, users of these macros do not have to be sensitive to which JES2 processing mode is currently being used.

$#JOE is changed to process chains (such as the CHAR JOE chain). Both $QJQE and $#JOE have improved their loop control and have usability enhancements.

$DOGJQE

The JQA returned by $DOGJQE always reflects the **z2 Mode** processing format. Code that processes an artificial JQE (JQA) need not examine the JES2 processing mode (**z2 Mode** or **R4 Mode**) to examine JQA fields.

$JQEJNUM

$JQEJNUM is a new macro used to obtain the binary job number for a particular JQE. This macro operates in **z2 Mode** and **R4 Mode** and processes the JQA or JQE that is passed.

$JBIDBLD

This macro converts a binary job number to a printable jobid. $JBIDBLD is updated to accept a JQE (or JQA) as input and operates in **z2 Mode** or **R4 Mode**. $JBIDBLD formats the jobid, in the correct way, based on the current job number range.

Migration Actions:

- Review your installation exit routines and thoroughly understand what changes you might need to make because of the various field, field names, and control block changes.

- Convert any "unique" installation code you use to use the new fields and field names.

- Process JOEs and JQEs as much as possible with the macros provided ($QJQE, $#JOE, $DOGJQE, $JQEJNUM, $JBIDBLD) instead of formulating your own unique code.

- Ensure that all the field and control block changes identified with JES2 z/OS V1R2 are incorporated, including updates to the surrounding logic.

$XMPOST Macro

- $XMPOST

  The $XMPOST macro is changed. When you specify $XMPOST XMPLIST=YES (or leave XMPLIST off and let it default), the mapping of the input parameter list is changed. The expected parameter list mapping is as follows:

  – +0 Error return address

  – +4 Address of ECB to be POSTed

  – +8 Address of related ASCB

  – +C ECB to be POSTed (if LOSTPOST=CHECK is specified on the macro call)

$QLOCNXT Macro

$QLOCNXT finds the next allocated job number after a specified job number.

New and Changed Macros

$DOGCAT

The $DOGCAT macro provides an interface to find the address of a CAT. It is also recommended that the CAT obtained using $DOGCAT be used for finding queue heads rather than $JQHEADS.

Migration Action: Use $DOGCAT wherever your code has used $CATABLE to compute the address of a CAT. Use the CAT found via $DOGCAT to access the job queue head for the class.

$DOGJQE

The $DOGJQE macro provides an interface to construct an artificial JQE. An artificial JQE is constructed in a temporary work area and is composed of the real JQE, a JQE extension (new in this release) and fields copied from an internal area (also new).

You need to be aware of the types of JQE/JQAs that JES2 provides to your exit, since JES2 processes these JQE/JQAs in differing ways. The types are:

- A real JQE. Your exit receives a read or update mode JQE/JQA.

- A read-mode JQA. Your exit receives an artificial JQE that is a temporary block of storage. This storage contains:

  – Almost the same information as the real JQE.

  – Information from the JQX (new in Version 2 Release 4).

  – Information from BERTs (another checkpointed area).

- An update-mode JQA. Your exit receives an artificial JQE that is a temporary block of storage. This storage is similar to the read-mode JQA. JES2 ensures the integrity of this JQA and manages the storage each JQA occupies.

- A work area containing a prototype JQE. In certain circumstances, your exit may be passed the address of a work area that contains a working copy of a JQE. See Exit 47 for more information about this.

Migration Action: Use $DOGJQE services rather than $CKPT if your exit modifies JQEs.

Ensure that your exit calls TIMECLOC via:

```
$CALL   TIMECLOC,PARM=JQE,PARM0=0
```

if the hold status of the JQE is changed or if the system affinity of the JQE is changed.

Exits normally want to use JQEs in read mode (data is extracted or pointed to when calling service routines) or in write mode (data in the JQE is modified). JES2 exit writers need to take the following actions when using a particular JQE/JQA as the JQE= keyword value on the $DOGJQE macro:

- If the JQE is needed only to access data and that data is within the bounds of the original real JQE, then only the address of the real JQE is needed. Regardless of what IBM has provided as the JQE address, use the following action to get the address of the real JQE:

  ```
  $DOGJQE ACTION=GETJQEADDR,CBADDR=jqe
  ```

- If the JQE is needed only to access data and that data is beyond the bounds of the original real JQE (that is, it is stored in fields where the first three characters of the field name are other than JQE), then a read mode JQA is needed. Regardless of what IBM has provided as the JQE address, use the following action to get the address of a read mode JQA. The address of the read mode is passed back (in R0).

  ```
  $DOGJQE ACTION=(FETCH,READ),JQE=jqe
  ```

- When you are finished, use the following action to free the memory used for the JQA (x is the address returned from the first $DOGJQE call):

  ```
  $DOGJQE ACTION=RETURN,CBADDR=x
  ```

- If the JQE is needed in write mode (the fields to be changed are either within the bounds or not within the bounds of the original JQE), use the following action to get the address of an update mode JQE, regardless of what IBM has provided as the JQE address. The address of the JQA is passed back (in R0). Make all changes to fields in the update mode JQA.

  ```
  $DOGJQE ACTION=(FETCH,UPDATE),JQE=jqe
  ```

- When you are finished, use the following action to free the memory used for JQA (x is the address from the first $DOGJQE call) and to ensure that the changes in the JQA get propagated to the real JQE, the JQX, and the BERT area.

  ```
  $DOGJQE ACTION=RETURN,CBADDR=x
  ```

When your exit returns a JQE/JQA to the JES2 systems through these actions, certain errors can occur if JES2 determines that what your exit has returned is not consistent with what JES2 knows to exist. JES2 uses the $ERROR macro and issues the following errors:

- DJ1- non-IBM code returned an IBM JQE/JQA that violates the consistency checks of JES2.
- DJ2- IBM code returned a non-IBM JQE/JQA that violates the consistency checks of JES2.

Notes:

1. You are encouraged to disregard the kind of JQE/JQA passed to your exit and always do the following:
   - To obtain the address of the real JQE (for example, your exit wants to compute the offset of the JQE), use:

     `$DOGJQE ACTION=GETJQEADDR`
   - To obtain the address of a read mode JQE/JQA (for example, your exit wants to examine the MAXCC field), use:

     `$DOGJQE ACTION=(FETCH,READ)`
   - To obtain the address of an update mode JQE/JQA (for example, your exit wants to change the SYSAFF, PRIORITY or MAXCC), use:

     `$DOGJQE ACTION=(FETCH,UPDATE)`

2. If you are writing exit 47, do not use $DOGJQE to access a JQE/JQA.

$DVIDBLD

In prior releases, the name associated with a device could be obtained directly from DCTDEVN. Now, some devices that have a device number greater than 9999 will have a name in DCTDEVN that is not consistent with the name normally displayed for a device. Use the $DVIDBLD macro to obtain a device name suitable for use in messages or other displays. The name returned from $DVIDBLD could be up to 10 characters long.

Migration Action: Review any exit that references DCTDEVN and if the field is being displayed, consider using $DVIDBLD to display a more consistent name for the device.

$QJQE

The $QJQE macro is used to traverse JQE queues. It has changed to always return the address of an artificial JQE.

Migration Action: If you need the address of a real JQE, for example, to compute the address of a real JQE offset, use the $DOGJQE ACTION=GETJQEADDR service to return the address of a real JQE.

$QGET

$QGET has been changed to return the address of an artificial JQE.

Migration Action: To return the memory for an artificial JQE, use one of the following: $DOGJQE ACTION=RETURN, $QMOD, or $QPUT.

$DILBERT

The $DILBERT macro provides an interface to delay operations upon an artificial JQE if that JQE is locked.

Migration Action: If it is necessary to get an artificial JQE in update mode and your code cannot afford to $WAIT for the JQE to become unlocked, then use the $DILBERT macro to automatically defer the action until it is.

$#BLD

The $#BLD macro is used to create a JOE. One of its inputs is a JQE address. In prior releases, the JQE was passed as an offset. Starting in Release 4, the JQE is passed as an address, and the address passed is the address of a real JQE.

Migration Action: Ensure any uses of $#BLD use the JQE address rather than the offset.

$PAIR
If the installation defines their own table pairs (either in line or in the UCT or MCT) they must be defined using the $PAIR macro.

Field, Field Name, and Control Block Changes

For JES2 to use up to 999,999 job numbers requires that you activate your JES2 z/OS V1R2 MAS to **z2 Mode** . Unless this activation is performed, JES2 operates in **R4 Mode**, and JES2 z/OS V1R2 behaves with the job number limitations to 65534.

Migration Actions

> **Attention !**
> In migrating to JES2 z/OS V1R2, it is important to refer to "General Field and Control Block Changes" on page 19 for the needed details about all specific fields, field names, and control block changes that take place both when migrating to JES2 z/OS V1R2 and when you activate.

In preparing for your migration to JES2 z/OS V1R2 you need to analyze:
- All your exit routines and "unique" code that would be affected by activating JES2 z/OS V1R2 in **z2 Mode**
- How your JES2 system processess the JOE, JQE, JQA, and JOT control blocks that are significantly changed in **z2 Mode**.
- Control blocks and fields used to store job numbers (such as the $SJB,$ JIB, $FSAXB, and $COMWORK).
- How to use JES2 macro services, such as $QJQE, $#JOE, $DOGJQE, $JBIDBLD.
- Your assembler errors that result from not thoroughly finding all places where fields, field names, and control block changes affect JES2.
- Your JES2 exits and how they are affected, making necessary changes, re-assembling, and integrating them.
- If your exits use JQE or JOE chaining fields or reference job number fields, then **z2 Mode** changes require updates to your exit routines. Field names have been changed to help identify where logic changes are needed.

Macro Updates Relating to JOE and JQE Changes

Some macros are created and updated to allow JES2 exit routines to be mode insensitive.

$QJQE and $#JOE
These macros process the JQE and JOE control block chains using proper chaining fields based on the current checkpoint processing mode (either **R4 Mode** or **z2 Mode**. Therefore, users of these macros do not have to be sensitive to which JES2 processing mode is currently being used.

$#JOE is changed to process chains (such as the CHAR JOE chain). Both $QJQE and $#JOE have improved their loop control and have usability enhancements.

$DOGJQE
The JQA returned by $DOGJQE always reflects the **z2 Mode** processing format. Code that processes an artificial JQE (JQA) need not examine the JES2 processing mode (**z2 Mode** or **R4 Mode**) to examine JQA fields.

$JQEJNUM
$JQEJNUM is a new macro used to obtain the binary job number for a

particular JQE. This macro operates in **z2 Mode** and **R4 Mode** and processes the JQA or JQE that is passed.

$JBIDBLD

This macro converts a binary job number to a printable jobid. $JBIDBLD is updated to accept a JQE (or JQA) as input and operates in **z2 Mode** or **R4 Mode**. $JBIDBLD formats the jobid, in the correct way, based on the current job number range.

Migration Actions

* Review your installation exit routines and thoroughly understand what changes you might need to make because of the various field, field names, and control block changes.
* Convert any "unique" installation code you use to use the new fields and field names.
* Process JOEs and JQEs as much as possible with the macros provided ($QJQE, $#JOE, $DOGJQE, $JQEJNUM, $JBIDBLD) instead of formulating your own unique code.
* Ensure that all the field and control block changes identified with JES2 z/OS V1R2 are incorporated, including updates to the surrounding logic.

$XMPOST Macro

* $XMPOST

The $XMPOST macro is changed. When you specify $XMPOST XMPLIST=YES (or leave XMPLIST off and let it default), the mapping of the input parameter list is changed. The expected parameter list mapping is as follows:

– +0 Error return address
– +4 Address of ECB to be POSTed
– +8 Address of related ASCB
– +C ECB to be POSTed (if LOSTPOST=CHECK is specified on the macro call)

$QLOCNXT Macro

$QLOCNXT finds the next allocated job number after a specified job number.

Initialization

Checkpoint Performance

JES2 checkpoint performance has been improved by reducing the number of I/O to the checkpoint data sets under certain conditions. These changes include:

* Elimination of a Read to CKPT2 during checkpoint locking operations (READ1) if at least one checkpoint is on a coupling facility.
* Reduction of the number of writes to CKPT2 when checkpoint is in duplex mode.

Migration Actions

As a result of these changes, the specifications for MASDEF HOLD= and DORMANCY= may need adjusting. In particular, if you have HOLD=0 specified in duplex mode, then prior to this change, the actual hold time was limited by the time it took to write to CKPT2. Now, a significant number of those writes have been eliminated and your actual hold time may become too short and cause thrashing.

If you are in duplex mode and have some members which specify DUPLEX=OFF, you may want to reconsider this specification. As of Release 7, the CKPT2 data set is only written once every 10 writes to the CKPT1 data set. One of the reasons to specify DUPLEX=OFF was to reduce the I/O's to CKPT2. The combination of the two may cause an unacceptable amount of time to elapse before CKPT2 is written. If the only reason DUPLEX=OFF was specified was to reduce the number of I/O's to CKPT2, then as with Release 7, all members should specify DUPLEX=ON.

PTF for APAR OW36796

If you are migrating from OS/390 JES2 Release 3, 4, 5, or 7, without PTF for APAR OW36796 applied, then ensure that you have not tried to change NJEDEF OWNNODE on a warmstart and replied "Y" to message HASP441 to continue initialization. If you have, then contact IBM service prior to installing z/OS V1R1 JES2.

To determine if you have attempted to change OWNNODE on a warm start, examine the syslogs from the most recent restart of JES2 for each member of your MAS. Look for the following sequence of messages:

```
HASP442   INITIALIZATION STATEMENTS CONFLICTING WITH
             SAVED VALUES

HASP496   NJEDEF OWNNODE=xx CANNOT BE CHANGED FROM SAVED
             VALUE OF yy

HASP441   REPLY 'Y' TO TO CONTINUE INITIALIZATION OR
             'N' TO TERMINATE IN RESPONSE TO HASP442
```

You should contact IBM service if you find the above messages in your log and a reply of "Y" was given to message HASP441. If you do not find these messages with a reply of "Y", then no migration action is required.

Checkpoint Data Sets

As of z/OS V1R1, JES2 no longer supports pre-release 4 checkpoints. JES2 can only read checkpoints that are in Release 4 mode. A Release 4 checkpoint is one that has been written by JES2 Release 4 or later after a $ACTIVATE command has been executed. To determine the checkpoint mode of your current checkpoint data set issue a $D ACTIVATE command. IF the checkpoint mode is not Release 4, you must activate to the Release 4 Mode before migrating to JES2 z/OS V1R2.

Migration Actions

- If your checkpoint is **R4 Mode** (You have used $ACTIVATE.), then no additional actions are needed. JES2 z/OS V1R2 can coexist with your existing systems or warm start using your existing checkpoint data set.
- If your checkpoint is **Pre-Release 4 Mode** (You have not used $ACTIVATE.), you must do so before you can add a JES2 z/OS V1R2 system to your MAS or before performing a warm start of a JES2 z/OS V1R2 system using your existing checkpoint data set.

SPOOLDEF FENCE= Changes

The FENCE= keyword parameter on the SPOOLDEF initialization statement is changed to allow you to specify fencing along with the number of spool volumes to be used.

When you specify

```
SPOOLDEF....FENCE=(ACTIVE=YES,VOLUMES=1).....
```

you get the same function as in the previous JES2 release when you specified FENCE=YES. . This specification of SPOOLDEF FENCE= is honored on $T SPOOLDEF and on cold start.

Migration Actions

SPOOLDEF FENCE=YES is still permitted for compatibility and is converted to ACTIVE=YES and VOLUMES=1.

SPOOLDEF TGBPERVL= Dropped

JES2 processing now controls the amount of SPOOL space cached for job writing data to SPOOL (the BLOB). In previous releases, this was controlled by the TGBPERVL parameter on SPOOLDEF. TGBPERVL= is no longer supported. In the initialization stream, this keyword is ignored (no error is returned). If used in a $TSPOOLDEF command, it will generate an error. In addition the HASP834 message that indicated TGBPERVL was set too high, has been deleted.

Migration Actions

Delete references to TGBPERVL from your initialization decks and any operator commands or automation scripts. Any automated action associated with the HASP834 message should be examined and potentially eliminated.

SPOOL Utilization

As part of automatically setting the size of the SPOOL cache (the BLOB), JES2 processing increased the amount of SPOOL space that traditionally was marked as allocated. In most installations, this change would not be noticeable. However, in certain smaller installations (such as test systems) the total SPOOL space that is available may be allocated to the BLOB. This makes determining the true SPOOL utilization much more difficult. To compensate for this, JES2 now considers SPOOL space in the BLOB as available rather than as allocated (as it had in previous releases). In small installations, with limited SPOOL space, this could decrease total SPOOL utilization. Also, multiple MAS members at different JES2 levels may report different SPOOL utilizations.

Migration Actions

If you have a limited amount of SPOOL space, you may have set your SPOOLDEF TGSPACE=WARN= value high knowing that when that level was reached, you still have the space in the BLOB that can be used before SPOOL volumes are full. You should re-evaluate your settings of TGSPACE=WARN= to account for the changes in how SPOOL utilization has changes.

Also, be aware that SPOOL utilization may differ between MAS members of different release levels. This is normal. The smaller the number of track groups in the system, the greater the difference. However, the situation where the HASP355 message (SPOOL VOLUMES ARE FULL) is issued remains unchanged even in z/OS V1R1 JES2.

TPDEF SNABUF Changes

To improve SNA performance and better align with other networking products, JES2 has increased the SNA buffer size from a maximum of 3840 to 32512. This is a change to the TPDEF SNABUF=SIZE= keywords.

Migration Actions

Your VTAM network definitions should be evaluated.

TRACEDEF PAGES= Changes

The default for TRACEDEF PAGE= is changed from 2 to 9. For compatibility with other releases of JES2, specification of 1-8 are converted automatically to 9.

Migration Actions

For compatibility with other releases of JES2, specification of 1-8 for TRACEDEF PAGES, are converted automatically to 9. No migration actions are necessary, but you should evaluate your use of extended CSA space. If your current TRACEDEF initialization statements accepts the old default of PAGES=2, this will be converted to 9. For TABLES=100, you'll get 900 pages of buffers. Your setting for TABWARN= on the TRACEDEF initialization statements should also be evaluated.

ZAPJOB Initialization Statement

The ZAPJOB initialization statement is new and is used to remove all traces of a job structure from the JES2 job queue. This statement is intended for situations where a job cannot be removed using normal JES2 commands or by a JES2 restart.

Migration Actions

You should be careful when using this statement. Zapping a job that is active can lead to ABENDs and the loss of a PCE that might require a JES2 restart.

Migrating to z2 Mode

Several installation actions are required to migrate to **z2 Mode**:

- Stabilize all MAS members running JES2 z/OS V1R2 in **R4 Mode**
- `$ACTIVATE LEVEL=z2` without changing any limits
- Only when your installation is stable, running in **z2 Mode**, consider increasing limits.

Large Volume Support

JES2 supports relative and absolute track addressing for processing the SPOOL. An initialization option indicates whether relative addressing is used, never used, or only used when needed. This option ensures that relative addressing is not used before any vendor product that looks at SPOOL is ready to use relative addressing.

The SPOOLDEF initialization statement is updated to allow you to specify whether you "always", "as needed", or "never" want to use relative MTTRs in processing the JES2 SPOOL. You use the new keyword, RELADDR= with a value of ALWAYS , ASNEEDED, or NEVER to specify your choice.

Notes:

1. Relative addressing in processing the JES2 SPOOL is only used if all members of the MAS support relative addressing. Any member that is not supporting relative addressing fails to initialize and a HASP401 message results.
2. The size of a SPOOL data set is still limited to a size of 64K tracks.

The $T SPOOLDEF command is updated to allow you to update the current setting for RELADDR=. If relative addressing is being used for a volume, the $D SPOOL,UNITDATA displays a new keyword, BASETRAK=. The value of BASETRAK= must be added to the TT (part of MTTR) to convert a relative track address to an absolute track address.

A new interface is provided to read records from JES2 SPOOL. This interface is implemented as an option on the SSI 71 call. When you use this interface, you need not know about internal JES2 MTTR processing.

Migration Actions

- Understand how to use the SPOOLDEF initialization keyword, RELADDR= to specify the absolute or relative track addressing for the SPOOL.
- Understand how to use the new interface to read records from the JES2 SPOOL.
- Examine your code to ensure that the code the reads the JES2 SPOOL is consistent with this function.

Dynamic PROCLIBs Support

Customer IPL outages are aggravated by a defective JES2 PROC. JES2 z/OS V1R2 addresses this problem in "3" ways:

- Minimizes the information in the JES2 PROC (thus limiting the chance that some change in the system makes the PROC unusable).
- Provides the ability to start JES2 with a generic PROC, such as IEASYSAS for those cases where the JES2 PROC does become unusable. The two most frequently updated statements in the JES2 PROC are the PROCLIB data set concatenations and the JES2 initialization data sets. Now, JES2 provides initialization statements for these two data sets.
- Provides PROCLIB and INCLUDE initialization statements. The PROCLIB initialization statement allows you to define a logical DD name that the converter can use. Up to 255 data set names can be associated with each PROCLIB initialization statement. The INCLUDE initialization statement gives you the ability to "include" additional initialization streams into your initialization process.

Additionally, the following commands provide additional flexibility in specifying PROCLIB data sets:

- $ADD PROCLIB creates a new dynamic PROCLIB concatenation.
- $DEL PROCLIB deletes an existing dynamic PROCLIB. This command does not affect the static PROCLIBs defined in the JES2 PROC.
- $T PROCLIB updates an existing dynamic PROCLIB
- $D PROCLIB displays the current dynamic PROCLIBs only and does not display the static ones. This command has a DEBUG option that provides additional information about the dynamic PROCLIB concatenation as well as logically deleted dynamic PROCLIBs that are still in use.

Migration Actions

- Review your JES2 PROC.
- Review/Update your JES2 initialization statements appropriately.
- Take advantage of the PROCLIB and INCLUDE initialization statements to create dynamic PROCLIB concatenations and "include" initialization streams into your JES2 startup.

JES2 Support of a Maximum of 999,999 Job Numbers

With **z2 Mode**, you will be able to define up to 999,999 job numbers. Currently, the limit is just 65,534. In order to support the larger job numbers, the jobid format is changed. The format used depends on the upper limit on the RANGE operand of the JOBDEF initialization statement.

- If the upper limit is > 99,999, then the jobid is Jxxxxxxx
- If the upper limit is <= 99,999 then the jobid is JOBxxxxx
- If the upper limit is > 99,999, then the jobid is Txxxxxxx
- If the upper limit is <= 99,999 then the jobid is TSUxxxxx

- If the upper limit is > 99,999, then the jobid is Sxxxxxxx
- If the upper limit is <= 99,999 then the jobid is STCxxxxx

Note: If you are in **z2 Mode** and RASSIGN=YES is specified (or defaulted) on the JOBDEF initialization statement, the NJE jobs could use the Jxxxxxxx form of jobid if the job number is > 99,999; this occurs regardless of the RANGE setting.

Migration Actions

ATTENTION:

You should review the general description of activating your JES2 MAS to **z2 Mode** as well as the fields, field names, and control block changes that occur when running in **z2 Mode**. Reread "z2 Mode for $ACTIVATE" on page 17 and "General Field and Control Block Changes" on page 19 for the details.

- Review how your installation processes jobs/job ids. Ensure your automation adapts to potential jobids up to 999,999.
- Have ready back out procedures.
- Adjust your JOBDEF initialization parameters accordingly.
- Consider setting RASSIGN=NO.

Spinning JESlog Data Sets for Long Running Jobs

You initialize JES2 to allow the installation to "spin" or "suppress" JESlog data sets as the job runs by using the JESLOG= keyword on the JOBCLASS initialization statement.

Migration Actions
- Examine your usage of JESMSGLG and JESYSMSG data sets.
- Review initialization parameters (JOBCLASS) and JOB JCL.
- Makes changes you desire for "spinning" or "suppressing" JESlog message output.
- Review/Update automation to decide whether to use $T Jxxxx,SPIN.
- If you use Spool Data Set Browse, examine your use and be aware of new function.

Checkpoint Improvements

Prior to JES2 z/OS V1R2, switching between checkpoint modes required an all member warm start. Now installations can switch checkpoint modes using the $T CKPTDEF,MODE=DUAL|DUPLEX command. An all member warm start is not required. Changing to DUPLEX mode is always allowed. Changing to DUAL mode is NOT allowed when either CKPTn or NEWCKPTn is on the coupling facility or if DUPLEX=OFF is specified on the CKPTDEF initialization statement and both CKPTs are in use.

Note: You can use the $T CKPTDEF,MODE=DUAL|DUPLEX command even though some MAS members are not at the JES2 z/OS V1R2 release.

Migration Actions
- Update your procedures that change the MODE.

Operations

$ACTIVATE

Before warm starting JES2 z/OS V1R2, ensure a $ACTIVATE has been performed on JES2 OS/390 V2R8. To do this, issue a $D ACTIVATE command.

$SJ Command

In migrating from JES2 OS/390 V2R4 to JES2 z/OS V1R1, the behavior of the $SJ command changes if you have not installed the PTF to fix APAR OW38962.

- Prior to OW38962, the $SJ command only considered members of the JES2 MAS that were eligible to run a job, but only if (1) the member of the MAS did NOT have a member draining situation (such as, when a $P is entered) and (2) the member did NOT have an initiator draining situation (such as, when a $P XEQ is entered).

- After the fix to APAR OW38962 is installed, members of the MAS that are in a member draining situation are still NOT considered. JES2 ignores the initiator draining situation, but only on the MAS member where the $SJ command is entered.

To illustrate this change in processing, suppose there are three members active in the JES2 MAS: SY1, SY2, and SY3. Furthermore, suppose that a $P XEQ is issued on SY1 and SY2, but NOT on SY3. In this situation, when a $SJ command is issued on SY3, SY3 is the only member that the job can run on, and then only if the affinity and SCHENV status permits it to run. If the $SJ command is issued on SY2, then both SY2 and SY3 are considered for running the job.

$T SPOOL(nnnn) Command

$T SPOOL(nnnn) is a new JES2 command. This commands allows the installation to assign its systems to spool volumes. This command has one valid keyword operand, SYSAFF=. The SYSAFF= keyword has a similar function SYSAFF on the $TINTRDR command. The value you specify for SYSAFF= on the $TSPOOL(nnnn) command is also added to the output produced by the $D SPOOL command.

Migration Actions

A new RACF authorization profile is required for the $T SPOOL command. The profile is JESx.MODIFY.SPOOL with ACCESS=CONTROL.

$ZAPJOB Command

$ZAPJOB removes all traces of a job structure from the JES2 job queue. You must ensure that the job and any output it may have created is not active in any JES2 process or active on a device. Zapping a job that is active can lead to ABENDs and the loss of a PCE until JES2 is restarted.

There is nothing to stop entering a $ZAPJOB command in the JES2 initialization stream, however, IBM recommends that the ZAPJOB initialization statement be used instead. A ZAPJOB command in the initialization stream will run after warm start processing completes whereas a ZAPJOB initialization statement will be processed before warm start processing begins, before verifying the job queue.

$HASP250 now indicates when a job has been zapped.

Migration Actions

A new RACF authorization profile is required for the $ZAPJOB command. The profile is JESx.ZAP.JOB with ACCESS=CONTROL.

$HASP250 has been changed to add that a job has been zapped.

$HASP834 Deleted

$HASP834 message has been deleted.

Migration Actions

Examine your installations message automation. There is no replacement for $HASP834.

Z2 Mode

Use the JOBDEF initialization statement to increase the amount of job numbers you can have in your installation. In **z2 Mode**, JES2 allows you to define up to 999,999 job numbers (expanding the limit from 65534).

With **z2 Mode**, you will be able to define up to 999,999 job numbers. Currently, the limit is just 65,534. In order to support the larger job numbers, the jobid format is changed. The format used depends on the upper limit on the RANGE operand of the JOBDEF initialization statement.

- If the upper limit is > 99,999, then the jobid is Jxxxxxxx
- If the upper limit is <= 99,999 then the jobid is JOBxxxxx
- If the upper limit is > 99,999, then the jobid is Txxxxxxx
- If the upper limit is <= 99,999 then the jobid is TSUxxxxx
- If the upper limit is > 99,999, then the jobid is Sxxxxxxx
- If the upper limit is <= 99,999 then the jobid is STCxxxxx

Note:

JES2 uses the appropriate jobid format in displays of job information. There is a transition period after JOBDEF RANGE= is increased above 99,999 where the "old" format jobids may still be displayed. This can occur for jobs that were created when the old range was being used. Until these jobs are purged from the system, the old format for jobids may be displayed as part of informational messages about those specific jobs.

See *z/OS JES2 Initialization and Tuning Reference* for the specific changes to the JOBDEF initialization statement

Operator commands that process jobs by job number are improved to handle the increase of job numbers.

$T JOBDEF,RANGE= is updated to allow you to set a job number limit up to a maximum of 999,999. Also, the following $T command changes allow you to take advantage of new maximum for the number of jobs, number of JOEs and number of BERTs as well as to decrease their settings:

- $T JOBDEF JOBNUM= for the number of jobs
- $T OUTDEF JOENUM= for the number of JOEs
- $T CKPTSPACE BERTNUM= for the number of BERTs

JOBID Formats

The format of job numbers being displayed as part of command responses or messages has changed in z2 mode. Currently job numbers are displayed as JOBnnnnn. Five characters is not large enough to display job numbers greater than 99,999. Therefore, job numbers will be displayed differently when job numbers greater than 99,999 are used or potentially used.

The "new" job number format will be as follows: If the maximum allowed job number (high value of the JOBDEF RANGE= statement) is above 99,999 then the job number format will be J0nnnnnn. This format will always be used unless the job number range is decreased below 100,000. Similarly, STCnnnnn will become S0nnnnnn and TSUnnnnn will become T0nnnnnn.

If the job number range is increased above 99,999 or decreased below 100,000 with the $T command, a transition period will exist where the old job number format and the new job number format will be displayed as part of commands, messages, and SMF records.

Rules for determining job number format:

- If the local maximum is > 99,999 then use the new format.
- If the local maximum is <= 99,999 then use the old format as long as the job number being displayed is<= 99,999.

Note: If you are in **z2 Mode** and `RASSIGN=YES` is specified (or defaulted) on the JOBDEF initialization statement, the NJE jobs could use the Jxxxxxxx form of jobid if the job number is > 99,999; this occurs regardless of the `RANGE` setting.

Migration Actions

- Reread "z2 Mode for $ACTIVATE" on page 17 to understand how to use the `$ACTIVATE` command to bring your JES2 to **z2 Mode** and what are the changes to JES2 job number processing.
- Examine and change your automation with respect to commands, messages, and job numbers.
- Implement appropriate procedures to return to **R4 Mode**.

Spinning JESlog Data Sets for Long Running Jobs

Spinning or suppressing JESlog data described in "Spinning JESlog Data Sets for Long Running Jobs" on page 27. Refer to this section for an explanation of this function. A description of migration actions follow.

Migration Actions

- Examine your JOBCLASS(v) initialization statements and change as appropriate to use this function.
- Your installation workload of jobs might require the use of JESLOG.

Attention:

- Jobs that are converted on a JES2 z/OS V1R2 MAS member but run on a pre-JES2 z/OS V1R2 MAS member cannot perform the JESLOG spin off.
- Jobs that are converted on a JES2 z/OS V1R1 (or earlier) MAS member cannot perform JESLOG spin off regardless of which MAS member they run on.
- The converter (and START commands) cannot recognize the JESLOG in a JES2 z/OS V1R1 (or earlier) MAS member. This condition produces an error.

Examine how your installation responds to the following messages:

- HASP138
- HASP860
- HASP130

See *z/OS JES2 Messages* for details.

Dynamic JES2 PROCLIB Commands

JES2 z/OS V1R2 provides a PROCLIB initialization statement you use to define a logical DD name that the converter can use. Up to 255 data set names can be associated with the PROCLIB initialization statement. Dedicated PROCLIB data set concatenations can be eliminated.

Migration Actions

- Use the PROCLIB initialization statements in your initialization stream.
- Understand the following commands:.
  - $ADD PROCLIB creates a new dynamic PROCLIB concatenation.
  - $DEL PROCLIB deletes an existing dynamic PROCLIB. This command does not affect static PROCLIBs defined in the JES2 PROC.
  - $T PROCLIB updates an existing dynamic PROCLIB
  - $D PROCLIB displays the current dynamic PROCLIBs only and does not display the static ones. This command has a DEBUG option that provides additional information about the dynamic PROCLIB concatenation as well as logically deleted dynamic PROCLIBs that are still in use.
- This function can operate independent of mode (**z2 Mode** or **R4 Mode**).

See *z/OS JES2 Initialization and Tuning Reference* and *z/OS JES2 Commands* for more details.

Large Volume Support

JES2 supports both a relative and absolute track addressing for SPOOL. The SPOOLDEF initialization statement is updated to allow you to specify whether you **always**, **as needed**, or **never** want to use relative MTTRs in processing the JES2 SPOOL. You use the new keyword, RELADDR= with a value of ALWAYS , ASNEEDED, or NEVER to specify your choice.

Notes:

1. Relative addressing in processing the JES2 SPOOL is only used if all members of the MAS support relative addressing. Any member that is not supporting relative addressing fails to initialize and a HASP401 message results.
2. The size of a SPOOL data set is still limited to 64K tracks.

Migration Actions

- Examine the `RELADDR=` on the SPOOLDEF initialization statement to ensure it is appropriate for the volume. Make adjustments as necessary.
- Ensure that all members of the MAS use relative addressing.
- This support works in **z2 Mode** or **R4 Mode**.

Checkpoint Improvements

Installations can switch checkpoint modes using the `$T CKPTDEF,MODE=DUAL|DUPLEX` command. An all member warm start is not required. Changing to `DUPLEX` mode is always allowed. Changing to `DUAL` mode is NOT allowed when either CKPTn or NEWCKPTn is on the coupling facility or if DUPLEX=OFF is specified on the CKPTDEF initialization statement and both CKPTs are in use.

Note: You can use the `$T CKPTDEF,MODE=DUAL|DUPLEX` command even through some MAS members are not at the JES2 z/OS V1R2 level.

Migration Actions

- Update your procedures that change the `MODE`.

HASP443 Message

The HASP443 message clarifies the reason for a SPOOL allocation failure.

Migration Actions

- Examine any automation that processes this message.

Command Syntax Migration Aid Removal

In JES2 OS/390 V2R4, JES2 provided a command syntax compatibility and migration aid in the form of an automatically installed exit 5 routine. The migration aid was intended as a temporary bridge to give customers time to adapt to the command syntax changes made in OS/390 JES2 Version 2 Release 4. As of JES2 z/OS V1R2, the migration aid is no longer automatically installed and enabled. If you are not using the command syntax documented in *z/OS JES2 Commands* and are still relying on the migration aid, you can experience erroneous system operation or receive HASP649, HASP650, HASP003 messages.

The following table lists the commands translated and the results of that translation.

Table 21. Command Translation Formats

JES2 OS/390 V2R4 Format	JES2 z/OS V1R2 Format
`$D'name',...`	`$DJOBQ'name',CMDAUTH=*,...`
`$T'name',...`	`$TJOBQ'name',...` **(See NOTE 1)**
`$DJ1,2,...`	`$DJ(1,2),...` **(See NOTES 2,3)**
`$DJ1-2,J3-4,...`	`$DJ(1-2,3-4),...` **(See NOTE 3)**
`$LJnnn,ALL`	`$DOJnnn`
`$LJnnn,H`	`$DOJnnn,HELD`
`$LJnnn,READY`	`$DOJnnn,READY`
`$LJnnn,OUTGRP=xxx`	`$DOJnnn,OUTGRP=xxx`
`$CJnnn,OUTGRP=xxx`	`$COJnn,OUTGRP=xxx`
`$PJnnn,OUTGRP=xxx`	`$POJnnn,OUTGRP=xxx` **(See NOTE 4)**

Table 21. Command Translation Formats (continued)

JES2 OS/390 V2R4 Format	JES2 z/OS V1R2 Format
$PJnnn,Q=x	$POJnnn,Q=x
$vJnnn,A=|Days=|Hours=	$vJnnn,A>|Days>|Hours>
$TJnnn,S=sid1,sid2,...	$TJnnn,S=(sid1,sid2,...)
$DSPL,JOBS=nn	$DJOBQ,SPOOL=(PERCENT>=nn)
$SSPL,V=xxxxxx,...	$SSPL(xxxxxx),...
$vIxx	$vI(xx)
$vIxx-yy	$vI(xx) **(See NOTE 5)**
$vJxxxx,Txxxx...	$vJOBQ(xxxx,xxxx) **(See NOTE 6)**
$TIxx,class-list	$TI(xx),C=class-list
$HQ,ALL	$TJOBCLASS(*),QHELD=Y
$HQ,C=xyz	$TJOBCLASS(x,y,z),QHEL=Y
$AQ,ALL	$TJOBLCASS(*),QHELD=N
$AQ,C=xyz	$TJOBCLASS(x,y,z),QHELD=N
$PQ,ALL,...	$POJOBQ,READY,...
$PQ,Q=xyz	$POJOBQ,READY,Q=xyz
$OQ,ALL,...	$OJOBQ,/R=LOCAL.*,...
$OQ,Q=xyz,...	$OJOBQ,/R=LOCAL.*,/Q=xyz,...
$TALL,sid1,sid2,...	$TJOBQ(*),/S=(sid1,sid2,...)
$LSYS	$DMEMBER
$ESYS,sid	$EMEMBER(sid)
$ESYS,RESET=sid	$ECKPTLOCK,HELDBY=sid
$TSYS,IND=Y|N	$TMEMBER(local),IND=Y|N

Notes:

1. Changes are similar for $A, $C, $E, $H, $L, $O, $P, $T, $TO
2. J can be J, JOB, S, STC, T, TSU
3. Changes are similar for $A, $C, $E, $H, $L, $O, $P, $TO
4. Unless Q= is a valid job queue (XEQ, PPU, etc.)
5. Where v is one of the command verbs D, P, S, T, or Z. This command processed a named range of initiators. You must now issue these commands as separate commands.
6. Where v is one of the commands verbs A, C, E, H, L, O, P, T, TO. This command processed a list of JOBs, TSO users, and STCs. You must now issue these commands either as multiple commands ($v Jxxxx... and $v Txxxx...) or use $v JOBQ(xxxx,xxxx). The JOBQ version has slightly different function (does not check job type).

Migration Actions

If you are exclusively using the command syntax documented in the *z/OS JES2 Commands* publication and are not relying on the command migration aid exit, then no action is required. If you do not know whether or not you are currently relying on the command migration aid exit, you need to do the following:

1. First determine if the migration exit is installed. Issue $DEXIT5 on your current production systems and if the command response does not include routine name HASX5CTR, then no further action is required.
2. If the exit is installed, look through SYSLOG for $HASP006 messages which display translated commands. The fact that commands are translated does not

necessarily indicate dependence on the migration aid exit because it conservatively translates several command formats that already have valid syntax. Commands with verbs with a single subscript or subscript range that are enclosed in parentheses are frequently unnecessarily translated as shown in the following examples:

```
$HASP006 $DJ5   TRANSLATED BY EXIT HASX05C TO $DJ(5)
$HASP006 $DJ5-6 TRANSLATED BY EXIT HASX05C TO $DJ(5-6)
```

If however, you find a $HASP006 message in SYSLOG with any other form, or in the above form but having additional operands that are altered, then you are currently dependent on the migration aid exit. If you find you are currently dependent on the migration aid exit, then determine the origin of the obsolete command syntax and take corrective action. Either:

- Have your operators begin using only the supported command syntax and convert your automation scripts to use valid syntax. Refer to the *z/OS JES2 Commands* for correct syntax of JES2 commands. IBM recommends you begin this task immediately on your current production systems and analyze $HASP006 messages in SYSLOG to measure your progress.

  OR

- Although IBM strongly recommends against it, install the command migration aid exit yourself and assume the risks described in the **CAUTION** below. As of HJE7705, the command migration aid exit is now shipped as member HASX05C in SYS1.SHASSAMP.

CAUTION !

The JES2 command migration aid, exit 5 routine HASX5CTR, is now shipped in member HASX05C of SYS1.SHASSAMP and is no longer supported. This exit is not warranted and is functionally stabilized. You are strongly urged to start eliminating your dependency on this exit now and stop using the exit as soon as possible. Failure to make an orderly transition now could result in a prematurely rushed transition later.

Application Development

APPLCOPY No Longer Supported

APPLCOPY is no longer supported.

APPLCOPY is a way to access data in the checkpoint, but is replaced with checkpoint versions. (SSI 71) applications that used APPLCOPY (such as SDSF) have migrated to now use checkpoint versions. Becuase of installation overhead concerns, especially with very large checkpoint data sets, support for APPLCOPY has been discontinued. The CKPTDEF APPLCOPY= keyword is dropped and using it will generate an error during initialization (causing other operands on CKPTDEF to be ignored).

Migration Actions
- Installations should delete references to APPLCOPY in their initialization decks.
- Installations that have an application that still uses APPLCOPY, need to have that application migrate to the use of checkpoint versions before installing JES2 z/OS V1R2.

Problem Determination and Diagnosis

Provide for JES2 IPCS Support

> **Attention!**
> - You must be proactive to install JES2 IPCS.
> - Make sure JES2 IPCS support works before you experience an emergency.
> - Make sure you have JES2 IPCS support for all combinations of JES2 and MVS releases in production. (*z/OS JES2 Migration* and *z/OS JES2 Diagnosis* provide information about JES2 IPCS support.)

- When JES2 dumps are being diagnosed on the same level of the system as the system on which the dumps were taken, do the following:
  - Ensure that SHASPARM is specified in the PARMLIB concatenation.
  - Ensure that SHASMIG is specified in the STEPLIB concatenation.
  - Ensure that SHASPNL0 is specified in the ISPPLIB concatenation.

  This ensures that IPCS can find the SMP/E-maintained copy of JES2 data.
- If the JES2 levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, but the level of MVS is the same, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also specify your PARMLIB data set, plus any other data sets containing IPCS parameters that you normally need when using IPCS.
  - Specify the SHASMIG library that corresponds to the JES2 level in the dump in the first position in the STEPLIB concatenation.
  - Specify the SHASPNL0 library that corresponds to the JES2 level in the dump in the first position in the ISPPLIB concatenation.
- If both the JES2 and system levels differ between the system on which the dump was taken and the system on which the dump will be examined using IPCS, do the following:
  - Add an IPCSPARM DD statement to your logon proc, and specify the SHASPARM library that corresponds to the JES2 level in the dump in the first position in the IPCSPARM DD concatenation. Also, specify the PARMLIB data set, plus any other data sets containing IPCS parameters that correspond to the system level in the dump.
  - Specify the SHASMIG and MIGLIB libraries that correspond to the system in the dump in the first and second positions in the STEPLIB concatenation.
  - Specify the SHASPNL0 and SBLSPNL0 libraries that correspond to the system in the dump in the first position in the ISPPLIB concatenation.

Note: IPCS requires libraries in addition to the libraries that JES2 uses. You may need to make other concatenations. For more information, see *z/OS MVS IPCS Customization*.

If dump analysis leads into data structures for other elements or products, you might need to add more data sets to the concatenations to complete the diagnosis. Refer to the appropriate product publications for information about establishing the correct IPCS environments for diagnosing other elements and products.

New Record for Trace ID 17

Trace ID 17 has been updated to contain new measurements (total PCE wait time, number of $CKPTs issued, MVS wait time, $QSUSE time) that JES2 gathers during different phases of checkpoint processing. A new record 3 has been added to the READ2, PRIMARY WRITE, IMMEDIATE WRITE and FINAL WRITE checkpoint cycles that contains these measurements.

Migration Actions

If you use a program that analyses Trace ID 17 records, it may need to be updated to incorporate this support.

$DPCE Command

The $DPCE command is enhanced to provide the following additional diagnostic information:

- Ended PCE counts - $DPCE,ENDED>0 displays the PCE types that have ended.
- Active PCE counts - $PDCE,ACTIVE>0 displays the PCE types that have active work.
- Detailed PCE information - $PCDE (CNVT) displays current detailed information for each converter PCE. This information includes: where the PCE is waiting and what it is waiting for, active and outstanding I/O counts, the exit the PCE is in, the current job the PCE is processing, the time the PCE waited, and whether the PCE has ended.

Migration Actions

Be aware of the additional diagnostic information provided by the $DPCE command.

Message HASP291

Message HASP291 is enhanced in this release to include information from the MVS IEDB. This information includes the IOS completion code and the full device sense data. Two completion codes are presented because IOS combines a multiple of its completion codes into one for use by EXCP callers. In some cases, the IOS completion code will more accurately describe the error.

Migration Actions

Be aware of the additional diagnostic information provided by the HASP291 message.

Message HASP064

The old I/O error message, HASP094, was incomplete and difficult to read. A new message, HASP064, is added to this release to provide additional data in a format that is easier to understand. The HASP064 message displays the same information as the HASP094 message, but in a format similar to the HASP291 message. In addition, data from the MVS IEDB is added to the HASP064 message and includes the IOS completion code and the device sense data. Two completion codes are presented because IOS combines a multiple of its completion codes into one for use by EXCP callers. In some cases, the IOS completion code will more accurately describe the error. Also included is the failing MTTR/BBCCHHR.

Migration Actions

Be aware of the format and diagnostic information provided by the HASP064 message.

$CBIO Services - Additional Error Recording

Additional error recording is added for the $CBIO services in the USER environment. CBIO errors will now generate a SYMREC similar to the one generated in the main task for control block I/O errors. In addition, the JES2 trace ID 3, previously from the FSS environment $BUFIO service, is moved to the USER environment $CBIO service. These changes provide additional data to help diagnose errors with SPOOL control blocks.

Migration Actions

Be aware of the additional error recording provided by $CBIO services.

Dumping all Members of the MAS

z/OS V1R1 JES2 dumps all members of the MAS when certain JES2 work queue errors occur. This is referred to as "multi-system dumps".

Migration Actions

You need to make sure you have the procedures in place to collect all the dumps from all the systems.

HASP443 Message

The HASP443 message is updated to better clarify the reason a SPOOL allocation failed. One or more "reason code" can be displayed with this message to further clarify this conditon.

Migration Actions

- Review how you examine the HASP443 message.
- If you have automation that interprets HASP443, then you might need to make changes.

Auto Reply to Termination WTORs

Automatic responses to HASP070, HASP089, and HASP098 occur if there is no operator response for 10 minutes. See *z/OS JES2 Messages* for message, HASP065 for more detail.

JES2 Installation Modifications

Fiber Channel Support (FICON)

JES2 I/O processing for spool and checkpoint has been updated to take advantage of the changes made in FICON (Fiber channels).

Migration Actions

Be aware of the updates to JES2 I/O processing for spool and checkpoint in respect to FICON.

Active DCTs

Active DCTs are now chained out of the HCT ($NJEADCT, $OFFADCT, $LCLADCT).

Note: If an error occurs in the DCT chain, a SYSOUT device may not be selected for work. Such might be the case when $S command is issued. JES2 rebuilds this chain should errors occur.

Migration Actions

Installations that manipulate DCTs need to be aware of these changes.

Modification Considerations

If you plan modifications to JES2 beyond using exit routines, your migration actions will be similar to those for exit routines. See "z/OS JES2 Version 1 Release 2 Overview" on page 17 and review each item in JES2 z/OS V1R2. The migration actions enumerated for these items will apply to your own special modifications.

Distribution Library Name Changes

Previous library data set names included version, modification, and release values (for example, SYS1.V2R10M0.SHASLINK. In JES2 z/OS V1R2, the *"vrm"* qualifier (which in the example is V2R10M0) is eliminated (For example, SYS1.SHASLINK is the proper name.).

Migration Actions

Change your installations JCL appropriately, including TSO and LOGON PROCs for IPCS.

Counting Null Records

Prior to JES2 z/OS V1R2, null records written to SYSOUT data sets were padded to a single blank record. Because of this padding, these records were counted as records in the job totals. This count is not consistent with SYSOUT received over NJE. Null records from NJE are not padded and not counted in the job totals.

Migration Actions

Since the processing to pad null SYSOUT records was deleted in JES2 z/OS V1R2, counting may change record counts for existing jobs. You would notice this change in a mixed MAS (JES2 z/OS V1R2 running with earlier levels of JES2).

Chapter 9. Version Summary Tables

This chapter provides an overview of z/OS JES2 changes. It summarizes interfaces that are new, changed, or deleted for JES2 z/OS V1R4.

The following tables describe what was added, deleted, or enhanced in each release for which migration to JES2 z/OS V1R4 applies.

Summary of Changed Initialization Statements

The following table describes changes to the JES2 initialization statements for each version/release. See *z/OS JES2 Initialization and Tuning Reference* for additional information.

Initialization Statement	Release	Short Description of Change
BUFDEF	OS/390 Release 7	Changed function: • Formula to calculate BELOWBUF (below the line) and EXTBUF (above the line) limits have been modified.
CKPTDEF	z/OS V1R2	Changed function: • MODE= can be altered with an operator command.
CKPTSPACE	z/OS V1R2	Parameter change: • BERTNUM= specification change
D LOADMOD	OS/390 Release 7	New parameter: • TABles displays any DYNAMIC tables which are defined in this module, along with the type of table.
D MODULE	OS/390 Release 7	New parameter: • TABles displays any DYNAMIC tables which are defined in this module, along with the type of table.
	APAR OW41855 (integrated in z/OS V1R1)	New parameter: • LASTPTF= allows you to display the last PTF applied to a module.
D OPTsdef	OS/390 Release 7	New parameter: • UNACTIVATE displays whether or not the UNACTIVATE START PARM was specified.
DEBUG	OS/390 Release 7	New parameter: • BERT specifies whether (Yes) or not (No) JES2 provides error code $BR3 to inform the operator that a data error has been detected in a chain of BERTs.
INCLUDE	z/OS V1R2	New Statement • Allows a new initialization data set to be processed in the initialization stream.
	z/OS V1R4	New Operand • MEMBER= Includes another member from the data set we are using. • PARMLIB_MEMBER= Includes a member from the logical PARMLIB data set.

Changed Initialization Statements

Initialization Statement	Release	Short Description of Change
INITDEF	z/OS V1R2	Clarification • The default PARTNUM=3 is clarified.
JOBCLASS	z/OS V1R2	Parameter changes • New keyword, JESLOG=, is used to specify the jobclass associated with "long running" jobs. The JESlog data sets associated with these jobs can be spun while the job runs. • COMMAND=EXECUTE and COMMAND=VERIFY are clarified.
JOBDEF	OS/390 Release 7	New parameter: • CNVT_ENQ= specifies how the system should process jobs that request JCLLIB data sets that are already enqueued at job conversion.
	z/OS V1R4	New Parameter • BAD_JOBNAME_CHAR= Specifies the character to use in place of invalid characters in a jobname.
MASDEF	OS/390 Release 7	Changed parameter: • RESTART specifies whether (Yes) or not (No) a JES2 MAS member can make the jobs on a failed member eligible for restart.
OFF(n).JR	z/OS V1R1	New parameter: • SCHENV= allows you to select work to reload based on scheduling environment. • SRVCLASS= allows you to select work to reload based on service class.
OFFLOAD	z/OS V1R4	New Parameter • TRACE= Specifies whether or not trace records are to be cut for data read or written by this offload device.
PROCLIB	z/OS V1R2	New Statement • Defines a dynamic PROCLIB concatenantion.
PRT(nnn)	z/OS V1R4	New Parameter • HONORTRC= Specifies whether or not the TRC=YES|NO parameter specified on the job's OUTPUT JCL statement will be honored or not honored for a given JES mode printer.
REQJOBID	z/OS V1R2	New Statement • Describes attributes to be assigned to Request Jobid address spaces.
Rn.PRT(nnn)	z/OS V1R4	New parameters: • HONORTRC= Specifies whether or not the TRC=YES|NO parameter specified on the job's OUTPUT JCL statement will be honored or not honored for a given JES mode printer. Changed parameter: • PLIM= Two notes added.

Initialization Statement	Release	Short Description of Change
SPOOLDEF	OS/390 Release 8	Changed parameter: • The TGBPERVL parameter is set to 255 or whatever will fit based on the number of spool volumes
	z/OS V1R1	Changed parameters: • The FENCE= keyword parameter is changed in how you indicate fencing and includes a VOLUMES= keyword as follows: `SPOOLDEF.....FENCE=(ACTIVE=YES|NO,VOLUMES=nnnn)......` Obsoleted parameter: • TGBPERVL should no longer be used; its need is superseded by code that automatically balances the SPOOL cache (BLOB) size.
	z/OS V1R2	New Keyword: • The RELADDR= keyword is used to specify whether or not relative addressing is to be used to process the JES SPOOL.
TPDEF	OS/390 Release 7	Changed parameters: The EXTBUF= and BELOWBUF= parameters have been changed to SNABUF= and BSCBUF= respectively. EXTBUF= and BELOWBUF= names are obsolete; however, the names have been retained as aliases for compatibility.
	z/OS V1R1	Changed parameters: The maximum value for parameter SNABUF=(SIZE=) is increased to 32512.
TRACEDEF	z/OS V1R1	Changed parameter: • PAGES= minimum value is increased from 1 to 9.
ZAPJOB	z/OS V1R1	New initialization statement. • Used to remove all traces of a job structure from the JES2 job queue. This statement is intended for situations where a job cannot be removed using normal JES2 commands or by a JES2 restart.

Summary of Changed Commands

The following table describes changes to the JES2 operator commands for each version/release. See *z/OS JES2 Commands* for additional information.

Command	Release	Short Description of Change
$ACTIVATE	z/OS V1R2	Changed function: • z2 MODE.
	OS/390 Release 7	Changed function: • $ACTIVATE expands the JES2 checkpoint to enable functions. Once the checkpoint has been expanded and the new function enabled, it is now possible to undo the effects of the $ACTIVATE command with a JES2 cold start at a prior release level, or by using the UNACTIVATE start option during a JES2 all-member restart.
	z/OS V1R1	Deleted Function • Inactive function. Activation of function is assumed to have occurred from prior JES2 release (Release 4 through Release 8)

Changed Commands

Command	Release	Short Description of Change
$ADD LINE(nnnn)	z/OS V1R2	New parameters: • JRNUM specifies network job receivers to be assoiated with the line. • JTNUM specifies network job transmitters to be associated with the line. • SRNUM specifies network SYSOUT receivers to be associated with the line. • STNUM specifies network SYSOUT transmitters to be associated witht the line.
$ADD PROCLIB	z/OS V1R2	New command: • Defines a dynamic PROCLIB concatenation to be used during conversion processing.
$C O JOB	OS/390 Release 7	Changed function: • For the Dest|Routecde=dest parameter, greater than (> or >=) and less than (< or <=) filters may be used in conjunction to support route code ranges.
$D JOBDEF	OS/390 Release 7	New parameter: • CNVT_ENQ - Displays the status of the JCLLIB data set.
$D LOADMOD	OS/390 Release 7	New parameter: • TABles - Displays any DYNAMIC tables which are defined in this module, along with the type of table.
$D MASDEF	OS/390 Release 7	New parameter: • RESTART - Displays whether (Yes) or not (No) a JES2 MAS member in a sysplex can make the jobs associated with a JES2 member that resided on a failed MVS member in the same sysplex eligible for restart.
$D MODULE	OS/390 Release 7	New parameter: • TABles - Displays any DYNAMIC tables which are defined in this module, along with the type of table.
	OW41855 (incorporated in z/OS V1R1)	New parameter: • LASTPTF= allows you to display the last PTF applied to a module.
$D OPTSDEF	OS/390 Release 7	New parameter: • UNACTivate - displays whether or not the UNACTivate START parm was specified.
$D O Job	OS/390 Release 7	Changed function: • For the Dest|Routecde=dest parameter, greater than (> or >=) and less than (< or <=) filters may be used in conjunction to support route code ranges.
$D PCE	z/OS V1R2	Enhanced function: • added filters,
	OS/390 Release 7	Enhanced function: • This command now supports filters, and displays new data such as ended PCE counts, $ACTIVE counts, and individual PCE details.
$D PROCLIB	z/OS V1R2	New function: • Displays dynamic PROCLIB concatenations.

Command	Release	Short Description of Change
$D REQJOBID	z/OS V1R2	New function: • Displays the attributes to be assigned to Request Jobid address spaces.
$D SPOOLDEF	z/OS V1R1	New and changed parameters: • FENCE=(VOLUMES= and ACTIVE=) are displayed.
$D TPDEF	OS/390 Release 7	Changed parameters: • The EXTBUF= and BELOWBUF= parameters have been changed to SNABUF= and BSCBUF= respectively. EXTBUF= and BELOWBUF= names are obsolete; however, the names have been retained as aliases for compatibility.
$DEL PROCLIB	z/OS V1R2	New function: • Removes a dynamic PROCLIB concatenation.
$E job	OS/390 Release 7	New parameter: • Cancel - Indicates that the job is to be cancelled and held prior to execution. This operand is equivalent to specying a $H and $C command, in addition to the $E command.
$J D Details	z/OS V1R4	New command: • $JDDETAILS is a JES2 monitor command to display resource, sampling, and wait statistics for the JES2 address space being monitored.
$J D History	z/OS V1R4	New command: • $JDHISTORY is a JES2 monitor command to display the history of resource usage and CPU statistics over time.
$J D JES	z/OS V1R4	New command: • $JDJES is a JES2 monitor command to display the current state of the JES2 being monitored.
$J D Monitor	z/OS V1R4	New command: • $JDMONITOR is a JES2 monitor command to display information about the JES2 monitor.
$J D Status	z/OS V1R4	New command: • $JDSTATUS is a JES2 monitor command to display the current status of JES2.
$J Stop	z/OS V1R4	New command: • $JSTOP is a JES2 monitor command to terminate the JES2 monitor address space.
$O Job	OS/390 Release 7	Changed function: • For the Dest|Routecde=dest parameter, greater than (> or >=) and less than (< or <=) filters may be used in conjunction to support route code ranges.
$P O Job	OS/390 Release 7	Changed function: • For the Dest|Routecde=dest parameter, greater than (> or >=) and less than (< or <=) filters may be used in conjunction to support route code ranges.
$S Job	z/OS V1R1	Changed Command: • Modifies selection of jobs to run on a MAS member based on each memeber's "draining state" and "initiator draining state".
$S SPOOL	z/OS V1R1	New Parameter: • The SYSAFF= parameter specifies the number of volumes to be used in spool fencing.

Changed Commands

Command	Release	Short Description of Change
$T DEBUG	OS/390 Release 7	New parameter: • BERT specifies whether (Yes) or not (No) JES2 provides error code $BR3 to inform the operator that a data error has been detected in a chain of BERTs.
$T Job	OW38962 (incorporated in z/OS V1R1)	New parameter: • SCHENV - specifies to modify the scheduling environment for a job.
	z/OS V1R2	Parameter Format: • You can specify up to a maximum job number of 999,999. • SPIN parameter added.
$T JOBCLASS	z/OS V1R2	New parameter: • JESLOG= used to "spin off" or suppress JESlog data sets for long running jobs.
$T JOBDEF	OS/390 Release 7	New parameters: • CNVT_ENQ - specifies if the converter should WAIT for the unavailable data sets to become available or FAIL (with a JCL error) a job that requests an unavailable JCLLIB data set.
	OW38962 (incorporated in z/OS V1R1)	New parameter: • SCHENV specifies the default scheduling environment.
	z/OS V1R2	Specifies an increased maximum range of jobs that can be in the JES2 job queue.
$T LINE(nnnn)	z/OS V1R2	New paramters: • JTNUM specifies the number of job transmitters to be associated with the line. • JRNUM specifies the number of job receivers to be associated with the line. • SRNUM specifies the number of SYSOUT receivers to be associated with the line. • STNUM specifies the number of SYSOUT transmitters to be associated with the line.
$T O Job (continued)	OS/390 Release 7	Changed function: • For the Dest|Routecde=dest parameter, greater than (> or >=) and less than (< or <=) filters may be used in conjunction to support route code ranges.

$T OFF(n).SR	z/OS V1R1	Added parameters:
		• You can now specify the following parameters:
		  – CREATOR
		  – FORMS
		  – JOBNAME
		  – PRMODE
		  – ROUTECDE
		  – WRITER

$T OFFLOAD	z/OS V1R4	New Parameter • TRACE= Specifies whether or not trace records are to be cut for data read or written by this offload device.
$T PCE	OS/390 Release 7	Enhanced function: • This command now supports filters, and displays new data such as ended PCE counts, $ACTIVE counts, and individual PCE details.

Command	Release	Short Description of Change
$T PROCLIB	z/OS V1R2	New function: • Modifies an existing dynamic PROCLIB concatenation.
$T PRT(nnn)	z/OS V1R4	New Parameter • HONORTRC= Specifies whether or not the TRC=YES|NO parameter specified on the job's OUTPUT JCL statement will be honored or not honored for a given JES mode printer.
$T Rn.PRnnn		New parameters: • HONORTRC= Specifies whether or not the TRC=YES|NO parameter specified on the job's OUTPUT JCL statement will be honored or not honored for a given JES mode printer. Changed parameter: • PLIM= Two notes added.
$T REQJOBID	z/OS V1R2	New function: • Specify or modify Jobid address spaces.
$T SPOOL(nnnn)	z/OS V1R1	New Command: • The SYSAFF= parameter specifies the number of volumes to be used in spool fencing.
$T SPOOLDEF	z/OS V1R1	Dropped parameter: • TGBPERVL is no longer valid. Use will cause an error message.
$T TPDEF	OS/390 Release 7	Changed parameters: • The EXTBUF= and BELOWBUF= parameters have been changed to SNABUF= and BSCBUF= respectively. EXTBUF= and BELOWBUF= names are obsolete; however, the names have been retained as aliases for compatibility.
	z/OS V1R1	Changed parameters: • The SNABUF=(SIZE=) parameter value has a maximum value of 8.
$ZAPJOB	z/OS V1R1	• The ZAPJOB initialization statement is new and is used to remove all traces of a job structure from the JES2 job queue. This statement is intended for situations where a job cannot be removed using normal JES2 commands or by a JES2 restart.

Summary of Changes to Messages

> **Note to the Reader**
>
> In previous editions, this section listed new, changed, and deleted messages. This was intended to help migrate automated operations. Because *z/OS JES2 Messages* provides a detailed list of changes to messages in its Summary of Changes, this book will no longer describe changes to messages. Instead, see the Summary of Changes section in *z/OS JES2 Messages*.

Summary of Changed Installation Exits

The following table describes changes to the JES2 installation exits for each version/release. See *z/OS JES2 Installation Exits* for additional information.

• Exit 9 Called immediately when limit reached (not at end of buffer)
• Exit 22 New parameter list

Changed Installation Exits

Exit	Release	Short Description of Change
All Exits	All	• In addition to the specific changes above, you must reassemble all installation exits to prevent problems with incompatible macro changes. Also see the Summary of Changed Macros table for changes to executable macros used in installation exits. • JES2 z/OS V1R2 restructures many control blocks and data areas that could affect your exits. You should review all these changes in preparing for your migration
Exit 5	z/OS V1R2	Function change: • The command migration aid exit 5 routine HASX5CTR in module HASX05C is no longer automatically installed or enabled. The exit is shipped in member HASX05C in SYS1.SHASSAMP.
Exit 8	OS/390 Release 7	Function change: • This Exit can now receive control for control block I/O ($CBIO) calls from the FSS address space.
Exit 9	z/OS V1R4	Function Change: • Called immediately when limit reached (not at end of buffer)
Exit 11	z/OS V1R1	Function change: • Installation exit 11 is unaffected if you use no new functions in z/OS Version 1 Release 1 JES2. However, if you turn on spool affinity to increase the number of spool volumes used in fencing, then exit 11 is affected.
Exit 12	z/OS V1R1	Function change: • Installation exit 12 is unaffected if you use no new functions in z/OS Version 1 Release 1 JES2. However, if you turn on spool affinity to increase the number of spool volumes used in fencing, then exit 12 is affected.
Exit 14	z/OS V1R2	Function change: • An indicator is provided to this exit that controls class list optimization.
Exit 20	OS/390 Release 7	Function change: • Register contents were modified. Register 1 was added, which contains the address of a parameter list mapped by $XPL.
Exit 22	z/OS V1R4	Function Change: • New parameter list
Exit 25	OS/390 Release 7	Function change: • $BUFIO and $BUFCK services in HASPFSSM have been deleted.
Exit 46	OS/390 Release 7	Function change: • Exit will now be invoked for NJE RCCS headers.
Exit 47	OS/390 Release 7	Function change: • Exit will now be invoked for NJE RCCS headers.
Exit 49	z/OS V1R2	Function change: • An indicator is provided to this exit that controls class list optimization.

Summary of Changed Macros for Installation Exits

The following table describes changes to the JES2 macros used in installation exits. See *z/OS JES2 Macros* for additional information.

$#BUSY New Parameter #POST=YES/NO $$WTO New parameters: CONNECT=
AREAID= MID= Updated: TEXT=, to allow specification of line type. $ENVIRON
Updated parameter ENVIRON=MONITOR $EXCP Added PCB=YES/NO $TRACE
Added value TYPE=ENTER $WTO Added TEXT= CONNECT=

Macro for Installation Exits	Release	Short Description of Change
$$POST	OS/390 Release 7	Changed parameter: • XMITJOB resource added to TYPE=. A JES2 job queue element (JQE) has been placed on the $XMIT queue to be transmitted to another node.
	z/OS V1R2	Changed parameter: • New resource character specification.
	z/OS V1R4	New parameters: • ERRET= • OKRET=
$$WTO	z/OS V1R4	New parameters: • AREAID= • CONNECT= • MID= Changed parameter: • Updated TEXT= to allow specification of line type.
$#BUSY	z/OS V1R4	New parameter: • #POST=YES|NO
$#DISPRO	OS/390 Release 5	New parameter: • IOT=Address I/O table for a specific spin JOE • PQE=Print Queue element address for a specific spin JOE • ENF=Specifies whether a ENF 58 signal should be generated Default=NO
$#JOE	OS/390 Release 5	New parameter: • DSERV=specifies the address of the checkpoint version of the DSERV control block
	z/OS V1R2	Parameter changes: • Q=, GETFIRST=, JQE=, CHAR=, PTREG=, PREG= New Processing: • to use methods for the new JQE and JOE addresses for 999,999 job numbers.
$#PUT	OS/390 Release 5	New Keywords: • IOT=Address I/O table for a specific spin JOE • PQE=Print Queue element address for a specific spin JOE • ENF=Specifies whether a ENF 58 signal should be generated Default is NO
$#TJEV	z/OS V1R1	New macro. • $#TJEV macro can be used in exits to find out why output is not printing.
$BERTTAB	OS/390 Release 7	New macro: • Maps and generates Block Extention Reuse Tables (BERT) entries.
$BUFIO	OS/390 Release 7	Deleted macro: • Macro made obsolete by $CBIO.

Changed Macros

Macro for Installation Exits	Release	Short Description of Change
$BUFCK	OS/390 Release 7	Deleted macro: • Macro made obsolete by $CBIO.
$CBIO	OS/390 Release 7	New parameter: • TYPE=WAIT waits for the completion of a $CBIO request that specified WAIT=NO.
$CKPT	OS/390 Release 5	Changed macro: • Exits can no longer code $CKPT ID=JQE
$CPOOL	OS/390 Release 7	Changed parameters: • LOC=CSABELOW • LOC=BELOW,ANY • LOC=CSABELOW,ANY
$DCTTAB	OS/390 Release 7	New parameter: • TABLE=DYNAMIC specifies that this is a dynamic table.
$DEST	OS/390 Release 5	Changed macro: • Additional values can be specified for IPFORM=
$DILBERT	OS/390 Release 7	New parameter: • POST=resource • ALLOWDUP=YES|NO • PACE=NO|YES • ERRET=label|Rn • OKRET=label|Rn Changed parameters: • PARM0
	z/OS V1R2	Parameter Changes • FUNCTION=PROCESS | CHECK • Return code changes when macro exits
	z/OS V1R4	Changed parameter: • New value PCEWORK for POST=.
$DOGBERT	OS/390 Release 7	New macro. • Copies data from the Block Extension Reuse Tables (BERT) portion of the checkpoint into local storage or returns that data to the checkpoint.
$DOGJQE	OS/390 Release 7	New parameter: • POST indicates whether the return of a locked JQA should force a post of all processors potentially waiting for that JQA.
	z/OS V1R2	Changed Parameters: • ACTION=(FETCH) • GETJQXADDR
$DOGWSCQ	OS/390 Release 7	Changed parameters: • ACTION=(FETCH,CREATE)
$DTETAB	OS/390 Release 7	New parameter: • TABLE=DYNAMIC specifies that this is a dynamic table.
$ENVIRON	z/OS V1R4	Changed parameter: • New value MONITOR for ENVIRON=
$ERROR	z/OS V1R2	Parameter change • RECVOPTS=
$EXCP	z/OS V1R4	New parameter: • PCB=YES|NO

Macro for Installation Exits	Release	Short Description of Change
$FREMAIN	OS/390 Release 7	New parameter: • WAIT=YES|NO
$GETBUF	OS/390 Release 7	New parameter: • TYPE=EXTHASP states that a control block buffer is to be constructed.
$GETMAIN	OS/390 Release 7	Changed parameter: • LOC=BELOW,ANY
	z/OS V1R2	Parameter changes: • OWNER=, LOC=, ZEROSTOR=, OKRET=, ERRET=
$GETABLE	OS/390 Release 7	New parameter: • $PAIR=table-pair Changed parameters: • TABLE=BERT • TABLE=WSR • TABLE=SCAN
$JBIDBLD	z/OS V1R2	Changed parameters: • JUMREG=, JQE=
$JQEJNUM	z/OS V1R2	New macro
$MODCHK	OS/390 Release 7	New parameter: • TEST=TABLES resolves any dynamic tables in the module.
$MODULE	OS/390 Release 7	Changed parameter: • $JIBX DSECTID deleted.
$PCETAB	OS/390 Release 7	Changed parameter: • TABLE=DYNAMIC
	z/OS V1R4	New parameters: • WARMLOCK=YES|NO Specifies whether or not this PCE type is allowed to obtain a JQE warm start lock. • XWTR=YES|NO Specifies whether or not this PCE type is a PCE that services PSO or SAPI external applications. • MONITOR=JOBID|JQEINDEX Specifies whether to set the JOBID, JQEINDEX or nothing for the JES2 monitor. Controls what is set in PCEJOBID or PCEJQEIX.
$PCETERM	z/OS V1R2	Changed parameters
$POST	OS/390 Release 7	Changed parameter: • XMITJOB resource added to TYPE=. A JES2 job queue element (JQE) has been placed on the $XMIT queue to be transmitted to another node.
	z/OS V1R2	Parameter change: • New resource specification.
$QJQE	z/OS V1R2	Changed parameters: • To process new JQE and JOE addresses for 999,999 job numbers. • GETFIRST=, REG=, LOOP=, NOMRE=, PREG=
$QLOCNXT	z/OS V1R2	New macro

Changed Macros

Macro for Installation Exits	Release	Short Description of Change
$SCANTAB	OS/390 Release 7	New parameter: • TABLE=DYNAMIC specifies that this is a dynamic table. Changed parameters: • table-name,ADDR • table-name,VCON
$SUBIT	OS/390 Release 5	New Keyword • FREESQD=YES - indicates that a SDQ should be obtained by $SUBIT and freed when the subtask is complete. The ECB in the SQD will never be posted.
$TIDTAB	OS/390 Release 7	New parameter: • TABLE=DYNAMIC specifies that this is a dynamic table.
$TRACE	z/OS V1R4	Changed parameter: • New value ENTER for TYPE=, allows entrance to the tracing routine without testing the necessity to do so.
$VERIFY	OS/390 Release 5	New parameter: • MTTR=specifies a track address (MTTR) to be compared aganist the MTTR of the control block being verified.
$VERTAB	OS/390 Release 5	New parameter: • MTTRFLD=specifies the offset of the field containing a MTTR to be verified
$WAIT	OS/390 Release 7	Changed parameters: • XMITJOB resource added to TYPE=. A JES2 job queue element (JQE) has been placed on the $XMIT queue to be transmitted to another node.
	z/OS V1R2	Parameter change: • New resource specification (ALICE)
$WSTAB	OS/390 Release 7	New parameter: • TABLE=DYNAMIC specifies that this is a dynamic table.
$WTO	z/OS V1R4	New parameters: • TEXT= • CONNECT=
$XMPOST	z/OS V1R2	Parameter changes: • XMPLIST changes

Summary of Changed Mapping Macros

The following table lists intended mapping macros that are new, changed, or deleted. For a summary of new, changed and deleted mapping macros, see the following:

- *z/OS JES2 Data Areas, Vol 1 $ALINDEX-$EVT*
- *z/OS JES2 Data Areas, Vol 2 $FCLWORK-$OUTWORK*
- *z/OS JES2 Data Areas, Vol 3 $PADDR-$XRQ*

Mapping Macro	Release	New/Changed/Deleted
$BERT	OS/390 Release 7	Changed
	z/OS V1R1	Changed

Mapping Macro	Release	New/Changed/Deleted
$BERTTAB	OS/390 Release 7	Changed
$BUFFER	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$CADDR	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$CAT	z/OS V1R1	Changed
$CATBERT	z/OS V1R1	Changed
$CCW	z/OS V1R1	New
$CIRWORK	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$CK	OS/390 Release 7	Changed
	OS/390 Release 8	Changed
	z/OS V1R1	Changed
$CKGPAR	OS/390 Release 7	Changed
	OS/390 Release 8	Changed
$CKPRECV	OS/390 Release 7	Changed
	OS/390 Release 8	Changed
$CKPWORK	OS/390 Release 7	Changed
	OS/390 Release 8	Changed
	z/OS V1R1	Changed
$CKW	OS/390 Release 7	Changed
	OS/390 Release 8	Changed
$CNVWORK	z/OS V1R1	Changed
$COMWORK	z/OS V1R1	Changed
$CPINDEX	OS/390 Release 7	Changed
$CTOKEN	z/OS V1R1	New
$CTW	OS/390 Release 7	Changed
	OS/390 Release 8	Changed
$CVCB	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$DAS	z/OS V1R1	Changed
	z/OS V1R2	Changed
$DCT	OS/390 Release 5	Changed
	z/OS V1R1	Changed
$DILWORK	OS/390 Release 7	Changed
$DTEACCT	OS/390 Release 5	Changed
$DTECNV	z/OS V1R1	New
	z/OS V1R2	Changed
$DTRCKCF	OS/390 Release 8	New
$DTEOFF	OS/390 Release 7	Changed

Changed Mapping Macros

Mapping Macro	Release	New/Changed/Deleted
$DTESPL	OS/390 Release 7	Changed
$DWA	OS/390 Release 7	Changed
$ERA	z/OS V1R1	New
$ERPL	z/OS V1R1	New
$FSACB	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$FSSCB	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$FSSWORK	z/OS V1R1	New
$GTW	z/OS V1R1	New
	z/OS V1R2	Changed
$HASPEQU	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$HASPGBL	OS/390 Release 5	Changed
	z/OS V1R1	Changed
$HASXB	OS/390 Release 7	Changed
$HCCT	OS/390 Release 5	Changed
	OS/390 Release 8	Changed
	z/OS V1R1	Changed
$HCT	OS/390 Release 7	Changed
	OS/390 Release 8	Changed
	z/OS V1R1	Changed
	z/OS V1R2	Changed
$HFCT	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$ICE	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
$INIWARM	OS/390 Release 7	Changed
$JCT	OS/390 Release 7	Changed
	z/OS V1R1	Changed
	z/OS V1R2	Changed
$JCTX	OS/390 Release 5	Changed
$JESLOG	z/OS V1R2	New
$JIB	OS/390 Release 4	Changed
	OS/390 Release 7	Changed
	z/OS V1R1	Changed
	z/OS V1R2	Changed
$JIBX	OS/390 Release 7	Deleted
$JNT	OS/390 Release 1	Changed

Mapping Macro	Release	New/Changed/Deleted
$JOE	S/390 Release 5	Changed
	z/OS V1R1	Changed
$JOT	z/OS V1R2	Changed
$JQE	OS/390 Release 4	Changed
	OS/390 Release 5	Changed
	z/OS V1R1	Changed
	z/OS V1R2	Changed
$KAWA	z/OS V1R1	Changed
$MCT	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$MIT	z/OS V1R1	Changed
$MLMWORK	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
$MSCWORK	z/OS V1R2	New
$NHD		
$NJTWORK	OS/390 Release 7	Changed
$NSRWORK	z/OS V1R1	Changed
$PAD	z/OS V1R2	New
$PADDR	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
	OS/390 Release 8	Changed
	z/OS V1R1	Changed
$PARMLST	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$PARMWRK	z/OS V1R2	New
$PCE	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$PDDB	OS/390 Release 5	Changed
$PERFCB	OS/390 Release 7	Changed
	OS/390 Release 8	Changed
$PPPWORK	OS/390 Release 5	Changed
$PQE	OS/390 Release 5	Changed
$PREBERT	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$PSO	OS/390 Release 5	Changed
	z/OS V1R2	Changed
$PSOWORK	OS/390 Release 5	Changed
$PSV	OS/390 Release 7	Changed
$QSE	OS/390 Release 7	Changed

Changed Mapping Macros

Mapping Macro	Release	New/Changed/Deleted
$RAT	OS/390 Release 7	Changed
$RDRWORK	z/OS V1R1	Changed
$REQJID	z/OS V1R2	New
$RESWORK	z/OS V1R1	Changed
$RGRPLST	OS/390 Release 10	New
$ROTT	OS/390 Release 8	Changed
	z/OS V1R2	Changed
$SAPID	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$SCANTAB	OS/390 Release 5	Changed
$SCANWA	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$SDB	OS/390 Release 7	Changed
$SJB	OS/390 Release 5	Changed
	OS/390 Release 8	Changed
	z/OS V1R1	Changed
	z/OS V1R2	Changed
$SJIOB	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
$SJXB	OS/390 Release 8	Changed
	z/OS V1R1	Changed
$SPIWORK	z/OS V1R1	New
$SPMWORK	OS/390 Release 5	Changed
$SPUD	OS/390 Release 7	Changed
$SQD	OS/390 Release 5	Changed
$SWR	OS/390 Release 5	Changed
$SXADDR	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$SYMCB	OS/390 Release 7	Changed
$TGB	OS/390 Release 7	Changed
$TOT	z/OS V1R1	New
$TRCA	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$TRE	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$TRX	z/OS V1R1	Changed
$UPL	z/OS V1R1	New
$USERCBS	OS/390 Release 1	Changed
$WARMWRK	OS/390 Release 5	Changed

Mapping Macro	Release	New/Changed/Deleted
$WLMD	z/OS V1R1	Changed
$WSA	z/OS V1R1	Changed
$WSP	OS/390 Release 5	Changed
	OS/390 Release 7	Changed
	z/OS V1R1	Changed
$XEQWORK	OS/390 Release 7	Changed
$XMAS	OS/390 Release 7	Changed
$XPL	OS/390 Release 7	Changed
	OS/390 Release 7	Changed
$XREQ	z/OS V1R1	New
$ZIP	z/OS V1R1	New

Summary of Changed SSI Function Codes

The following tables identify all JES2 changes made to the SSI function codes as a result of OS/390 and z/OS JES2 releases. See *z/OS MVS Using the Subsystem Interface* for a complete explanation of SSI function codes.

Table 22. Changes to SSI Function Codes for z/OS Version 1 Release 4 JES2

Code	Short Description of Change
79	SAPI JMR information.
71	Instorage Buffer Enhancements

Changed SSI Function Codes

Appendix A. An Example of ACTIVATING Your System

The following description of ACTIVATing your system is provided as an example of migrating from JES2 OS/390 V2R8 (HJE6608) to JES2 OS/390 V2R10 (HJE7703). This description assumes an installation has a two system MAS, SY1 and SY2. The installation has been running the MAS at the JES2 OS/390 V2R8: level in pre-JES2 OS/390 V2R4 mode **Pre-R4 Mode**. The migration description focuses on ACTIVATION levels, the implications of each, and the ways to $ACTIVATE the systems.

JES2 OS/390 V2R10 no longer supports a **Pre-R4 Mode** that facilitated pre-JES2 OS/390 V2R4 release migrations. The major difference between **Pre-R4 Mode** and full JES2 OS/390 V2R4 mode **R4 Mode** is the format of the checkpoint data sets. **Pre-R4 Mode** and **R4 Mode** cannot tolerate each other because of the differing checkpoint data set formats. To set the JES2 OS/390 V2R8 system in our example from **Pre-R4 Mode** to **R4 Mode**, you issue the $ACTIVATE command on the JES2 OS/390 V2R8 system.

If you are unsure of the activation mode of your system, you can issue the $D ACTIVATE command. This command shows (with $HASP895) the current JES2 checkpoint level of that system, indicates if the checkpoint data set is large enough to hold the checkpoint, and whether the $ACTIVATE on that system will succeed or fail.

For the two systems, SY1 and SY2, you issue the $D ACTIVATE command on each system. $HASP895 tells you the activation status of each.

For example, the $D ACTIVATE on SY1 shows the following:

```
$D ACTIVATE

$HASP895 JES2 CHECKPOINT LEVEL IS NOW PRE-RELEASE 4
$HASP895 A TOTAL OF 90 4K RECORDS ARE REQUIRED FOR $ACTIVATE
$HASP895 ALL INUSE=YES ARE AVAILABE AND LARGE ENOUGH
$HASP895 $ACTIVATE WILL SUCCEED IF ISSUED FROM THIS MEMBER
```

THE SY2 system shows the following:

```
$D ACTIVATE

$HASP895 JES2 CHECKPOINT LEVEL IS NOW PRE-RELEASE 4
$HASP895 A TOTAL OF 84 4K RECORDS ARE REQUIRED FOR $ACTIVATE
$HASP895 ALL INUSE=YES ARE AVAILABE AND LARGE ENOUGH
$HASP895 $ACTIVATE WILL SUCCEED IF ISSUED FROM THIS MEMBER
```

The SY1 system is in **Pre-R4 Mode**. You need to issue the $ACTIVATE command on SY1 so that you can install your new JES2 OS/390 V2R10 (HJE7703) system.

After you have performed a $ACTIVATE on SY1, you will not have to do the same for SY2. Because SY1 and SY2 are both member of the MAS, they are sharing the same checkpoint definition. Whatever member issues the $ACTIVATE will change the checkpoint from **Pre-R2 Mode** to **R2 Mode**. Your new JES2 release, JES2 OS/390 V2R10 can use this checkpoint data set.

If you decide in the future to drop back to the JES2 OS/390 V2R8 release, it will initialize in **R2 Mode** because that is what the checkpoint dataset became when

you performed the $ACTIVATE. To initialize this "fall back" to JES2 OS/390 V2R8 and return to **Pre-R2 Mode**, you will need to warm start JES2 and specify UNACT to the startup prompt messsage ($HASP426). APAR OW32920, which introduced UNACT start parameter and the *JES2 Commands* give details about warm starting with UNACT. After you have started JES2 in this way, you will be running in **Pre-R2 Mode**.

Note: The z/OS version of APAR OW32920 also allows you to start JES2 with the UNACT start parameter; but starting JES2 with UNACT in z/OS returns your system to **R2 Mode**.

In summary remember these points:

- You must have the compatibility apar OW42299 (UW69012) installed on your OS/390 JES2 Version 2 Release 8 system.
- It is recommended that you issue the $ACTIVATE on the SY1 (or SY2) and run in **R2 Mode** for a while before installing JES2 OS/390 V2R10. This allows you to see if there are any operational or other concerns that manifest themselves in **R2 Mode**. Once you are stable, you can initialize OS/390 JES2 Version 2 Release 10.
- In **R2 Mode** for both SY1 and SY2, the JOBCLASS(v) attributes become MAS-wide, not only 'by member'.

  The JOBCLASS(v) implications will depend on how you run your system. If you already have all of the JOBCLASS(v) specifications coded the same on both SY1 and SY2, then you most likely will notice nothing of consequence. However, if JOBCLASS(v) on SY1 is different that JOBCLASS(v) on SY2, when you perform the $ACTIVATE on SY1, those JOBCLASS attributes become the same for both SY1 and SY2. As a result, complications may occur.

  For example, in **Pre-R2 Mode** for SY1, JOBCLASS(A) TIME=1440 is defined. For SY2, JOBCLASS(A) TIME=(05,00) is defined. In this **Pre-R2 Mode**, a job will be allowed to run without timing out if it runs on SY1, but will be abended after 5 minutes if it runs on SY2. After $ACTIVATE to establish **R2 Mode** for both SY1 and SY2, JOBCLASS(A) TIME= will be the same on both systems. You can make subsequent changes on SY1 and SY2 to minimize job processing disruptions by using the $D JOBCLASS(x) command to show the values that in affect for your MAS. You can then change these values with the $T JOBCLASS(v) command.

Appendix B. Sample Exits

Sample Exit Modules (unnumbered)

HASXHUBA	Routines for sample exit HASXOCA
HASXJEA	Exit 28 and exit 25 routines
HASXOCA	Exit 30 and 33 routines
HASXJECL	Exits 1, 4, 46, 47 to show $JCTX services
HASPXJ00	Sample table pairs macros (for use by HASX00B)
HASPXITO	Pre-Initialization exit (used with module HASPXJ00)
HASXDNYT	Sample dynamic table pairs

Sample Exit Modules (numbered)

HASX00A	Pre-initialization exit (creates user scan tables in $MCT)
HASX00B	Pre-initialization exit (initializes $UCT)
HASX01A	Separator page exit (old-style)
HASX01B	New-style (JES2 Ver. 4) separator page exit
HASX02A	JOB card scan exit
HASX04A	JCL card scan exit
HASX05A	Command pre-scan exit
HASX05C	Convert Pre-HJE6604 format commands
HASX06A	Converter/interpreter text scan exit
HASX06B	Routine that uses the $CALL XINTKEY subroutine
HASX07A	JES2 control block I/O exit (JES2 environment)
HASX08A	JES2 control block I/O exit (user environment)
HASX09B	Job output excession exit
HASX10A	$WTO exit
HASX13A	NetMail Receive Exit
HASX15A	Dataset separator page exit
HASX23A	FSS job separator page exit
HASX24A	Post-initialization exit
HASX27A	PCE Attach/Detach exit
	Exit 28 - Job termination - see HASXJEA
HASX29A	End of memory exit
	Exit 30 - Data set open - see HASXOCA
HASX31A	Data set allocation exit
HASX32A	Job selection exit
	Exit 33 - Data set close - see HASXOCA
HASX34A	Data set unallocation exit
	Exit 35 - End-of-task - see HASXJEA
HASX36A	Pre-SAF exit
HASX37A	Post-SAF exit
HASX38A	TSO receive (security failures - 3.1.3 only)
HASX38B	TSO receive (security failures)
HASX39A	SYSOUT receive (security failures)
HASX40A	JOE build exit
HASX41A	Output grouping exit
HASX42A	Notify message exit
HASX43A	APPC TP selection/termination
HASX44A	Converter (main task) exit
HASX45A	SJF services exit

Sample Exits

HASX46A	NJE header transmission exit
HASX47A	NJE header receive exit
HASX48A	Data set unallocation exit
HASX255A	User-defined exit invoked by HASX003A

Appendix C. Split/New/Deleted/Resequenced Parts for JES2

Split/New/Deleted/Resequenced Parts for HJE7707 (z/OS Version 1 Release 4 JES2)

New parts in z/OS V1R4:
- $DTEEOM
- $EMWORK
- $HJCT
- $MONCB
- $MSD
- $MWE
- $PBLK
- $STAC
- HASCSRJM
- HASJCMDS
- HASJMON
- HASJSPLR
- HASMACE
- HASMACT
- HASMBFD
- HASMPBLK
- HASMSTAC

Resequenced parts in z/OS V1R4:
- HASCHAM
- HASX44A

Split/New/Deleted/Resequenced Parts for HJE7705 (z/OS Version 1 Release 2 JES2)

New parts in z/OS V1R2:
- $#INO
- $#OIN
- $JESLOG
- $JQEJNUM
- $MSCWORK
- $PAD
- $PARMWRK
- $QINO
- $QLOCNXT
- $QOIN
- $REQJID
- HASLJOCS
- HASMBUF2
- HASMHEXD
- HASMCEB
- HASMTGB

Resequenced parts in z/OS V1R2:
- HASPSXIT
- HASCDAU

  • HASCSJFA

Split/New/Deleted/Resequenced Parts for HJE7703 (z/OS Version 1 Release 1 JES2)

New parts in z/OS V1R1:
  • HASMJNT
  • HASMJWEL
  • HASMPRE
  • HASMTQE

Split/New/Deleted/Resequenced Parts for HJE6608 (OS/390 Release 8 JES2)

New parts in Release 8:
  • $DTECKCR
  • HASPCKCR

Parts resequenced in Release 8:
  • $CADDR
  • $CALLP
  • $CKGPAR
  • $CKPWORK
  • $CKW
  • $CSVLIST
  • $DTE
  • $END
  • $HASPEQU
  • $HCCT
  • $HCT
  • $JCAN
  • $MODULE
  • $OFFSTBL
  • $PARMLST
  • $RDRWORK
  • $WAIT
  • HASCDSOC
  • HASCENF
  • HASCHAM
  • HASCJBTR
  • HASCPOOL
  • HASCSIRQ
  • HASCSISC
  • HASCSRJB
  • HASIBLD
  • HASMCADR
  • HASMITAB
  • HASMRDW
  • HASPCFAL
  • HASPCFLE
  • HASPCFR2
  • HASPCKDS
  • HASPCKPT
  • HASPCNVT
  • HASPCOMM

- HASPDOC
- HASPFSSM
- HASPIRDA
- HASPIRMA
- HASPJQS
- HASPMGS
- HASPNJT
- HASPNSR
- HASPNUC
- HASPPRPU
- HASPRAS
- HASPRDR
- HASPRTAM
- HASPSERV
- HASPSXDV
- HASPSXIT
- HASPTERM
- HASPTRAK
- HASPWARM
- HASPXEQ
- HASX01A
- HASX46A

Split/New/Deleted/Resequenced Parts for HJE6607 (OS/390 Release 7 JES2)

New parts in Release 7:
- $PAIR
- $PUTABLE
- $RETABLE
- HASXDYNT

Parts resequenced in Release 7:
- $BUFFER
- $ENTRY
- $GETMAIN
- $PADDR
- $PARMLST
- HASCDSS
- HASCLINK
- HASCSRDS
- HASCSRIC
- HASMICE
- HASMPADR
- HASMPREB
- HASPCKDS
- HASPCKPT
- HASPCOMM
- HASPCSV
- HASPDOC
- HASPIRDA
- HASPJQS
- HASPMSG
- HASPNJT
- HASPNPM

JES2 Parts

- HASPNUC
- HASPPSO
- HASPRAS
- HASPRTAM
- HASPSCAN
- HASPSNA
- HASPSPOL
- HASPSXDV
- HASPSXIT
- HASPSXOT
- HASPTABS
- HASPTERM
- HASPTRAK
- HASPXEQ

Parts deleted in Release 7:
- HASCUBES
- HASCUBFR
- HASCUBRM
- $BUFIO
- $BUFCK
- $JIBX
- HASMJIBX

Split/New/Deleted/Resequenced Parts for HJE6605 (OS/390 Release 5 JES2)

New parts in Release 5:
- HASMFTEX
- HASMIPSV
- HASMPERF
- HASMWTLB
- HASMXECB

Parts resequenced in Release 5:
- HASCDASL
- HASCHAM
- HASCPOOL
- HASCSAPI
- HASCSIRQ
- HASCSISC
- HASCSJFS
- HASCSRDS
- HASCSRJB
- HASMFMTM
- HASMICE
- HASMPADR
- HASMPCE9
- HASPARM
- HASPCKPT
- HASPCNVS
- HASPCNVT
- HASPCOMM
- HASPFSSM
- HASPMSG

- HASPHOPE
- HASPIRA
- HASPIRPL
- HASPJOS
- HASPJQS
- HASPMSG
- HASPNET
- HASPNUC
- HASPPRPU
- HASPPSO
- HASPRDR
- HASPRTAM
- HASPSCAN
- HASPSERV
- HASPSNA
- HASCSRJB
- HASPSPIN
- HASPSPOL
- HASPSTAB
- HASPSUBS
- HASPSXDV
- HASPSXIT
- HASPSXJB
- HASPTERM
- HASPTRAK
- HASPWARM
- HASPXCF
- HASPXEQ

JES2 Parts

Appendix D. 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.

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.

Notices

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.

Programming Interface Information

This publication primarily documents information that is NOT intended to be used as Programming Interfaces of z/OS.

This publication also documents intended Programming Interfaces that allow the customer to write programs to obtain the services of z/OS. This information is identified where it occurs, either by an introductory statement to a chapter or section or by the following marking:
- Programming Interface information
- End of Programming Interface information

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, or other countries, or both:
- AFP
- AS/400
- BookManager
- DFSMS/MVS
- ES/3090
- ES/9000
- IBM
- IBMLink
- IMS
- MVS/DFP
- MVS/ESA
- NetView
- OS/390

- PR/SM
- Print Services Facility
- RACF
- Resource Link
- RMF
- SP
- SP1
- SP2
- System/390
- VTAM
- z/OS
- z/OS.e

NetView is a trademark of International Business Machines 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 JES2 documentation. If you do not find the term you are looking for, refer to the index of the appropriate JES2 manual or view *IBM Glossary of Computing Terms*, available from:

www.ibm.com/ibm/terminology

This glossary includes terms and definitions from:

American National Standard Dictionary for Information Systems, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 11 West 42nd Street, New York, New York 10036. Definitions are identified by an asterisk (*) that appears between the term and the beginning of the definition; a single definition taken from ANSI is identified by an asterisk after the item number for that definition.

A

ACB.   Access control block

ACF.   Advanced communication function

address space.   The complete range of addresses available to a program. See also *virtual address space*.

Advanced Function Presentation (AFP).   A set of licensed programs, together with user applications, that use the all-points-addressable concept to print on presentation devices. AFP includes creating, formatting, archiving, retrieving, viewing, distributing, and printing information. See *presentation device*.

affinity.   The condition under which one or more members of a JES2 multi-access spool configuration may be designated as qualified to execute certain jobs.

AFP.   See *Advanced Function Presentation*

all-member warm start.   A JES2 member restart of the first member in a multi-access spool (MAS) configuration. Either the JES2 member previously ended without error or there must be an IPL of the MVS system.

all points addressability.   The ability to address, reference, and position text, overlays, and images at any defined position or pel on the printable area of the paper. This capability depends on the ability of the hardware to address and to display each picture element.

allocate.   To assign a resource for use in performing a specific task.

APA.   See *all points addressability*

APAR.   Authorized program analysis report

APPC.   Advanced Program-to-Program Communication.

APT.   Application table

artificial JQE.   An artificial JQE consists of the base JQE, the JQX, and additional fields defined in the JQA.

automatic restart.   A restart that takes place during the current run, that is, without resubmitting the job. An automatic restart can occur within a job step or at the beginning of a job step. Contrast with *deferred restart*. See also *checkpoint restart*.

automatic volume recognition (AVR).   A feature that allows the operator to mount labeled volumes on available I/O devices before the volumes are needed by a job step.

AVR.   Automatic volume recognition

B

background.   (1) In multiprogramming, the environment in which low-priority programs are executed. (2) Under TSO/E the environment in which jobs submitted through the SUBMIT command or SYSIN are executed. One job step at a time is assigned to a region of central storage, and it remains in central storage to completion. Contrast with *foreground*.

background job.   (1) A low-priority job, usually a batched or non-interactive job. (2) Under TSO, a job entered through the SUBMIT command or through SYSIN. Contrast with *foreground job*.

BAL.   Basic assembler language

batch processing.   (1) *Pertaining to the technique of executing a set of computer programs such that each is completed before the next program of the set is started. (2) *Pertaining to the sequential input of computer programs or data. (3) *Loosely, the serial execution of computer programs. (4) Under TSO, the processing of one job step in a region, so called because jobs are submitted in a group or batch.

baud. (1) A unit of signaling speed equal to the number of discrete conditions or signal events per second. For example, one baud equals one-half dot cycle per second in Morse code, one bit per second in a train of binary signals, and one 3-bit value per second in a train of signals each of which can assume one of eight different states. (2) In asynchronous transmission, the unit of modulation rate corresponding to one unit interval per second; that is, if the duration of the unit interval is 20 milliseconds, the modulation rate is 50 baud.

binary synchronous communication (BSC). Communication using binary synchronous transmission.

binary synchronous transmission. Data transmission in which synchronization of characters is controlled by timing signals generated at the sending and receiving stations.

bind. In SNA products, a request to activate a session between two logical units.

broadcast data set. Under TSO, a system data set containing messages and notices from the system operator, administrators, and other users. Its contents are displayed to each terminal user when he logs on the system, unless suppressed by the user.

BSAM. Basic sequential access method

BSC. Binary synchronous communication

BSCA. Binary synchronous communication adapter

burst. *To separate continuous-form paper into discrete sheets.

C

cataloged data set. A data set that is represented in an index or hierarchy of indexes that provide the means for locating it.

cataloged procedure. A set of job control statements that has been placed in a library and that can be retrieved by name.

CCW. Channel command word

central storage. (1) In z/OS or System/390 virtual storage systems, the storage of a z/OS or System/390 computing system from which the central processing unit can directly obtain instructions and data, and to which it can directly return results. (Formerly referred to as "real storage".) (2) Synonymous with *processor storage*.

centralized control. Control in which all the primary station functions of the data link are centralized in one data station. Contrast with *independent control*.

CES. Connection event sequence

chain printer. An impact printer that has a revolving chain with links that carry the type slugs.

change log. Area of the checkpoint data set that contains the specific control blocks changed by the last member of the multi-access spool configuration to own the checkpoint data set.

channel-to-channel (CTC). A method of connecting two computing devices.

channel-to-channel (CTC) adapter. A device for connecting two channels on the same processor or on different processors.

checkpoint. (1) *A place in a routine where a check, or a recording of data for restart purposes, is performed. (2) A point at which information about the status of a job and the system can be recorded so that the job step can be later started. (3) To record information about the status of a job and the system for restart purposes.

checkpoint data set. A data set in which information about the status of a job and the system can be recorded so that the job step can be restarted later.

checkpoint reconfiguration. A process used by JES2 to dynamically redefine checkpoint data set specifications for a JES2 MAS.

checkpoint reconfiguration dialog. An interactive form of a JES2 checkpoint reconfiguration where the operator directs the reconfiguration process with replies to a series of WTOR messages.

checkpoint restart. The process of resuming a job at a checkpoint within the job step that caused abnormal termination. The restart may be automatic or deferred, where deferred restart involves resubmitting the job. See also *automatic restart*; *deferred restart*. Contrast with *step restart*.

checkpoint write. Any write to the checkpoint data set. A general term for the primary, intermediate, and final writes that update any checkpoint data set.

checkpoint/restart facility. (1) A facility for restarting execution of a program at some point other than at the beginning, after the program was terminated due to a program or system failure. A restart can begin at a checkpoint or from the beginning of a job step, and uses checkpoint records to reinitialize the system. (2) Under TCAM, a facility that records the status of the teleprocessing network at designated intervals or following certain events. Following system failure, the system can be restarted and continue without loss of messages.

checkpointing. Preserving processing information during a program's operation that allows such processing to be restarted and duplicated.

CKPT1. The checkpoint data set designed as the one on which the reserve is acquired. In a DUAL mode configuration, CKPT1 is one of the alternately used primary data sets from which JES2 reads and writes the checkpoint. In a DUPLEX mode configuration, CKPT1 is the primary checkpoint data set.

CKPT2. In a DUAL mode configuration, CKPT2 is one of the alternately-used checkpoint data sets from which JES2 reads and writes the checkpoint. In a DUPLEX mode configuration, CKPT2 is the back-up copy (generally down-level) of the primary checkpoint data set (CKPT1) which can be used to replace CKPT1 if necessary. CKPT2 is formatted the same as CKPT1. (Previously CKPT2 was the DUPLEX checkpoint data set).

CLPA. Common link pack area

CMB. Console message buffer

CMS. Cross memory services

cold start. A JES2 member start that initializes data areas and accounting information in central storage and the job and output queues.

communication line. Any physical link, such as a wire or telephone circuit, for connecting geographically dispersed computer systems.

complex. The maximum set of hardware and software resources that support one or more images of a single operating system.

configuration. The arrangement of a computer system or network as defined by the nature, number, and chief characteristics of its functional units.

connection event sequence. A clock value that indicates the time a connection took place or was broken. This is copied to NCC records and used by the path manager to determine the "most current" record when keeping track of NJE connections.

console. Any device from which operators can enter commands or receive messages. For JES2, the same device from which an operator also enters MVS base control program commands.

control statements. Statements placed into an input stream to identify special JES2 processing options for jobs.

CSA. Common service area

CSECT. Control section

CTC. Channel-to-channel adapter

D

DASD. Direct access storage device

data integrity point. The generic name given to the point in the 3800 model 3 printing process at which the data is known to be secure. (Also called the stacker.)

data set forwarding. The dynamic replacement of the checkpoint data set specifications (data set name and volume) with new specifications.

data set separator pages. Those pages of printed output that delimit data sets.

DCT. Device control table

deallocate. To release a resource that is assigned to a specific task.

dedicated. Pertaining to the assignment of a system resource - a device, a program, or a whole system - to an application or purpose.

deferred-printing mode. A printing mode that spools output through JES to a data set instead of printing it immediately. Output is controlled by JCL statements.

deferred restart. A restart performed by the system when a user resubmits a job. The operator submits the restart deck to the system through a system input reader. See also *checkpoint restart*. Contrast with *automatic restart*.

dependent job control (DJC). A method of handling multiple jobs that must be run in a specific order because of job dependencies.

despooling. The process of reading records off the spool into central storage. During the despooling process, the physical track addresses of the spool records are determined.

destination. A combination of a node name and one of the following: a userid, a remote printer or punch, a special local printer, or LOCAL (the default if only a node name is specified).

destination identifier (destid). The 8-character subscript on the DESTID(jxxxxxxx) initialization statement or command that corresponds to a combination of a first-level destination and a second-level destination that determines where data should be sent in a JES2 installation. A destid can be either a symbolic destination or an explicit destination.

destination node. Node to which data is sent.

device partitioning. A pool of devices (called a fence) to be used exclusively by a set of jobs in a specific job class allowing an installation to tailor its device usage to its anticipated workload.

direct access storage device (DASD). A device in which the access time is effectively independent of the location of the data.

DJC. Dependent job control.

DUAL mode. A checkpointing mode that provides the alternate use of two primary checkpoint data sets (CKPT1 and CKPT2). The data sets are referred to as the to-be-read-from and to-be-written-to data sets.

dump. A report showing the contents of storage. Dumps are typically produced following program failures, for use as diagnostic aids.

DUPLEX mode. A checkpointing mode that provides the continuous use of only one checkpoint data set. A second (backup) data set is defined, but it is written to less frequently than the primary.

dynamic allocation. Assignment of system resources to a program at the time the program is executed rather than at the time it is loaded into central storage.

dynamic connection. A connection created via sign-on or NCC record sent from another node Synonymous with *non-static connection*.

dynamic table. An installation-defined table that is used to extend, modify, or delete the JES2 default processing specifications. See also *table pair*.

E

EBCDIC. Extended binary coded decimal interchange code

ECSA. Extended common service area

EM. End of media

end of block (EOB). A code that marks the end of a block of data.

end-of-file mark (EOF). A code that signals that the last record of a file has been read.

EOB. End of block

EOF. End of file

EPVT. Extended private storage area

execution node. The JES2 network job entry node upon which a job is to be executed.

exit points. The place in the code where a routine (exit) receives control from the system.

explicit destination. A destination identifier of the form Nnnnn, Rmmmm, RMmmmm, RMTmmmm, NnnnnRmmmm or Unnnn. See also *destination identifier* and *symbolic destination*.

extended binary coded decimal interchange code (EBCDIC). A set of 256 characters, each represented by 8 bits.

external writer. A program that supports the ability to write SYSOUT data in ways and to devices not supported by the job entry subsystem.

F

facility. (1) A feature of an operating system, designed to service a particular purpose, for example, the checkpoint/restart facility. (2) A measure of how easy it is to use a data processing system. Together with system performance, a major factor on which the total productivity of an installation depends. (3) Anything used or available for use in furnishing communication service. Commonly, a general term for communications paths.

FCB. Forms control buffer

final write. A write of the same information as the intermediate write done at the end of the checkpoint cycle. See also *intermediate write*.

first-level destination. The nodal portion of a destination (the node to which the data goes).

foreground. (1) in multiprogramming, the environment in which high-priority programs are executed. (2) Under TSO, the environment in which programs are swapped in and out of central storage to allow CPU time to be shared among terminal users. All command processor programs execute in the foreground. Contrast with *background*.

foreground job. (1) A high-priority job, usually a real-time job. (2) A teleprocessing or graphic display job that has an indefinite running time during which communication is established with one or more users at local or remote terminals. (3) Under TSO, any job executing in a swapped region of central storage, such as a command processor or a terminal user's program. Contrast with *background job*.

forms control buffer (FCB). A buffer that is used to store vertical formatting information for printing; each position corresponding to a line on the form.

forwarding. The dynamic replacement of the checkpoint data set specifications (data set name and volume) with new specifications.

FSA. Functional subsystem application

FSA startup. That part of system initialization when the FSA is loaded into the functional subsystem address space and begins initializing itself.

FSI. Functional subsystem interface

FSI connect. The FSI communication service which establishes communication between JES2 and the FSA or functional subsystem.

FSI disconnect. The FSI communication service which severs the communication between JES2 and the FSA or functional subsystem.

FSI services. A collection of services available to users (JES2) of the FSI. These services comprise communication services, data set services, and control services.

FSS. Functional subsystem

full function mode. The state that permits a printer to produce page-mode output.

functional subsystem (FSS). An address space uniquely identified as performing a specific function related to the JES. For JES2, an example of an FSS is the Print Services Facility program that operates the 3800 Model 3 and 3820 printers.

functional subsystem application (FSA). The functional application program managed by the functional subsystem.

functional subsystem interface (FSI). The interface through which JES2 or JES3 communicate with the functional subsystem.

functional subsystem startup. That process part of system initialization when the functional subsystem address space is created.

G

global command. A command that is recognized and honored by any node in a JES2 network.

global processor. In JES3, the processor that controls job scheduling and device allocation for a complex of processors.

GMT. Greenwich mean time.

Greenwich mean time (GMT). The mean solar time of the meridian of Greenwich used as the prime basis of standard time throughout the world. See also *TOD clock*.

H

handshaking. Exchange of predetermined signals when a connection is established between two data set devices.

HASP. Houston automatic spooling priority. A computer program that provides supplementary job management, data management, and task management functions, such as: control of job flow, ordering of tasks, and spooling. See also *JES2*.

HASP table. See *JES2 table.*

HCT. HASP communication table

host processor. (1) *In a network, the processing unit in which resides the access method for that network. (2) In an SNA network, the processing unit that contains a system services control point (SSCP).

host system. *The data processing system to which a network is connected and with which the system can communicate.

host-id. The unique 10-digit CPU identification made up of the 6-digit CPU serial number followed by a 4-digit model number.

hot start. A JES2 member restart performed when a member ends abnormally and the MVS system is not re-IPLed.

I

I/O. input/output

IBM-defined exit. The point in source code where IBM has added an exit point where an installation routine can receive control from the operating system. Contrast with *installation-defined exit*.

impact printer. *A printer in which printing results from mechanical impacts.

independent control. In JES2, the process by which each processor in a complex controls its own job input, scheduling, and job output. Contrast with *centralized control*.

independent mode. A means of isolating a processor for testing purposes. A processor so designated will only process jobs that are both routed to it and are themselves designated to execute on a processor in independent mode.

initial program load (IPL). The initialization procedure that causes an operating system to commence operation.

initialization data set. The data set that contains the initialization statements and their parameters that controls the initialization and ultimate processing of JES2.

initialization parameter. An installation-specified parameter that controls the initialization and ultimate operation of JES2.

initialization statement. An installation-specified statement that controls the initialization and ultimate operation of JES2.

initiating task. The job management task that controls the selection of a job and the preparation of the steps of that job for execution.

initiator. That part of an operating system that reads and processes operation control language statements from the system input device.

initiator/terminator. The job scheduler function that selects jobs and job steps to be executed, allocates input/output devices for them, places them under task control, and at completion of the job, supplies control information for writing job output on a system output unit.

input service processing. In JES2, the process of performing the following for each job: reading the input data, building the system input data set, and building control table entries.

input stream control. Synonymous with *JES2 reader*.

installation-defined exit. The point in source code where an installation adds an exit point where an installation routine can receive control from the operating system. Contrast with *IBM-defined exit*.

interface. Hardware, software, or both, that links systems, programs, or devices.

intermediate write. In DUAL mode, the write of the change log records containing the control blocks that have been updated since the last checkpoint write. In DUPLEX mode (or DUAL mode where the change log overflows the first track) the checkpoint write of the 4K records.

internal reader. A facility that transfers jobs to JES.

interrupt. (1) *To stop a process in such a way that it can be resumed. (2) In data transmission, to take an action at a receiving station that causes the transmitting station to terminate a transmission.

IOT. input/output table

IPL. initial program load

IPS. Installation performance specification

J

JCL. Job control language

JCT. Job control table

JES2. Job entry subsystem 2. An MVS subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In an installation with more than one processor, each processor's JES2 subsystem independently controls job input, scheduling, and output processing.

JES2 reader. In MVS, the part of the job entry subsystem that controls the input stream and its associated job control statements. Synonymous with *input stream control*.

JES2 table. A JES2-defined table that is used to specify the default characteristics of many of its initialization parameters, commands, and other externals. See also *table pair*.

JES2 writer. In MVS, the part of the job entry subsystem that controls the output of specified data sets. Synonymous with *output stream control*.

JES3. Job entry subsystem 3. An MVS subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In an installation with multiple processors (a JES3 complex), one processor's JES3 subsystem exercises centralized control over the other processors and distributes jobs to them through use of a common job queue.

JIX. Job queue index

JMR. Job management record

job. A unit of work for an operating system. Jobs are defined by JCL statements.

job class. Any one of a number of job categories that can be defined. With the classification of jobs and direction of initiator/terminators to initiate specific classes of jobs, it is possible to control the mixture of jobs that are performed concurrently.

job control language (JCL). A programming language used to code job control statements.

job control language (JCL) statements. Statements placed into an input stream to define work to be done, methods to be used, and the resources needed.

job control statement. *A statement in a job that is used in identifying the job or describing its requirements to the operating system.

job entry subsystem (JES). An MVS facility that receives jobs into the system and processes output data produced by the jobs. See also *JES2* and *JES3*.

job entry subsystem 2. See *JES2*.

job entry subsystem 3. See *JES3*.

job output element (JOE). Information that describes a unit of work for the output processor and represents that unit of work for queuing purposes.

job priority. A value assigned to a job that is used as a measure of the job's relative importance while the job contends with other jobs for system resources.

job queue element (JQE). A control block that represents an element of work for the system (job) and is moved from queue to queue as that work moves through each successive stage of JES2 processing.

job separator page data area (JSPA). A data area that contains job-level information for a data set. This information is used to generate job header, job trailer or data set header pages. The JSPA can be used by an installation-defined JES2 exit routine to duplicate the information currently in the JES2 separator page exit routine.

job separator pages. Those pages of printed output that delimit jobs.

JOE. Job output element

JOT. Job output table

K

keyword. A part of a command operand that consists of a specific character string (such as DSNAME=).

keyword parameter. A parameter that consists of a keyword, followed by one or more values. Contrast with *positional parameter*. See also *parameter*.

L

label. (1) *One or more characters used to identify a statement or an item of data in a computer program. (2) An identification record for a tape or disk file.

line mode. A type of data with format controls that only allow a printer to format data as a line.

line mode data. A type of data that is formatted on a physical page by a printer only as a single line.

LMT. Load module table

local devices. Those devices that are directly attached to the operating system without the need for transmission facilities.

local processing environment. The collection of devices all of which are locally attached. That is, they are connected without the need for transmission facilities.

local system queue area (LSQA). In MVS, one or more segments associated with each virtual storage region that contain job-related system control blocks.

locally attached. A manner of device connection without the need for transmission facilities.

logical unit (LU). The combination of programming and hardware of a teleprocessing subsystem that functions like a terminal to VTAM.

logoff. (1) The procedure by which a user ends a terminal session. (2) In VTAM, a request that a terminal be disconnected from a VTAM application program.

logon. (1) The procedure by which a user begins a terminal session. (2) In VTAM, a request that a terminal be connected to a VTAM application program.

loop. A situation in which an instruction or a group of instructions execute repeatedly.

LPA. Link pack area

LRECL. Logical record length

LSQA. Local system queue area

LU. Logical unit

M

machine check interruption. An interruption that occurs as a result of an equipment malfunction or error.

MAS. See *multi-access spool configuration*.

MCS. Multiple console support

member. A JES2 instance of a MVS system

message. For communication lines, a combination of characters and symbols transmitted from one point to another. See also *operator message*.

MIT. Module information table

MLU. Multiple logical unit

multi-access spool complex. See *multi-access spool configuration*.

multi-access spool configuration. Multiple systems sharing the JES2 input, job and output queues (via a checkpoint data set or coupling facility).

multi-access spool multiprocessing. Two or more computing systems interconnected by an I/O channel-to-channel adapter. The CPs can be different types and have their own unique configurations.

multiple console support (MCS). A feature of MVS that permits selective message routing to up to 32 operator's consoles.

Multiple Virtual Storage (MVS). An operating system that manages resources and work flow while jobs are running.

multiprocessing. (1) *Pertaining to the simultaneous execution of two or more computer programs or sequences of instructions by a computer network. (2) *Loosely, parallel processing. (3) Simultaneous execution of two or more sequences of instructions by a multiprocessor.

multiprocessing system. A computing system employing two or more interconnected processing units to execute programs simultaneously.

multiprocessor. (1) A computer employing two or more processing units under integrated control. (2) A system consisting of two or more CPs (or ALUs, or processors) that can communicate without manual intervention.

MVS. Multiple virtual storage.

N

NACT. Network account table

NAT. The nodes attached table, which is an internal JES2 control block containing information about each pair of nodes connected, or recently disconnected.

NCC record. The network connection and control records.

NCP. Network control program

NCP/VS. Network control program/VS

NDH. Network data set header

network. For JES2, two or more systems and the connections over which jobs and data are distributed to the systems. The other systems can be non-JES2 systems with compatible networking facilities. Connections can be established through communications paths using SNA or BSC protocols.

network job entry (NJE). A JES2 facility that provides for the passing of selected jobs, system output data, operator commands, and messages between communicating job entry subsystems connected by binary-synchronous communication lines, channel-to-channel adapters, and shared queues.

Network Job Entry (NJE) facility. In JES2, a facility which provides for the transmission of selected jobs, operator commands, messages, SYSOUT data, and accounting information between communicating job entry nodes that are connected in a network either by binary synchronous communication (BSC) lines channel-to-channel (CTC) adapters, or by System Network Architecture (SNA).

Network Job Entry facility. In JES2, a facility which provides for the transmission of selected jobs, operator commands, messages, SYSOUT data, and accounting information between communicating job entry nodes that are connected in a network either by binary synchronous communication (BSC) lines or by channel-to-channel (CTC) adapters.

network operator. (1) The person responsible for controlling the operation of a telecommunication

network. (2) A VTAM application program authorized to issue network operator commands.

NIP. Nucleus initialization program.

NIT. The node information table, which is an internal JES2 control block containing information about each NJE node.

NJE. Network job entry

NJH. Network job header

node. (1) One of the systems in a network of systems connected by communication lines or CTC adapters. (2) In VTAM, an addressable point in a telecommunication system defined by a symbolic name. (3) In JES2 NJE, one or more job entry subsystems sharing a common job queue.

node name. An 8-character alphameric name which represents a node to other parts of the NJE network.

non-impact printer. *A printer in which printing is not the result of mechanical impacts; for example, thermal printers, electrostatic printers, photographic printers.

non-static connection. A connection created via sign-on or NCC record sent from another node Synonymous with *dynamic connection*.

nonpageable dynamic area. *In MVS, an area of virtual storage whose virtual addresses are identical to real addresses; it is used for programs or parts of programs that are not to be paged during execution. Synonymous with *V=R dynamic area*.

nonpageable region. In MVS, a subdivision of the nonpageable dynamic area that is allocated to a job step or system task that is not to be paged during execution. In a nonpageable region, each virtual address is identical to its real address. Synonymous with *V=R region*.

nucleus. That portion of a control program that always remains in central storage.

nucleus initialization program (NIP). The MVS component that initializes the resident control program.

O

offload. Moving jobs and work off the work queues to remove them from contention for system resources, or off spool to free up system work space.

operand. (1) *That which is operated upon. An operand is usually identified by an address part of an instruction. (2) Information entered with a command name to define the data on which a command processor operates and to control the execution of the command processor.

operator commands. Statements that system operators may use to get information, alter operations, initiate new operations, or end operations.

operator message. A message from an operating system directing the operator to perform a specific function, such as mounting a tape reel; or informing the operator of specific conditions within the system, such as an error condition.

operator orientation point. The generic name given to the point in the 3800 model 3 printing process at which the data becomes visible to the operator, and is therefore the point at which all operator commands are directed. Synonymous with *transfer station*.

output group. A set of a job's output data sets that share output characteristics, such as class, destination, and external writer.

output stream control. Synonymous with *JES2 writer*.

output writer. A part of the job scheduler that transcribes specified output data sets onto a system output device independently of the program that produced the data sets.

overlays. A collection of predefined data such as lines, shading, text, boxes, or logos, that can be merged with the variable data on a page while printing.

P

page. (1) In virtual storage systems, a fixed-length block of instructions, data, or both, that can be transferred between central storage and external page storage. (2) To transfer instructions, data, or both, between central storage and external page storage. (3) The unit of output from an AFP printer, such as the 3800-3, running with full function capability or 3820 printer.

page data set. In z/OS or System/390 virtual storage systems, a data set in external page storage in which pages are stored.

page fault. In z/OS or System/390 virtual storage systems, a program interruption that occurs when a page that is marked "not in central storage" is referred to by an active page.

page mode. The mode of operation in which the AFP print (such as the 3800 Printing Subsystem) can accept a page of data from a host processor to be printed on an all points addressable output medium.

page mode data. A type of data that can be formatted anywhere on a physical page. This data requires specialized processing such as provided by the Print Services Facility for AFP printers, such as the 3800-3 and 3820.

page mode environment checkpointing. That process which preserves the information necessary to resume page-mode printing.

page mode printer. An AFP printer, such as the 3800 model 3 and 3820, that can print page mode data.

pageable region. In MVS, a subdivision of the pageable dynamic area that is allocated to a job step or a system task that can be paged during execution. Synonymous with *V=V region*.

paging. In z/OS or System/390 virtual storage systems, the process of transferring pages between central storage and external page storage.

paging device. In z/OS or System/390 virtual storage systems, a direct access storage device on which pages (and possibly other data) are stored.

parameter. (1) *A variable that is given a constant value for a specific purpose or process. (2) See *keyword parameter, positional parameter*.

password. A unique string of characters that a program, computer operator, or user must supply to meet security requirements for gaining access to data.

patch. *To modify a routine in a rough or expedient way.

path. In VTAM, the intervening nodes and lines connected a terminal and an application program in the host CPU.

path manager. The part of JES2 that controls NJE sign-on, sign-off, keeps track of all other nodes and connections in the network, and determines the best path to reach those nodes. (JES2 is unique among other NJE subsystems in keeping track of the network topology through NCC records.)

PCE. Processor control element

pel. Picture element

PDDB. Peripheral data definition block

PEP. Partitioned emulator program

physical unit (PU). (1) The control unit or cluster controller of an SNA terminal. (2) The part of the control unit or cluster controller that fulfills the role of a physical unit as defined by systems network architecture (SNA).

PLPA. Pageable link pack area

poly-JES. Concurrent operation of multiple copies of JES2 on a single MVS system to allow an installation to separate its primary production system(s) and test system(s).

positional parameter. A parameter that must appear in a specified location, relative to other parameters. Contrast with *keyword parameter*. See also *parameter*.

PPL. Purge parameter list

PRE. Processor recovery element

presentation device. A device that produces character shapes, graphics pictures, images, or bar code symbols on a physical medium. Examples of physical media are display screens, paper, foils, microfilm, and labels.

primary write. The write of the 4K records to the down-level checkpoint data set to make it current.

Print Services Facility (PSF). An IBM licensed program that produces printer commands from the data set to it. PSF programs run on the z/OS, OS/390, MVS, VM, VSE, OS/2, AIX, and OS/400 operating platforms. For JES, PSF programs operates the 3800 model 3 and 3820 printers. PSF operates as a functional subsystem.

priority aging. A function of JES2 by which the longer a job waits to be selected for processing, the greater become its chances of being selected to run.

private connection. A connection known only to the two nodes making the connection.

process mode. The mode in which SYSOUT data exists and is to be processed by a JES output device. There are two IBM-defined process modes: line mode and page mode.

processor storage. See *central storage*.

program temporary fix (PTF). A temporary solution or bypass for a problem diagnosed by IBM as the result of a defect in a current unaltered release of the program.

protocols. Rules for using communication lines. Protocols can identify the direction of data flow, where data begins and ends, how much data is being transmitted, and whether data or control information is being sent.

PSF. Print Services Facility

PTF. Program temporary fix

PU. Physical unit.

Q

QSE. Shared queue element

queue. A line or list formed by items in a system waiting for processing.

quick start. A JES2 member restart in an existing multi-access spool (MAS) configuration. The JES2 member previously ended without error.

quiescing. *The process of bringing a device or a system to a halt by rejection of new requests for work.

R

RACF. Resource Access Control Facility

read 1. A read of the first track of a checkpoint data set. Usually performed as the initial I/O operation to a checkpoint data set.

read 2. A read of the 4K page data records and any change log records not contained on the first track from a checkpoint data set. Usually performed after a READ 1 as the second checkpoint I/O operation in a checkpoint cycle.

reader. A program that reads jobs from an input device or data base file and places them on the job queue.

real address. In virtual storage systems, the address of a location in central storage.

real storage. See *central storage*.

remote. RMT

remote job entry (RJE). Submission of job control statements and data from a remote terminal, causing the jobs described to be scheduled and executed as though encountered in the input stream.

remote station. *Data terminal equipment for communicating with a data processing system from a location that is time, space, or electrically distant.

remote terminal. An input/output control unit and one or more input/output devices attached to a system through a data link.

remote terminal access method (RTAM). A facility that controls operations between the job entry subsystem (JES2) and remote terminals.

remote workstation. (1) *Data terminal equipment for communicating with a data processing system from a location that is time, space, or electrically distant. Synonymous with *remote station*. (2) A workstation that is connected to a system by means of data transmission facilities.

RJE. Remote job entry

RMS. Recovery management support

RMT. Remote

RMT generation. Generation of remote workstations for remote job entry.

routing. (1) The assignment of the communications path by which a message or telephone call will reach its destination. (2) In NJE, the path, as determined by NJE or explicitly by the operator, that a job or SYSOUT data set will take to reach its destination.

routing code. A code assigned to an operator message and used, in systems with multiple console support (MCS), to route the message to the proper console.

RPL. Request parameter list

RPS. Rotational position sensing

RTAM. Remote terminal access method

RTP. Remote terminal program

S

SAF. Security authorization facility

SAM. Sequential access method

SDLC. Synchronous data link control

SDSB. Spool data set browse

second-level destination. Specifies a remote workstation, special local route code, userid, or LOCAL or ANYLOCAL (for data not associated with a specific routing).

secondary console. In a system with multiple consoles, any console except the master console. The secondary console handles one or more assigned functions on the multiple console system.

security classification. (1) An installation-defined level of security printed on the separator pages of printed output. (2) In RACF, the use of security categories, a security level, or both, to impose additional access controls on sensitive resources. An alternative way to provide security classifications it to use security labels.

segments. A collection of composed text and images, prepared before formatting and included in a document when it is printed.

session. (1) The period of time during which a user of a terminal can communicate with an interactive system; usually, the elapsed time from when a terminal is logged on to the system until it is logged off the system. (2) The period of time during which programs or devices can communicate with each other. (3) In VTAM, the period of time during which a node is connected to an application program.

setup. The preparation of a computing system to perform a job or job step. Setup is usually performed by an operator and often involves performing routine functions, such as mounting tape reels and loading card decks.

shared broadcasting. The two TSO data sets SYS1.UADS (TSO user definition) and SYS1.BRODCAST (TSO message transmission definition) are shared by all systems in the multi-access spool (MAS) complex.

simultaneous peripheral operations online (spool). The reading and writing of input and output streams on auxiliary storage devices, concurrently while a job is running, in a format convenient for later processing or output operations.

single-member warm start. A JES2 member restart of a new member in an existing multi-access spool (MAS) configuration. The JES2 member previously ended abnormally. Before the restart can occur, there must be an IPL of the MVS system.

single-processor complex. A processing environment in which only one processor (computer) accesses the spool and comprises the entire node.

SMF. System management facilities

SNA. Systems Network Architecture

special local. A routing in the form Unnnn, where 'nnnn' signifies a numeric value in the range of 1–32767. Usually, installations use this routing to specify local printers and punches.

spin data set. A data set that is deallocated (available for printing) when it is closed. Spin off data set support is provided for output data sets just prior to the termination of the job that created the data set.

spool. Simultaneous peripheral operations online.

spooled data set. A data set written on an auxiliary storage device and managed by JES.

spooled data set browse (SDSB). An application that allows a program to read spool data sets.

spooling. The reading and writing of input and output streams on auxiliary storage devices, concurrently with job execution, in a format convenient for later processing or output operations.

SQA. System queue area

SRM. System resources manager

static connection. A connection (also called "predefined connection" in earlier releases) between two nodes created by either a JES2 initialization or an operator command.

STC. Started task control

step restart. A restart that begins at the beginning of a job step. The restart may be automatic or deferred, where deferral involves resubmitting the job. Contrast with *checkpoint restart*.

subnet. Subset of a NJE network identified by an eight-character 'SUBNET' name on the JES2 NODE initialization statement. The grouping of nodes into "SubNets" is based on the assumption that if you have access to any node in the subnet, you have access to them all.

subsystem. A secondary or subordinate system, usually capable of operating independently of, or asynchronously with, a controlling system.

SVC. Supervisor call instruction

SVC interruption. An interruption caused by the execution of a supervisor call instruction, causing control to be passed to the supervisor.

SWA. Scheduler work area

swap data set. A data set dedicated to the swapping operation.

swapping. An MVS paging operation that writes the active pages of a job to auxiliary storage and reads pages of another job from auxiliary storage into central storage.

symbol. (1) *A representation of something by reason of relationship, association, or convention. (2) In MVS, a group of 1 to 8 characters, including alphanumeric characters and the three characters: #, @, $. The symbol begins with either an alphabetic character or one of the three characters (#,@,$).

symbolic address. *An address expressed in symbols convenient to the computer programmer.

symbolic destination. A destination identifier specifying a symbolic name that represents a JES2 destination. See also *destination identifier* and *explicit destination*.

synchronous data link control (SDLC). A discipline for managing synchronous, transparent, serial-by-bit information transfer over a communication channel. Transmission exchanges may be duplex or half-duplex over switched or nonswitched data links. The communication channel configuration may be point-to-point, multipoint, or loop.

syntax. (1) *The structure of expressions in a language. (2) The rules governing the structure of a language.

SYSIN. A system input stream; also, the name used as the data definition name of a data set in the input stream.

SYSLOG. System log

SYSOUT. A system output stream; also, an indicator used in data definition statements to signify that a data set is to be written on a system output unit.

sysplex. A set of MVS systems communicating and cooperating with each other through certain multisystem hardware components and software services to process customer workloads.

system affinity. See *affinity*.

system control programming. IBM-supplied programming that is fundamental to the operation and maintenance of the system. It serves as an interface with program products and user programs and is available without additional charge.

system management facilities (SMF). An MVS component that provides the means for gathering and recording information that can be used to evaluate system usage.

system output writer. A job scheduler function that transcribes specified output data sets onto a system output unit, independently of the program that produced the data sets.

system queue area (SQA). In MVS, an area of virtual storage reserved for system-related control blocks.

system services control point. *In SNA, the focal point within an SNA network for managing the configuration, coordinating network operator and problem determination requests, and providing directory support and other session services for end users of the network.

systems network architecture (SNA). The total description of the logical structure, formats, protocols, and operational sequences for transmitting information units through a communication system.

T

table pair. A set of JES2-defined, USER-defined, and dynamic tables that an installation can use to modify JES2 processing.

TCAM. Telecommunications access method.

telecommunications access method (TCAM). A method used to transfer data between central storage and remote or local terminals. Application programs use either GET and PUT or READ and WRITE macro instructions to request the transfer of data, which is performed by a message control program. The message control program synchronizes the transfer, thus eliminating delays for terminal/output operations.

teleprocessing. The processing of data that is received from or sent to remote locations by way of telecommunication lines.

terminal. A device, usually equipped with a keyboard and some kind of display, capable of sending and receiving information over a communication channel.

text transparency. A provision that allows BSC to send and receive messages containing any or all of the 256 character combinations in EBCDIC, including transmission control characters. Transmission control characters sent in a message are treated as data unless they are preceded by the data link escape (DLE) control character.

TGB. Track group block

TGBE. Track group block entry

tightly-coupled multiprocessing. Two computing systems operating simultaneously under one control program while sharing resources.

Time Sharing Option Extensions (TSO/E). A licensed program that is based on the Time Sharing Option (TSO). It allows MVS users to interactively share computer time and resources.

time tolerance. The difference between the TOD clocks on two adjacent nodes, beyond which the path manager will not allow a session to be established.

time-of-day clock. See *TOD clock*.

TOD. Time-of-day

TOD clock. A timing device that counts units of time based on the starting point of 00 hours, 00 minutes, and 00 seconds on January 1, 1900. Time-of-day (TOD) information is used, for example, to monitor computer operations and events.

token. Specifically defined for JES2 checkpoint processing as a checkpoint identifier that is used to determine checkpoint I/O status.

trace. (1) The record of a series of events. (2) To record a series of a events as they occur. (3) A report showing data relevant to a particular point in the processing of a program. Traces are typically produced for analysis of program performance, but they can also be valuable diagnostic aids.

tracing routine. *A routine that provides a historical record of specified events in the execution of a program.

traffic. In data communication, the quantity of data transmitted past a particular point in a path.

train printer. A printer in which the type slugs are assembled in a train that moves along a track. Contrast with *chain printer*.

transfer station. The point in the 3800 model 3 printing process at which the data set becomes visible

to the operator, and is therefore the point at which all operator commands are directed. Synonymous with *operator orientation point*.

TSO. Time-sharing option. See *Time Sharing Option Extensions (TSO/E)*.

TSO/E. Time Sharing Option Extensions

TSU. Time-sharing user

TTE. Trace table entry

type font. In printing, a set of type that is of a particular size and style (for example, 10-point century school book).

U

UCB. Unit control block

UCS. Universal character set.

unallocate. See *deallocate*.

unit. (1) *A device having a special function. (2) A basic element.

unit address. The address of a particular device, specified at the time a system is installed; for example, 191 or 293.

universal character set (UCS). A printer feature that permits the use of a variety of character arrays.

user identification (USERID). A 1-8 character symbol identifying a system user.

user table. An installation-defined table that is used to extend, modify, or delete the JES2 default processing specifications. See also *table pair*.

USERID. User identification.

V

V=R dynamic area. Synonymous with *nonpageable dynamic area*.

V=R region. Synonymous with *nonpageable region*.

V=V region. Synonymous with *pageable region*.

VIO. virtual input/output

virtual address space. In virtual storage systems, the virtual storage assigned to a job, terminal user, or system task. See also *address space*.

Virtual Telecommunications Access Method (VTAM). A set of programs that control communication between terminals and application programs running under MVS.

VTAM. Virtual Telecommunications Access Method.

W

warm start. A general term for a JES2 member restart. See also *hot start; quick start; single-member warm start; all-member warm start*.

writer. See *output writer*.

WTO. Write-to-operator

WTOR. Write-to-operator with reply

X

XFER. Transfer

XIT. Exit information table

XRT. Exit routine table

Numerics

3800 compatibility mode. Operating the 3800 model 3 printer as a 3800 Model 1 printer.

3800 model 3 startup. That process part of system initialization when the 3800 model 3 printer is initializing.

Index

A
accessibility   157
application development
  considerations   4

B
base release
  terminology   2

C
coexistence, definition   2
common migration activities   5
compatible change release
  terminology   2

D
developing a migration strategy   3
diagnosis   4
disability   157
documents, licensed   xiii

I
incompatible change release
  terminology   2
initialization
  considerations   4
interface considerations   4

K
keyboard   157

L
licensed documents   xiii
LookAt message retrieval tool   xiii

M
MAS   3
member of MAS   3
message retrieval tool, LookAt   xiii
migration
  overview   1
  roadmap   9
  strategy   3
  terminology   2
modifications
  general considerations   4
multi-access spool   3

O
operational considerations   4
OS/390 JES2 summary
  SSI function codes   145
overview of changes
  for JES2 releases   15
overview, migration   1

P
planning for migration   3
problem determination   4
processing considerations   4

R
RACROUTE REQUEST=EXTRACT
  macro   7
roadmap, migration   9

S
shortcut keys   157
strategy, migration   3
summary of changes
  for JES2 z/OS V1R1   12
  for JES2 z/OS V1R2   11
  for JES2 z/OS V1R4   10
  for OS/390 JES2 Version 2 Release
   5   14
  for OS/390 JES2 Version 2 Release
   7   13, 34
  for OS/390 JES2 Version 2 Release
   8   13
supported migration paths   9

T
target release
  terminology   2
task considerations   4

Y
year 2000 support
  ICHEINTY macro   7
  RACF report writer   7
  RACROUTE REQUEST=EXTRACT   7

Readers' Comments — We'd Like to Hear from You

z/OS
JES2 Migration

Publication No. GA22-7538-03

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.

IBM®

Fold and Tape   **Please do not staple**   Fold and Tape

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

IBM®

Program Number: 5694-A01

Printed in U.S.A.