

NaSPA

November 5, 2008 – New York, NY



Migrating to z/OS 1.10 - Part 2

Marna WALLE
IBM Systems and Technology Group
Poughkeepsie New York
mwalle@us.ibm.com



© 2008 IBM Corporation

Systems and Technology Group IBM

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

AFP	DFSMSWin	FICON	MVS/SP	PFOSM	VTAM
ADK	DFSMSWin	HiperStorage	NetSpool	Print Services Facility	WebSphere
AspIlet	DFSORT	IBM	NetView	PrintWiz	zArchitecture
Base Mail manager	ESDA	IBM i	Open Class	RACF	zOS
CICS	ESCON	IBM i	OS/390	Resource Link	zVM
DIR	Eserv	Integrity	Parallel Sysplex	RMF	zSeries
DIR Universal Database	FICON	Language Environment		SG50	
DFSMS/DP		Multiple		SG50 Parallel Enterprise Server	
DFSMS/HS		MVS		SplexTimer	
		MVS/ESA		SystemPac	
				System z9	
				ThinL	
				TotalStorage	
				VisualAge	
				VSE/ESA	

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Jaws and all Jaws-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.

UNIX is a registered trademark of The Open Group in the United States and other countries.

SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is based throughout this PDF only based on measurements and procedures using standard IBM hardware in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of user's jobs, the I/O configuration, the channel configuration, and the channel processor. Therefore, an accuracy can be given that an individual user will achieve throughput/operations per second to the performance value stated here.

IBM hardware products are manufactured from new parts, or new and reworkable used parts. Regardless, we usually have to apply.

All customer computers cited or described in this publication are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual customer results and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document internationally, and the information may be subject to change without notice. Contact your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and expressed only as intentions only.

Information about new IBM products is obtained from the manufacturers of these products or their published announcements. IBM has not tested these products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Decisions on the capabilities of non-IBM products should be addressed to the suppliers of these products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

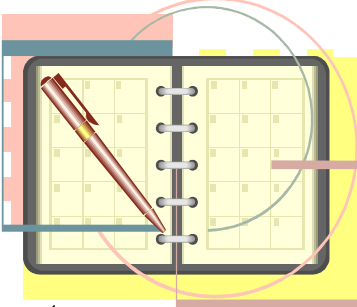
ENABLING BUSINESS.
A THROUGH Z.

© 2008 IBM Corporation

z/OS 1.10 Migration Actions IBM

z/OS 1.10 Migration Part 2 Agenda

- **Migration Actions:**
 - General Migration Actions for Everyone
 - Element-Specific Migration Actions for z/OS R10
 - BCP
 - Communications Server
 - DFSMS
 - JES2
 - Language Environment
 - z/OS UNIX
- **z/OS R10 Installation Enhancements**
- **z/OS R10 System Programmer Enhancements**
- **Summary**

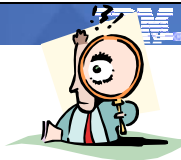



3 © 2008 IBM Corporation

z/OS 1.10 Migration Actions


Definition of "Migration Action":

- ▶ **Migration is**
 - the installation of a new version or release of a program to replace an earlier version or release.
- ▶ **After a successful migration, the applications and resources on the new system function the same way they did on the old system, if possible.**
- ▶ **Migration does not include exploitation of new functions except for new functions that are now required to use.**
- ▶ **Migration actions are classified as:**
 - **Required:** required for all users
 - **Required-IF:** only required in certain cases
 - **Recommended:** good to do because it 1) is good practice, 2) may be required in the future, 3) resolves performance or usability problem
- ▶ **Migration actions are also classified as when they may be performed:**
 - **NOW, Pre-First IPL, or Post-First IPL**





Means "don't overlook!"



Means a Health Check (or the Migration Checker) can help

4

Migration Definitions and Classifications

Migration is the first of two stages in upgrading to a new release of z/OS. The two stages are:

- **Stage 1: Migration.** During this stage you install your new system with the objective of making it functionally compatible with the previous system. After a successful migration, the applications and resources on the new system function the same way (or similar to the way) they did on the old system or, if that is not possible, in a way that accommodates the new system differences so that existing workloads can continue to run. Migration does not include exploitation of new functions except for new functions that are now required.
- **Stage 2: Exploitation.** During this stage you do whatever customizing and programming are necessary to take advantage of (exploit) the enhancements available in the new release. Exploitation follows migration.

Migration Requirement Classification and Timing

The migration actions are classified as to their requirement status:

- **Required.** The migration action is required in all cases.
- **Required-IF.** The migration action is required only in a certain case. Most of the migration actions in this presentation are in this category.
- **Recommended.** The migration action is not required but is recommended because it is a good programming practice, because it will be required in the future, or because it resolves unacceptable system behavior (such as poor usability or poor performance) even though resolution might require a change in behavior.

To identify the timing of migration actions, this presentation uses three types of headings:

- **Now.** These are migration actions that you perform on your current system, either because they require the current system or because they are possible on the current system. You don't need the z/OS V1R10 level of code to make these changes, and the changes don't require the z/OS V1R10 level of code to run once they are made. Examples are installing coexistence and fallback PTFs on your current system, discontinuing use of hardware or software that will no longer be supported, and starting to use existing functions that were optional on prior releases but required in z/OS V1R10.
- **Pre-First IPL.** These are migration actions that you perform after you've installed z/OS V1R10 but before the first time you IPL. These actions require the z/OS V1R10 level of code to be installed but don't require it to be active. That is, you need the z/OS V1R10 programs, utilities, and samples in order to perform the migration actions, but the z/OS V1R10 system does not have to be IPLed in order for the programs to run. Examples are running sysplex utilities and updating the RACF database template.

Migrating to z/OS 1.10 - Part 2

It is possible to perform some of the migration actions in this category even earlier. If you prepare a system on which you will install z/OS V1R10 by making a clone of your old system, you can perform migration actions that involve customization data on this newly prepared system before installing z/OS V1R10 on it. Examples of such migration actions are updating configuration files and updating automation scripts.

- **Post-First IPL.** These are migration actions that you can perform only after you've IPLed z/OS V1R10. You need a running z/OS V1R10 system to perform these actions. An example is issuing RACF commands related to new functions. Note that the term "first IPL" does not mean that you have to perform these actions after the very first IPL, but rather that you need z/OS V1R10 to be active to perform the task. You might perform the task quite a while after the first IPL.

Icons used in the subsequent foils:



means that you shouldn't overlook this migration action.



means that a Health Check (using the IBM Health Checker for z/OS function) or the IBM Migration Checker for z/OS tool can help you with this migration action.

z/OS 1.10 Migration Actions

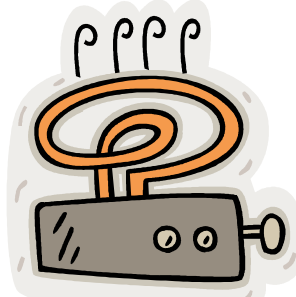
Elements with Migration Actions for z/OS R10

▶ **Documented in z/OS VIRI0 Migration (available on 8/5/08)**

- For complete migration tasks for z/OS R10, see this book!
 - from R8 to R10, and R9 to R10, customized books
 - "When behaviors aren't the same anymore, migration actions are called for."

▶ **From z/OS R8 to z/OS R10:**

<ul style="list-style-type: none"> ➢ BCP ▪ CIM ➢ Communications Server • Cryptographic Services – ICSF and OCSF ➢ DFSMS ▪ DFSORT • Distributed File Service – zFS and SMB • FFST • HCD • HCM • IBM Tivoli Directory Server • ICKDSF • Infoprint Server • Integrated Security Services - LDAP 	<ul style="list-style-type: none"> • ISPF ➢ JES2 • JES3 ➢ Language Environment • Library Server • NFS • RMF • SDSF • Security Server (RACF) • SMP/E • TSO/E • XL C/C++ ➢ z/OS UNIX
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



➢ means that some of that element's migration actions follow within this presentation
 • means that you need to look for that element's migration actions in the book


5
© 2008 IBM Corporation

Migration Actions for Elements Between z/OS R8 and z/OS R10

When migrating from z/OS R8 to z/OS R10, the specified elements in the foil above have required migration actions. Refer to *z/OS Migration* for complete information on the required migration actions for all elements. Some migration actions for selected elements follow in this presentation.

If you are migrating from z/OS R9 to R10, there are fewer migration actions that are applicable. Use the customized *z/OS Migration* book to see just the migration actions appropriate for this path.

z/OS 1.10 Migration Actions



General Migration Actions for z/OS R10 from z/OS R8

► Migration Actions You Can Do NOW:


- **Install coexistence and fallback PTFs (Required)** - Verify this programmatically (with FIXCAT IBM.Coexistence.z/OS.V1R10 or using EPSPT extract files)
- **Add or change volumes to keep file systems in a single data set (Recommended)**
 - z/OS version root (and subsystems!) fs are getting larger. One 3390-3 is probably too small.

► Migration Actions Pre-First IPL:

- **Verify that virtual storage limits are set properly (Required)**
 - Default MEMLIMIT changed in z/OS R10 from zero to 2 GB. If you want to use the default (2GB), then you may want to remove your SMFPRMxx setting of MEMLIMIT.
 - The following programs ignore (by default) region and MEMLIMIT values specified through (or affected by) the IEFUSI exit in z/OS R10: HASJES20 (JES2), IATINTK (JES3), and IXGBLF01 (system logger initialization).
 - The NOHONORIEFUSIREGION keyword, new in z/OS V1R10, is set for those programs in the IBM-defined default program properties table (PPT).

6
© 2008 IBM Corporation

z/OS 1.10 Migration Actions



General Migration Actions for z/OS R10 from z/OS R8

Migration Actions Pre-First IPL:

- **Remove references to deleted data sets and path (Required)**
 - DFSORT ISPF panel removal deletes some SYSI.SICE* data sets (may have logon proc impacts)..
- **Add references to new data sets and paths (Required)**
 - New in R10: SDSF data sets (ISF.SISFMOD1 needs to be in the link list and APF-authorized)! ...
 - New in R9: System Rexx data sets SYSI.SAXR* , and Alternate Library for REXX data sets REXX.SEAG*...
- **Accommodate new address spaces (Recommended)**
 - No new address spaces in R10.
 - New in R9: **AXR** (for SysREXX). Automatically started, can be manually restarted. Requires security customization for system address spaces.
 - New in R9: **CEA**, to deliver z/OS events to C-language clients, such as the z/OS CIM server, it automatically started and does not terminate.
 - New in R9: **ARCnXXXX**: One of these DFSMSdss address spaces is started automatically by DFSMSshm whenever a dump, restore, migration, backup, recover, or CDS backup function is invoked. Terminates when hsm terminates.


7
© 2008 IBM Corporation

z/OS 1.10 Migration Actions

General Migration Actions for z/OS R10 from z/OS R8

Migration Actions Pre-First IPL:

- **Update your check customization for modified IBM Health Checker z/OS checks (Recommended)**
 - IBM-changed checks that you've updated, should be reviewed to make sure continue to work as intended.
 - Checks updated in R10: RACF_SENSITIVE_RESOURCES, XCF_CDS_SEPARATION, XCF_CF_STR_PREFLIST, XCF_SFM_ACTIVE, XCF_SIG_PATH_SEPARATION, XCF_SIG_STR_SIZE
- **Accommodate new SCOPE=COMMON data spaces in IEASYSxx MAXCAD (Required-IF)**
 - In R10: One data space for XES. Some XES common storage control blocks were moved to this new data space.
 - In R9: One data space used by common event adapter (CEA) to process messages. This data space is created during CEA initialization.
 - MAXCAD default is 50 (as of z/OS R6).
 - **IBM Health Checker for z/OS can help you determine what to specify for the MAXCAD value!**



8
© 2008 IBM Corporation

General Migration Actions Between z/OS V1 R8 and z/OS V1 R10

These migration actions were taken from *z/OS Migration*. Some descriptions and actions have been shortened for inclusion in this presentation. For the complete descriptions and actions, refer to *z/OS Migration*.

General Migration Actions You Can Do Now

Install coexistence and fallback PTFs (Required)

Migration action: Install coexistence and fallback PTFs on your systems to allow those systems to coexist with z/OS V1R10 systems during your migration, and allow backout from z/OS V1R10 if necessary. See *z/OS Migration* or the handout for the Part 1 for a list of required coexistence PTFs. Using the new SMP/E V3R5 FIXCAT support (the category for coexistence with z/OS R10 is IBM.Coexistence.z/OS.V1R10), or the EPSPT tool with the PSP extract files can make this task much easier.



Add or change volumes to keep your z/OS root file system in a single data set

(Recommended)

Recommended for ease of management if your z/OS root (or subsystem) file system resides on a 3390-3 volume (or another DASD volume that is close to the 3390-3 limit of 3339 cylinders).

Because of enhancements and PTF service, the z/OS root file system (or “version root file system”) and subsystem file systems continue to grow in size from release to release. As of z/OS V1R10, the size of the z/OS root file system, whether HFS or zFS, was approximately 3100 3390 cylinders. This is approaching the 3390-3 size limit of 3339 cylinders. It is advisable to have the z/OS root file system and subsystem file systems within a single data set for ease of management.

Migration action: To keep the z/OS root file system in a single data set, do one of the following:

- Move your z/OS root file system to a larger DASD volume geometry.
- Use multiple volumes for the z/OS root file system data set.

If your z/OS root data set cannot fit on the volume or volumes you have defined for it, divide the z/OS root, with the smaller file systems being managed together. Remember that all systems to which you deploy the z/OS root file system need sufficient DASD space to hold the z/OS root.

Tip: File systems for subsystems and products other than the z/OS product itself might also increase in size. When examining the volume for how much space your z/OS file system is using, check other product file system sizes too.

Use SOFTCAP to identify the effect of capacity changes (Recommended)

Not required, but is recommended to help in assessing processor capacity and available resources when migrating to new software levels, and when migrating to z/Architecture.

Migration action:

- Download SoftCap from one of the following Web sites:
 - Customers: <http://www.ibm.com/support/techdocs/atmastr.nsf/WebIndex/PRS268>
 - Business partners: <http://partners.boulder.ibm.com/src/atmastr.nsf/Web/Techdocs>. Note that this requires an ID on PartnerWorld®. Run SoftCap to determine your expected increase in CPU utilization (if any) and to identify your storage requirements, such as how much storage is needed to IPL.

Reference information: *SoftCap User's Guide*, which is provided with the tool.



Migrate from z/OS.e to z/OS (Required-IF)

Required if you use z/OS.e.

zNALC is replacing New Application License Charges (NALC) and z/OS.e, and is intended to be IBM's strategic z/OS offering for new workloads.

Migration action:

1. **Determine whether your application qualifies for zNALC pricing.**
2. **Decide whether you will implement full-capacity zNALC pricing or subcapacity zNALC pricing.**
3. **Decide which servers and LPARs will be used to run z/OS with zNALC pricing.**
4. **If you will use subcapacity zNALC pricing, submit to IBM an initial subcapacity report containing z/OS zNALC MSUs.**
5. **If you currently run z/OS.e V1R8 and want to run that same level, but with z/OS zNALC pricing, do the following:**
 - a) Obtain your z/OS license with zNALC pricing. Ensure that you order the same optional priced features under z/OS with zNALC pricing that you had with z/OS.e.
 - b) Shut down your z/OS.e system.
 - c) Modify the z/OS.e customization for your system as follows:
 - In parmlib member IFAPRDxx, convert all the enabled z/OS.e statements for ID 5655-G52 to z/OS statements for ID 5694-A01. For example, change all statements like this:

```
PRODUCT OWNER('IBM CORP')
NAME('z/OS')
ID(5655-G52)
VERSION(*) RELEASE(*) MOD(*)
FEATURENAME('z/OS')
STATE(ENABLED)
```

to this:

```
PRODUCT OWNER('IBM CORP')
NAME('z/OS')
ID(5694-A01)
VERSION(*) RELEASE(*) MOD(*)
FEATURENAME('z/OS')
STATE(ENABLED)
```
 - Make the change to identify the z/OS LPAR as zNALC in either of the following ways:
 - o Specify system parameter LICENSE=ZNALC (with any LPAR name).
 - o Specify ZNALxxxx as the LPAR name, along with one of the following:
 - No system parameter specified for LICENSE (which will allow the default of Z/OS)
 - System parameter LICENSE=Z/OS
 - System parameter LICENSE=ZNALC (which is available on z/OS V1R6 and later with APAR OA20314)
 - d) IPL your z/OS system with zNALC pricing.
7. **If you currently run z/OS.e V1R8 and want to change to z/OS with zNALC pricing when migrating to z/OS V1R10, do the following:**
 - a. When ordering your z/OS V1R10 system, specify that you want zNALC pricing. Ensure that you order the same optional priced features under z/OS with zNALC pricing that you had with z/OS.e.
 - b. If sharing parmlib members between your z/OS.e and z/OS zNALC systems, ensure that you are using correct values for both your z/OS.e system and your z/OS zNALC system:
 - In parmlib member IFAPRDxx, your z/OS zNALC system should have statements for ID 5694-A01. There should be no usage of ID 5655-G52, which is for z/OS.e. (**Tip:** If you use the IFAPRDxx sample parmlib

Migrating to z/OS 1.10 - Part 2

member that is shipped with your z/OS V1R10 ServerPac or CBPDO, this will be done for you.) You cannot share your z/OS R10 parmlib with lower level systems, if you still need the z/OS.e specifications. Your z/OS zNALC LPAR should have statements similar to the following:

```
PRODUCT OWNER('IBM CORP')
NAME('z/OS')
ID(5694-A01)
VERSION(*) RELEASE(*) MOD(*)
FEATURENAME('z/OS')
STATE(ENABLED)
```

- Make the change to identify the z/OS LPAR as zNALC in either of the following ways:
 - Specify system parameter LICENSE=ZNALC (with any LPAR name).
 - Specify ZNALxxxx as the LPAR name, along with one of the following:
 - No system parameter specified for LICENSE (which will allow the default of Z/OS)
 - System parameter LICENSE=Z/OS
 - System parameter LICENSE=ZNALC (which is available on z/OS V1R6 and later with APAR OA20314)
- c. IPL your z/OS system with zNALC pricing.

Upgrade Windows 2000, 95, 98, and NT clients (Recommended)

Recommended because z/OS does not support service for client operating systems whose service is withdrawn by the operating system manufacturer.

z/OS no longer supports service for client operating systems whose service is withdrawn by the operating system manufacturer. As a result, IBM no longer supports service for clients running Windows 2000, Windows 95, Windows 98, or Windows NT Workstation 4.xx.

Migration action: Use a supported follow-on to Windows 2000, Windows 95, Windows 98, or Windows NT Workstation 4.xx.

Reference information: For client software supported with z/OS, see *z/OS Planning for Installation*.

General Migration Actions Pre-First IPL

Set up your IPCS environment (Required)

Migration action: Set up an IPCS environment. For guidance, use the documents listed in the reference information below. During setup, ensure that your logon procedure points to the target system's level of IPCS data sets, which are shown in *z/OS Migration*.

Reference information: For more information about IPCS, see *z/OS MVS IPCS Customization*. For more information about the correct logon procedure updates, see the *z/OS Program Directory*. For information about setting up the JES2 IPCS environment, see *z/OS JES2 Diagnosis*. For information about setting up the JES3 IPCS environment, see *z/OS JES3 Diagnosis*.

Use IBM-supplied PARMLIB and PROCLIB (Required)

Migration action: For parmlib, add the data set pointed to by the z/OS V1R10 PARMLIB DDDEF to your parmlib concatenation. The data set should generally be added last in the concatenation, and you should make sure that the other data sets in the concatenation don't have members with the same names as IBM-supplied members. If you place the data set on the system residence volume and use an indirect catalog entry, future migrations won't require this particular migration step.

- For proclib:
 - Ensure that the default proclib members have been copied to your default proclib to pick up the new and changed members. An example of a change is the following: the LLA startup procedure was updated in z/OS V1R10 (and z/OS V1R9 by APAR OA21932) to specify REGION=0M.
 - Update individual sample members provided and ensure they are accessible to the system, as shown in the table of proclib member updates in *z/OS Program Directory*.
 - Ensure that the procedure libraries listed in the table of libraries to be added to the proclib concatenation in *z/OS Program Directory* have been placed in the necessary procedure library concatenations and are available to the system.

Reference information: For lists of parmlib and proclib members that are shipped, see *z/OS Program Directory*.



Migrate /etc and /var system control files (Required)

Migration action: The /etc and /var directories contain system control files: the /etc directory contains customization data that you maintain and the /var directory contains customization data that IBM maintains. During installation, subdirectories of /etc and /var are created. If you install z/OS using ServerPac, some files are loaded into /etc and /var due to the customization performed in ServerPac. You have to merge the files in /etc and /var with those on your previous system. If you install z/OS using CBPDO, you should copy the files from your old system to the z/OS V1R10 /etc and /var subdirectories.

Copy files from your old system to the z/OS V1R10 /etc and /var subdirectories, and then modify the files as necessary to reflect z/OS V1R10 requirements. If you have other files under your existing /var directory, then you will have to merge the old and new files under /var. The easiest way to do this is to create a copy of your current /var HFS and then copy the new /var files into the copy.

The following z/OS V1R10 elements and features use /etc:

- CIM
- Communications Server – IP
- Cryptographic Services – PKI Services and System SSL
- DCE Base Services
- Distributed File Service. The SMB server uses /etc/dfs.
- IBM HTTP Server
- IBM Tivoli Directory Server
- Infoprint Server uses /etc/Printsrv.
- Integrated Security Services - Firewall Technologies, LDAP Server, and Network Authentication Service
- Library Server
- z/OS UNIX System Services

The following z/OS V1R10 elements and features use /var:

- Cryptographic Services – OCSF
- IBM Tivoli Directory Server
- Infoprint Server
- Integrated Security Services - Network Authentication Service uses /var/skrb.

Reference information: For information about copying your existing /etc and /var directories, see *z/OS Migration*.



Verify that virtual storage (MEMLIMIT) is set properly (Required)

Migration action: Determine how much virtual storage use to allow above the 2 GB bar. While there is no practical limit to the number of virtual addresses an address space can request above the bar, the system can limit the amount of virtual storage above the bar that an address space is allowed to use. The amount of virtual storage above the bar is determined as follows. The MEMLIMIT parameter in parmlib member SMFPRMxx sets the default system-wide limit, which defaults to 2 GB as of z/OS V1R10 (and zero prior to z/OS V1R10). However, the system-wide default MEMLIMIT can be overridden by specifying REGION=0M or MEMLIMIT on JOB or EXEC statements in JCL. To set a limit on the use of virtual storage above the bar, use the SMF exit IEFUSI. For more information, see *Limiting the use of memory objects in z/OS MVS Programming: Extended Addressability Guide*.

If you want to control the use of virtual storage above the 2 GB bar, do one or more of the following:

- For MEMLIMIT prior to z/OS V1R10, you must specify a nonzero MEMLIMIT in an active SMFPRMxx member of parmlib to establish a system default other than zero for available virtual storage above 2 GB. (The default MEMLIMIT is zero prior to z/OS V1R10.) As of z/OS V1R10, the MEMLIMIT default is 2 GB. If this 2 GB default value is acceptable to you, no change to SMFPRMxx is necessary.
- You can specify MEMLIMIT explicitly in JCL to override the system default that was set (or allowed to default) in SMFPRMxx.
- You can specify REGION=0M on the job statement in JCL to implicitly set MEMLIMIT to NOLIMIT, which also overrides the system default (from SMFPRMxx).
- You can use IEFUSI both to establish a system default MEMLIMIT for different classes of work (for example, job, TSO, STC) and limit the amount of virtual storage that can be used above the bar, provided that an explicit or implicit nonzero MEMLIMIT is in effect from JCL or SMFPRMxx.

Note: By default, region and MEMLIMIT values specified through, or otherwise affected by, the IEFUSI exit are ignored in z/OS V1R10 for the following programs: HASJES20 (JES2), IATINTK (JES3), and IXGBLF01 (system logger initialization). The reason is that the HONORIEFUSIREGION | NOHONORIEFUSIREGION keyword, new in z/OS V1R10, is set to NOHONORIEFUSIREGION for those programs in the IBM-defined default program properties table (PPT).

Tip: Use IBM Health Checker for z/OS to help determine whether your virtual storage limits are set properly. The check RSM_MEMLIMIT checks the current setting for the MEMLIMIT parameter in SMFPRMxx, which affects the amount of virtual storage above the 2 GB bar that is available to jobs. This check verifies that a nonzero MEMLIMIT value is in use.

Reference information: Information about how to evaluate the central storage configuration can be found in the Washington Systems Center white paper *z/OS Performance: Managing Processor Storage in a 64-bit Environment - V1* at <http://www.ibm.com/support/techdocs> (Search for "WP100269").

Back virtual storage with real and auxiliary storage (Required)

Migration action: As you exploit additional virtual storage by defining additional address spaces or by exploiting memory objects, ensure that you have defined sufficient real and auxiliary storage. Review real storage concentration indicators via an RMF report to evaluate if additional real or auxiliary storage is needed:

- Check UIC and average available frames.
- Check demand page rates.
- Check the percentage of auxiliary slots in use.

Reference information: For more information about memory objects, see *z/OS MVS Programming: Extended Addressability Guide* and Washington Systems Center flash 10165 at <http://www.ibm.com/support/techdocs>. (Search for "flash10165".)



Remove references to deleted data sets and path (Required)

Migration action: Using the table in *z/OS Migration*, "Data sets and paths deleted from z/OS V1R10, and R9" as a guide, remove references to data sets and paths that no longer exist. Remove the references from the following places:

- Parmlib
- Proclib
- Logon procedures (especially the removed z/OS R10 DFSORT libraries, which supported the ISPF panels)
- Catalogs
- Security definitions, including program control definitions
- DFSMS ACS routines
- /etc/profile
- SMP/E DDDEF entry
- Backup and recovery procedures, as well as any references to them In the table, the high-level qualifiers in the data set names are the default qualifiers.

Note: Do not remove any data sets, paths, or references that are needed by earlier-level systems until those systems no longer need them, and you are sure you won't need them for fallback.

Reference information: *z/OS Migration* contains the list of all removed data sets and paths in z/OS R10 and R9.



Add references to new data sets (Required)

Migration action: Using the lists that are found in *z/OS Migration* as a guide, add references in the following places for data sets that have been added to z/OS:

- Parmlib
- Proclib
- Logon procedures
- Catalogs
- Security definitions, including program control definitions
- DFSMS ACS routines

Migrating to z/OS 1.10 - Part 2

- Any backup and recovery procedures. Of special note are data sets:
 - **REXX.SEAGALT** (for Alternate Library for REXX). This data set should be added to your link list (or to your lpa list) if you are not using the IBM Library for REXX. The IBM Library for REXX provides data set REXX.SEAGLPA, which should be put into the lpa list. If REXX.SEAGLPA is put into the lpa list, then REXX.SEAGALT does not need to be added to the link list or lpa list.
 - **ISF.SISFMOD1** (for SDSF). This new data set must be added to the link list and must be APF-authorized.
- Reference information:** *z/OS Migration* contains the list of all new data sets and paths in z/OS R10 and R9.

Accommodate new address spaces (Recommended)

Not required, but recommended to keep interested personnel aware of changes in the system and to ensure that your MAXUSER value in parmlib member IEASYSxx is adequate.

The MAXUSER value in parmlib member IEASYSxx specifies a value that the system uses to limit the number of jobs and started tasks that can run concurrently during a given IPL. You might want to increase your MAXUSER value to take new address spaces into account. (A modest overspecification of MAXUSER should not hurt system performance. The number of total address spaces is the sum of M/S, TS USERS, SYSAS, and INITS. If you change your MAXUSER value, you must re-IPL to make the change effective.)

There are no new address spaces in z/OS V1R10.

The following address spaces were new in z/OS V1R9:

- **ARCnXXXX**: If a SETSYS DSSXMMODE(Y) command is present in parmlib member ARCCMDxx, one of these DFSMSdss address spaces is started automatically by DFSMSHsm whenever a dump, restore, migration, backup, recover, or CDS backup function is invoked. (A DFSMSdss address space is not started for recall tasks.) These DFSMSdss address spaces can reduce the storage used in the DFSMSHsm address space, enabling more tasks to be started within the DFSMSHsm address space. When DFSMSHsm invokes DFSMSdss through the DFSMSdss cross-memory application interface, DFSMSHsm requests that DFSMSdss use a unique address space identifier for each unique DFSMSHsm function and host ID. The address space identifier for each function is in the form ARCnXXXX, where *n* is a unique DFSMSHsm host ID and XXXX is an abbreviation of a DFSMSHsm function. The abbreviations and corresponding functions are: **DUMP** for dump REST for full volume recover, **MIGR** for migration BKUP for backup RCVR for data set recover, **CDSB** for CDS backup. For instance, migration for DFSMSHsm host ID 1 would result in a generated address space identifier of ARC1MIGR. The address space terminates automatically when DFSMSHsm terminates.
Notes: 1. Before ARCnXXXX can be started, it must be added to the installation's designated RACF group for system address spaces.
2. ARCnXXXX function is integrated in z/OS V1R9 but is also available on earlier releases with APAR OA16192 and command PATCH.MCVT.+433 X'FF'.
3. Startup of ARCnXXXX was **always** automatic in z/OS V1R9 until APAR OA23236 introduced the SETSYS DSSXMMODE(Y|N) command, with SETSYS DSSXMMODE(N) being the default.
- **AXR**: This address space is used by the System REXX for z/OS Base component of the BCP. The address space contains the server infrastructure necessary to run REXX execs. REXX work requests originate either from the program interface (AXREXX) or from the operator console (MODIFY AXR). A group of subtasks within the AXR address space processes TSO=N requests; TSO=Y requests are processed in separate address spaces (AXR00-AXR07). The AXR address space is started automatically during z/OS initialization and can be manually restarted. However, before the address space can be started, it must be added to the installation's designated RACF group for system address spaces. For information about how to add it, see *z/OS MVS Programming: Authorized Assembler Services Guide*.
- **CEA**: The common event adapter (CEA) provides the ability to deliver z/OS events to C-language clients, such as the z/OS CIM server. The CEA address space is started automatically during z/OS initialization and does not terminate.

Note: TN3270E Telnet Server was a new address space added in z/OS R6, that is required as of z/OS R9. See migration action "Migration to the TN3270E Telnet server that runs in its own address space".

Rework and install user modifications (Required-IF)

Required if you have made any user modifications that necessitate changes.

Migration action: Use the z/OS SMP/E Planning Migration Assistant to help determine which user modifications need to be reworked and which just have to be reinstalled. The Top or New Intermediate Product Migration Changes Report uses data found on your system, combined with IBM-supplied information from the Software Information Base, to show you the current levels of products available as well as product migration and functional changes using a comparison of FMIDs. You can use this report to determine the product migration impacts by

reviewing the “changed” FMIDs. This can help you assess how many user modifications have to be reworked if you issued the LIST SYSMOD USERMOD FORFMID (listing the “changed” FMIDs) command. All other user modifications can be reinstalled without having to be reworked.

Note: IBM recommends using exit routines for any user modifications where possible, and installing the exit routines with SMP/E. By using SMP/E, it is easier to bring forward desired modifications to the z/OS release you are installing.

Several elements and features have their default options set by assembling and link editing one or more modules. These include:

- XL C/C++
- DFSORT. Consider using ICEPRMxx parmlib members, introduced in z/OS V1R10, to eliminate the assembler language installation option modules
- HLASM
- ISPF (specifically, the ISPF configuration table)
- Language Environment. Investigate using CEEROPT, which can be used to specify run-time options for CICS, IMS LRR, and other LRR users. Even better, consider using the function added in z/OS R7 to eliminate your assembler language run-time option modules in CEEPRMxx parmlib member!

Reconnect subsystems and non-IBM products (Required-IF)

Required if you use any ISV products and need to reconnect them after performing a ServerPac installation, or if you intend to use any subsystems with your z/OS system.

Migration action: Follow the instructions for each ISV product that you use to reconnect it to your z/OS V1R10 ServerPac.

Ensure that any required service is installed prior to using the subsystem with the new z/OS V1R10 system, as well as any required SVCs, system modifications, parmlib setup, and proclib setup. Follow the instructions for the subsystem that you need to reconnect.

Reference information: For a list of independent software vendors (ISVs) that support z/OS, as well as announcements, testimonials, and other information, see <http://www.ibm.com/eserver/zseries/solutions/s390da/>. For a directory of ISV products that support z/OS, see the Global Solutions Directory at <http://www.ibm.com/software/solutions/isv>.

Update operational and other procedures (Required)

Migration action: Review your operation, automation, administration, security, backup, and recovery procedures, and make any necessary changes depending on how you installed and which functions you plan to exploit. Some possible changes are:

- Allowing applicable users access to new high-level qualifiers that you may have. There is a new default high-level qualifier introduced in z/OS R9 with new target data sets – REXX - for Alternate Library for REXX.
- Updating and testing your backup and recovery procedures to accommodate the new target system.
- Updating and testing any disaster recovery procedures.
- Updating and testing any automation procedures to take advantage of new functions.
- Updating security system definitions, such as defining new users and resources, permitting users to use new resources, and defining new profiles in the RACF FACILITY class.

Reference information: For information about the new functions incorporated into z/OS V1R10, see *z/OS Introduction and Release Guide*.

Update your check customization for modified IBM Health Checker for z/OS checks (Recommended)

Recommended to ensure that your checks continue to work as you intend them to work.

Changes that IBM makes to the checks provided by IBM Health Checker for z/OS can affect any updates you might have made. The checks that were changed by IBM in z/OS V1R10 are:

IBMRACF,RACF_SENSITIVE_RESOURCES, IBMXCF,XCF_CDS_SEPARATION,
IBMXCF,XCF_CF_STR_PREFLIST, IBMXCF,XCF_SFM_ACTIVE , IBMXCF,XCF_SIG_PATH_SEPARATION,
IBMXCF,XCF_SIG_STR_SIZE .

Migration action:

1. Look at the updated checks in *IBM Health Checker for z/OS: User's Guide*.
2. Review changes you made for those checks, in HZSPRMxx parmlib members, for example.
3. Make any further updates for the checks to ensure that they continue to work as intended.



Accommodate new SCOPE=COMMON data spaces (Required-IF)

Required if you use the MAXCAD parameter of parmlib member IEASYSxx and the value of MAXCAD specified is inadequate for your z/OS R10 system.

The MAXCAD parameter of parmlib member IEASYSxx specifies the maximum number of SCOPE=COMMON data spaces to be allowed during an IPL. The new SCOPE=COMMON data spaces are:

- Added in z/OS V1R10: One data space used by cross-system extended services (XES). In support of virtual storage constraint relief, new storage management techniques were implemented for cross-system coupling facility (XCF) and XES to reduce the usage of 31-bit common storage below the 2 GB bar. Some XCF common storage control blocks were moved to XCF private storage, and some XES common storage control blocks were moved to a new SCOPE=COMMON data space owned by XES.
- Added in z/OS V1R9: One data space used by common event adapter (CEA) to process messages. This data space is created during CEA initialization.

Your MAXCAD setting must be adequate to accommodate this new data space.

Migration Action: Increase the limit for the number of SCOPE=COMMON data spaces defined on the MAXCAD parameter if your specification is not adequate to cover the new data spaces that have been added. Note that in z/OS V1R6, the MAXCAD default was increased from 25 to 50. If this default is acceptable for your environment, you might want to remove your MAXCAD specification and allow it to default.

Tip: The IBM Health Checker for z/OS can help you determine what to specify for the MAXCAD value. Use the check IBMRSM,RSM_MAXCADS. This check is available as of z/OS V1R7 and also in APAR OA09366 back to z/OS V1R4. By running this check, you can find out:

- The MAXCAD value you specified during IPL
- The number of SCOPE=COMMON data spaces currently in use
- The high water mark, which is the highest usage of SCOPE=COMMON data spaces used during this IPL


Use this information to help you set your MAXCAD specification in IEASYSxx.

General Migration Actions Post-First IPL

<none>

z/OS 1.10 Migration Actions

BCP Migration Actions for z/OS R10 from z/OS R8




Migration Actions You Can Do NOW:

- **Stop using CPU affinity (Required IF, as of R10)**
 - If you have **AFF(xxx)** in your **SCHEDxx**, remove it or change it to **AFF(NONE)**.
 - In programs, look for use of **SCHEDULE** macro where **SRBCPAFF** field is set to a value other than 0 or **X'FFFF'**, and use **IEAMSCHD** macro with **CPUMASK** keyword with an operand other than 0 or **X'FFFF'**. (IBM recommends using **IEAMSCHD** rather than the **SCHEDULE** macro.) Change programs to not require a specific processor on which to run.
 - In programs, look for use of **IWMCPAFN** service. Change programs to not require a specific processor (that is, remove use of **IWMCPAFN**).
 - Any attempt to assign CPU affinity is ignored, if you do not make these changes.
- **Use unique console names and console logon IDs within a GRS ring (Req IF, as of R10)**
 - Consoles (**MCS**, **SMCS**, subsystem, or **EMCS**) have never been allowed to be active on more than one system in a sysplex at one time.
 - As of R10, a new enqueue **SYSZCNZ CONNAME#consolename** is held when active. If enqueue cannot be obtained, console doesn't activate. (Similar enqueue is also obtained when a user logs onto an **MCS** or **SMCS** console – **SYSZCNZ USERID#userid**).
 - In **GRS** ring where the systems in the ring aren't part of the same sysplex, console can only be active on one system.
 - Note that multiple **GRS** complex sharing through 3rd party alternative serialization products will **not** cause problems, because all related **ENQs** are issued with **RNL=NO** to prevent 3rd party management.
 - If you will be affected by the new enqueues, uniquely name the consoles and console logon IDs within the **GRS** ring.

9 © 2008 IBM Corporation

z/OS 1.10 Migration Actions

BCP Migration Actions for z/OS R10 from z/OS R8




Migration Actions Pre-First IPL:

- **Update automation and exits for the modified MCS logon command format (Required-IF, as of R10)**
 - Before R10, there was no mapping of the **LOGON** command format.
 - As of R10, new macro **CNZMYLGN** must be used, and operands must appear in a defined order and location.
- **Discontinue the use of console names LOGON and LOGOFF (Required-IF, as of R10)**
 - Before R10, **LOGON** and **LOGOFF** were valid **MCS**, **SMCS**, **EMCS**, or subsystem console names. As of R10, names are reserved.
- **Modify applications that use unsupported subsystem console functions (Required-IF, as of R10)**
 - As of R10, these functions aren't supported: demand select for an **MCS** console; changing the routing codes for a subsystem console; release. Set or reset master authority can be made only for a subsystem console that is active on the system that makes the request.
 - Use Tracking Facility to help, look "IEAVG700" instances.

10 © 2008 IBM Corporation

z/OS 1.10 Migration Actions

BCP Migration Actions for z/OS R10 from z/OS R8



Migration Actions Pre-First IPL:


- Discontinue use of the DISPLAY CONSOLES,HCONLY command **(Required-IF, as of R10)**
- Update automation to accommodate new responses to the DISPLAY CONSOLES command **(Required-IF, as of R10)**
 - Before R10, IEE889I, IEE890I, or IEE047I were issued.
 - As of R10, new messages CNZ4100I, CNZ4102I, or CNZ4130I are issued.
 - Also, when ACTIVE parm is specified (or defaulted), displays active subsystem consoles (as well as MCS and SMCS, as before), and NACTIVE displays inactive subsystem consoles.
- Remove PPT entries for GRS ENQ/RESERVE/DEQ monitor (Recommended, as of R10)
 - Before R10, you had to ensure the GRS ENQ/RESERVE/DEQ monitor ran at the highest possible dispatching priority.
 - As of R10, monitor address space runs at appropriate dispatching priority. SCHEDxx entries are ignored.
 - Note that in R10, the monitor address space is an extension of the GRS address space, therefore the monitor's CPU time is added to SMF type 30 records for the GRS address space.

11

© 2008 IBM Corporation

z/OS 1.10 Migration Actions

BCP Migration Actions for z/OS R10 from z/OS R8



Migration Actions Pre-First IPL:


- Modify GRS ISV-oriented exit ISGNQXITAQUEUED1 **(Required-IF, as of R10)**
 - As of R10, the parameter list for this exit was changed. If you (or your ISV) is using this exit, modify it to enable handling of the parameter list change.
- Use the new default for VSM ALLOWUSERKEYCSA parameter of DIAGxx (Recommended, as of R9)
 - This parameter allows (YES) or prevents (NO) the allocation of storage in the user key common storage area (CSA).
 - Before R9, default was YES (to allow this allocation).
 - As of R9, default is NO (to prevent this allocation).
 - Investigate any software updates needed to support the default of NO. If you cannot obtain necessary updates for NO, then change default to YES.

12

© 2008 IBM Corporation

z/OS 1.10 Migration Actions

BCP Migration Actions for z/OS R10 from z/OS R8



Migration Actions Pre-First IPL:


- Provide an appropriate HZSPDATA data set on fallback **(Required-IF, as of R10)**
 - As of R10, the format of the HZSPDATA data set has changed. Data written to the HZSPDATA by R10 cannot be read by z/OS R9 or R8.
 - If you fallback from R10 and have run Health Checker on R10, then you have to provide an appropriate HZSPDATA data set upon fallback to R9 or R8.
 - Note that R10 has no problem reading R9 or R8 HZSPDATA, but the reverse isn't true!
 - Make a copy of your current HZSPDATA before moving to R10, and use that copy upon fallback to R9 or R8.

13

© 2008 IBM Corporation

z/OS 1.10 Migration Actions

BCP Migration Actions for z/OS R10 from z/OS R8



Migration Actions Post-First IPL:


- Reallocate the WLM couple data set **(Required-IF, as of R10)**
 - As of R10, the WLM policy structures have increased in size. Your existing WLM cds is probably not large enough for a policy activation on R10.
 - Use R10 libraries (STEPLIB) to allocate a new WLM cds. The values you provide will be used by WLM to calculate the appropriate space required for R10.
- Do not specify VOLUME for DISP=UNCATLG (for dynamic or batch allocation) **(Required-IF, as of R10)**
 - Before R10, data set was uncataloged whether or not it was on the volume specified. So, wrong data set could have been uncataloged.
 - As of R10, when VOLUME and DISP=UNCATLG are given, data set is not uncataloged. This prevents accidental uncataloging of the wrong data set.
 - Remove VOLUME specification, if you have it for DISP=UNCATLG requests.

14

© 2008 IBM Corporation

z/OS 1.10 Migration Actions

BCP Migration Actions for z/OS R10 from z/OS R8



Migration Actions Post-First IPL:

- Update callers of CNZMXURF that do not hold appropriate serialization (**Required-IF, as of R10**)
 - CNZMXURF locates the console control block (UCME) that contains a specific console ID. Applications need to hold serialization while using CNZMXURF and accessing the UCME that it returns.
 - As of R10, if the application does not hold serialization in “distributed” mode, *it can fail*.
 - R10 introduces “distributed” mode (“shared” mode was the old pre-R10 mode, and is the default in R10). “Distributed” mode is where many nice R10 enhancements were introduced (such as 99 MCS, SMCS and subsystem consoles to be active per z/OS image)!
 - Use Tracking Facility to help, look “UCME-CMS” instances. This Tracking Facility support is in R10, and rolled back with OA19948.

15
© 2008 IBM Corporation

BCP Migration Actions Between z/OS V1 R8 and z/OS V1 R10

These migration actions were taken from *z/OS Migration*. Some descriptions and actions have been shortened for inclusion in this presentation. For the complete descriptions and actions, refer to *z/OS Migration*.

BCP Migration Actions You Can Do Now

Evaluate your stand-alone dump data set allocations and your IPCS processing of them (Recommended)

Not required, but recommended because of changes to stand-alone dump processing (that reorder dump records with the intent of recording more important data early), and especially recommended if you deploy any LPARs with significantly more main storage than previously used.

In z/OS V1R6, support was introduced for extended-format sequential data sets, a form of data set that is SMS-managed and can occupy more than 64 K tracks per volume. In z/OS V1R7, this support was supplemented with support for large format sequential data sets (DSNTYPE=LARGE), a form of data set that is essentially the same as conventional sequential data sets except that more than 64 K tracks may be spanned per volume. If your stand-alone dump data sets are spread over more volumes than you want, both types of support can help you gain better control over the number of volumes used for each stand-alone dump data set.

Migration action:

- Use multi-volume stand-alone dump data sets. Adjust the number of volumes and their separation to achieve tolerable stand-alone dump capture times.
- Use extended-format sequential data sets or large format sequential data sets. Copy their contents to an extended format, compressed, striped data set using the IPCS COPYDUMP subcommand prior to analysis. Use the same or a larger striping factor than you used for your stand-alone dump data sets. Dump data sets to which stand-alone dump can write may be neither compressed nor striped, but both attributes are advantageous for the target of the copy operation.
- Use a large CISIZE and striping for IPCS dump directories, and blocking, striping, and compression for the stand-alone dump data set. Very large stand-alone dumps might require that you define your directory with the extended addressing attribute, allowing it to hold more than 4 GB.

Use unique console names and console logon IDs within a GRS ring complex (Req-IF, as of R10)

Required if console names and console logon IDs are not unique within a global resource serialization ring complex where the systems in the ring are not all part of the same sysplex.

Consoles, whether MCS, SMCS, subsystem, or EMCS, have never been allowed to be active on more than one system in a sysplex at one time. Beginning with z/OS V1R10, a new enqueue, SYSZCNZ CONNAME#*consolename*, is held while a console is active. If the enqueue cannot be obtained, the console does not activate. If serialization is managed within a global resource serialization ring complex, the console can only be active on one system, even if the systems are not part of the same sysplex. Attempting to activate the console could result in an abend. A similar enqueue, SYSZCNZ USERID#*userid*, is obtained when a user logs on to an MCS or SMCS console. The same naming restrictions would apply to the logon ID. Note that multiple global resource serialization complex sharing through third-party alternate serialization products will not cause this problem because all the related ENQs are issued with RNL=NO to prevent third-party management.

Migration action: Issue D GRS,LINK to determine whether global resource serialization is managing any CTCs. If it is, do the following:

1. Issue D C,L,S and D EMCS,S from the sysplex and each standalone system in the ring to compare the defined consoles.
2. Uniquely name the consoles on the systems within the global resource serialization ring complex.

Stop using CPU affinity (Required-IF, as of R10)

Required if you use CPU affinity.

IBM has removed support for CPU affinity. CPU affinity originally provided support for assigning work to the same CP. However, this predates Processor Resource/Systems Manager (PR/SM), which introduced logical processors. CPU affinity has long been obsolete for z/OS partitions operating under PR/SM, which makes the CP on which a piece of work executes unpredictable. z/OS V1R9 is the last release to provide support for the assignment of CPU affinity to a logical processor. Beginning with z/OS V1R10, any attempt to assign CPU affinity is ignored. For example, specifications for assigning a program to a specific logical processor or processors using the program properties table (PPT) or a SCHEDxx member of parmlib are ignored.

Migration action:

1. In parmlib member SCHEDxx, look for AFF(xxx) in the PPT section and either remove it or update it to AFF(NONE). Also, update the specified executable program so that it does not have any dependencies on affinity.
2. In programs, look for use of the SCHEDULE macro where the SRBCPAFF field is set to any value other than 0 or X'FFFF', and use of the IEAMSCHD macro with the CPUMASK keyword with an operand other than 0 or X'FFFF'. Change your programs so that they do not require a specific processor on which to run.
3. In programs, look for use of the IWMCPAFN service. Change your programs so that they do not require a specific program on which to run, which has the effect of removing the programs' use of IWMCPAFN.

If you do not make the changes above, the affinity specification is ignored and the result is what it would have been if affinity were not specified.

Migrate from the prelinker to the program management binder (Recommended)

Not required, but recommended because the prelinker is not planned to be enhanced. Enhancements will only be made to the program management binder.

Migration action: Follow the instructions for migrating from the prelinker to binder in *z/OS MVS Program Management: User's Guide and Reference*.

Check for bind jobs that override output record format (Required-IF, as of R9)

Required if you direct the output of the program management binder (SYSLMOD) to a PDS that is not RECFM=U (or explicitly override the record format with RECFM=U).

Before z/OS V1R9, the program management binder (and the linkage editor before it) forced RECFM=U when writing to a PDS. This sometimes caused problems for users when SYSLMOD was accidentally directed to a data set not intended to hold programs.

Beginning with z/OS V1R9, in an attempt to address these problems, the program management binder rejects a request to write to a PDS that already has a record format other than U, unless RECFM=U is provided as an explicit override. As part of this change, any explicitly specified RECFM other than U also causes the program management binder to reject the request. With either rejection, the following occurs:

- For the batch and interactive interfaces, existing message IEW2735S is issued. (The message text is OUTPUT DATA SET FOR DDNAME xxxxxxxx HAS INVALID RECORD FORMAT. RECFM=U IS REQUIRED.) processing stops for the current control statement but the program management binder processes any subsequent control statements. The return code at the end of processing is 12.

Migrating to z/OS 1.10 - Part 2

- For applications calling the program management binder through its API, existing reason code 83000418 is returned from the SAVE request. The program management binder continues to process subsequent requests from the calling application.

Note that the change is only for PDSs, and the new behavior for PDSs matches the current behavior for PDSEs. An equivalent rule has always been in place for PDSEs (DSNTYPE=LIBRARY). Another way of stating the rule, which now applies to both PDSs and PDSEs, is that the program management binder will only write the program if the merge of data set and JCL information has either no record format at all or record format U. The merge is done with JCL information overriding data set information.

Migration action:

- If Check for jobs that specify a RECFM other than U for SYSLMOD. Either remove the RECFM parameter or change it to U. If the same data sets are used to contain data in other record formats, store the data in separate data sets.
- Check for jobs that allocate SYSLMOD to a suspicious destination, such as a maclib or proclib. Modify the destination data set to something more appropriate.

Modify programs that use the program management binder fast data access service (Required-IF, as of R9)

Required if (1) you run any programs that do not use variable length parameter lists for the fast data access service or (2) your system does not make available to your program enough above-the-bar storage for the fast data access service to hold program objects for processing.

The fast data access service of the program management binder has been rewritten for z/OS V1R9. The service is functionally compatible with previous releases except for the following:

- The service now uses 64-bit storage rather than a data space. If MEMLIMIT is zero or very small, the service might display a message in the job log and return error code 12 to the caller.
- The documentation has always indicated that variable length parameter lists must be used, with the high bit being set in the last parameter. However, the program management binder code in z/OS V1R5 through V1R8 did not enforce this requirement for the request code call interface introduced in z/OS V1R5. The requirement is now enforced by default. (You can override the default.)

For most processing, the fast data access service must be able to create a 64-bit memory object large enough to hold the program object as stored on DASD. Program objects could in theory be up to 2 GB in size, though they rarely exceed 100 MB and most are smaller than 1 MB. If the program cannot obtain the 64-bit memory it requires, it displays the following message in the job log (WTO ROUTCODE=11): IEW2091S IEWBFDAT UNABLE TO GET *nnnn* MB OF 64-BIT STORAGE, IARV64 CODE *xxxx*.

If the fast data access service is called to obtain information about a program object in a system data set (normally one in the SYS1.LINKLIB or SYS1.SVCLIB concatenation), it might now be forced to obtain space to hold the program object in 31-bit storage within the caller's address space. Previously, this space would have been obtained by creating a data space. This change could cause a storage constraint if very large program objects are in the SYS1.LINKLIB or SYS1.SVCLIB concatenation, and if the fast data access service is called to provide information about them.

Migration action:

- Assume that some applications might need to use 64-bit storage. Code the IEFUSI installation exit to allow such usage.
- Consider changing SMF defaults to permit storage above the 2 GB bar. If the SMF default is not changed, ensure that a MEMLIMIT parameter is added in jobs that use the fast data access service.
- If locally developed programs use the fast data access service, ask the developers to verify their parameter list format.
- Use the IEWBQHOB bypass temporarily, if needed, to avoid disruption.
- Make sure that ISV software that you are using has been tested with z/OS V1R10. The fast data access service might be used internally by these products. Typical products that use it are debuggers, performance analyzers, and compliance analysis programs.

BCP Migration Actions Pre-First IPL

Create IPL text (Required)

Migration action: Update and run the IPLTEXT job to write a new copy of the IPL text. If you install z/OS with a ServerPac, an installation dialog job is provided to perform this action. If you install z/OS with a CBPDO, instructions to perform this action are provided in *z/OS Program Directory*.

Reference information: For a sample IPLTEXT job, see *z/OS Program Directory*. ServerPac provides a similar job for accomplishing this task; see *ServerPac: Installing Your Order*.

Reassemble the stand-alone dump program (Required)

Migration action: Reassemble the stand-alone dump program. If you install z/OS with a ServerPac, an installation dialog job is provided to perform this action. If you install z/OS with a CBPDO, instructions to perform this action are provided in *z/OS Program Directory*. Once the stand-alone dump program is properly created on a DASD residence volume, it resides in the SYS1.PAGEDUMP.Vvolser data set.

Reference information: *ServerPac: Installing Your Order*, *z/OS Program Directory*, and *z/OS MVS Diagnosis: Tools and Service Aids*

Update automation and exits for the modified MCS logon command forum (Required-IF as of R10)

Required if the MCS LOGON command is changed by automation or exit routines.

CNZMYLGN is a new macro in z/OS V1R10. Automation and command exits that modify MCS LOGON commands must use the format mapped by the CNZMYLGN macro. Before z/OS V1R10, there was no mapping of the LOGON command format, so programs scanning the command and making modifications could reorder the command's operands. With z/OS V1R10, the operands must appear in a defined order and location. CNZMYLGN maps these locations.

Migration action: Change automation or command exits that modify the MCS LOGON command to ensure that the format mapped by CNZMYLGN is used.

Discontinue the use of console names LOGON and LOGOFF (Required-IF as of R10)

Required if a console name of LOGON or LOGOFF is used.

Before z/OS V1R10, LOGON and LOGOFF were accepted as valid console names for an MCS, SMCS, EMCS, or subsystem console. As of z/OS V1R10, LOGON and LOGOFF are not acceptable as console names. Queries of a console name of LOGON or LOGOFF by using the CONVCON or CnzConv service will indicate that the console name is reserved.

Migration action:

- Change your CONSOLxx parmlib members to remove use of console names LOGON and LOGOFF.
- Change any SEND commands that have a target console name of LOGON or LOGOFF.
- Change your code that invokes MCSOPER with consoles names of LOGON or LOGOFF.
- Change your code that invokes CnzConv or CONVCON with console names of LOGON or LOGOFF.

Accommodate VARY command changes (Required-IF as of R10)

Required if COMMNDxx or IEACMDxx parmlib members contain the VARY attribute, VARY CN(),OFFLINE with attribute, or VARY CN(),LU='NONE' commands, or if automation programs or exit routines issue MGCRC macros that execute a VARY attribute or VARY CN(),OFFLINE with attribute command.

- VARY command attribute processing is changed in z/OS V1R10 to no longer allow attributes to be altered for inactive MCS, SMCS, and subsystem consoles. There are two exceptions: the LU and LOGON attributes can be changed for inactive SMCS consoles. Before z/OS V1R10, VARY processing allowed all attributes to be changed for inactive MCS, SMCS, and subsystem consoles. Only the attributes for inactive EMCS consoles could not be changed. Beginning with z/OS V1R10, VARY command attribute processing no longer allows attributes to be altered for inactive MCS, SMCS, and subsystem consoles (as it does for inactive EMCS consoles). The only exceptions are the LU and LOGON attributes for inactive SMCS consoles. If a VARY command requests a change to an SMCS console attribute that is not LU or LOGON, message IEE871I CN=consname NOT ACTIVE. CAN NOT CHANGE CONSOLE ATTRIBUTES is issued.
- The VARY CN(),OFFLINE command is changed in z/OS V1R10 to no longer allow attributes to be altered when specified with the OFFLINE keyword. OFFLINE processing will still deactivate the console (except if the LU or LOGON keywords are specified) but the attributes will not be changed. Before z/OS V1R10, the VARY CN(),OFFLINE command allowed all attributes to be specified and their attributes changed along with the console being deactivated. Only the attributes for inactive EMCS consoles could not be changed. In z/OS V1R10, a VARY CN(),OFFLINE command with attributes specified is rejected, as follows:
 - If the command specifies either the LU or LOGON attribute, the console is not deactivated and message ASA103I is issued indicating that the LU or LOGON keyword cannot be specified with the OFFLINE keyword.
 - If the command specifies other attributes, the console is deactivated and the following message is issued: IEE871I CN=consname NOT ACTIVE. CAN NOT CHANGE CONSOLE ATTRIBUTES.

The reason there are two different messages (ASA103I and IEE871I) is that the LU and LOGON attributes are the only attributes that can be changed for inactive SMCS consoles, and it is necessary to differentiate these attributes from the rest of the console attributes (ROUT, INTIDS, MSCOPE, and so forth). Message

IEE8711 does not apply to LU and LOGON because those attributes can be changed when the console is not active.

- VARY command attribute processing is changed in z/OS V1R10 to allow an SMCS console to change its LU attribute to *NONE* independent of the SMCS console's LOGON attribute.

Before z/OS V1R10, VARY processing required that the LOGON keyword be specified when the LU=*NONE* attribute was specified. The exception was when the SMCS console already had the LOGON=REQUIRED attribute, changing the LU to *NONE* would not require the LOGON attribute to be specified. In z/OS V1R10, it does not matter what the LOGON attribute is for an SMCS console. The LU attribute can be changed to *NONE* anytime. Note that having the LOGON attribute specified with the LU=*NONE* attribute still works; the LOGON attribute is just no longer required.

Migration action:

- Check if the consoles' attributes really have to change. If they do, ensure that the consoles are active before changing their attributes. Do not issue attribute changes on a VARY CN(),OFFLINE command. If that command is required, first issue the VARY attribute command while the console is active, and then deactivate the console with a VARY CN(),OFFLINE command.
- Check the CONSOLxx parmlib member and verify that the console attributes are set correctly so that the consoles do not need to be changed by using the VARY command.
- Check if automation programs and exit routines issue MGCRC macros that execute a VARY attribute or VARY CN(),OFFLINE with attribute command.
- Notify system operators about the VARY command changes.
- If LU=*NONE* is specified, remove the LOGON attribute because it no longer needs to be specified. The LOGON attribute is still processed but it is no longer necessary.



Modify applications that use unsupported subsystem console functions (Required-IF as of

R10)

Required if subsystem consoles are being used.

Before z/OS V1R10, the following subsystem console requests were supported:

- A demand select request for an MCS console as well as a subsystem console
- A request to change the routing codes for a subsystem console
- A request to release, set master authority, or reset master authority for a subsystem console that is active on another system.

As of z/OS V1R10, the functions are no longer supported, as follows:

- A demand select request can no longer be made for an MCS console. Also, the SCSUNIT and SCSUNIT4 fields in the IEZVG100 parameter list must not be specified, meaning they must be zero.
- A request to change the routing codes for a subsystem console is ignored and returned with return code zero (assuming no other errors are detected).
- A request to release, set master authority, or reset master authority can be made only for a subsystem console that is active on the system that makes the request.

Migration action:

- Request for demand select:
 - – *What to look for:* Invocations of IEAVG700 in programs that run on z/OS V1R6 and later where the subsystem console service routine parameter list (SCSR as mapped by IEZVG100) flag SCSDEMSL is being set and the console being requested is not a subsystem console, or where fields SCSUNIT or SCSUNIT4 are being used.
 - – *What to do:* Change the request to be for a subsystem console and do not set fields SCSUNIT or SCSUNIT4. Note that while it is necessary to make this change for programs that run on z/OS V1R8 and later with the PTF for APAR OA18204 installed, it is acceptable and advisable to change all such programs regardless of release level and regardless of whether the PTF is installed.
- Request to change routing codes:
 - – *What to look for:* Invocations of IEAVG700 in programs that will run on z/OS V1R10 and later where the subsystem console service routine parameter list (SCSR as mapped by IEZVG100) flag SCSRTCDF or field SCSRTCDP is being set.
 - – *What to do:* It is not necessary to make any changes because requests to change routing codes are accepted but not processed beginning with z/OS V1R10. It is, however, advisable to examine those programs to ensure that they will continue to function correctly when their requests to change routing codes are accepted but not processed. It is advisable that such an examination be done for all such programs regardless of release level.

- Request to release, set master authority, or reset master authority on nonowning system:
 - – *What to look for:* Invocations of IEAVG700 in programs that run on z/OS V1R8 and later where the subsystem console service routine parameter list (SCSR as mapped by IEZVG100) flags SCSRELSE, SCSPMSTR, or SCSNMSTR are being set and the request is for a subsystem console that is owned on some other system.
 - – *What to do:* Change the invocation so that it is executed on the system where the console is owned. Note that while it is necessary to make this change for programs that run on z/OS V1R8 and later with the PTF for APAR 0A18204 installed, it is acceptable and advisable to change all such programs regardless of release level and regardless of whether the PTF is installed.

You can use the Tracking Facility to help with this migration action. In Tracking Facility output, look for violations that start with "IEAVG700".

Discontinue use of the DISPLAY CONSOLES,HCONLY command (Required-IF as of R10)

Required if you have code that either issues DISPLAY CONSOLES,HCONLY or looks for the command response from DISPLAY CONSOLES,HCONLY.

Before z/OS V1R10, specifying the DISPLAY CONSOLES,HCONLY command resulted in the reporting of message and SYNCH loss. As of z/OS V1R10, specifying the DISPLAY CONSOLES,HCONLY command results in message IEE535I being issued to indicate that an invalid DISPLAY parameter was specified.

Migration action: Do not issue the DISPLAY CONSOLES,HCONLY command anymore. Update any programs that automate on the command response to the DISPLAY CONSOLES,HCONLY command to no longer automate on it.



Update automation to accommodate new responses to the DISPLAY CONSOLES command (Required-IF as of R10)

Required if you have code that looks for message IEE889I, IEE890I, or IEE047I.

Before z/OS V1R10, specifying the DISPLAY CONSOLES command resulted in the issuance of message IEE889I, IEE890I, or IEE047I. In addition, all subsystem consoles were displayed when the NACTIVE parameter was specified. No subsystem consoles were displayed when the ACTIVE parameter was specified (or defaulted). As of z/OS V1R10, specifying the DISPLAY CONSOLES command results in the issuance of new message CNZ4100I, CNZ4102I, or CNZ4103I. In addition, the DISPLAY CONSOLES system command displays active subsystem consoles when the ACTIVE parameter is specified (or defaulted) and inactive subsystem consoles when the NACTIVE parameter is specified.

Migration action: Update any programs that automate on message IEE889I, IEE890I, or IEE047I to automate on CNZ4100I, CNZ4102I, and CNZ4103I instead. Also take into account the change in behavior for subsystem consoles.

Remove PPT entries for the GRS ENQ/RESERVE/DEQ monitor (Recommended, as of R10)

Not required, but recommended if you have any PPT entries for the GRS ENQ/RESERVE/DEQ monitor to avoid possible confusion in the future.

Before z/OS V1R10, users of the global resource serialization ENQ/RESERVE/DEQ monitor were required to ensure that its address space ran at the highest possible dispatching priority. Beginning with z/OS V1R10, the monitor address space is run at the appropriate dispatching priority. Therefore, any program properties table (PPT) entries (defined in parmlib member SCHEDxx) that are related to the monitor's dispatching priority (specifically, the PRIV keyword) are no longer required and can be removed, as they could be misleading. There is no longer any need to have a PPT entry for the monitor. The suggested program name (PGMNAME in the PPT) for the monitor was ISGAUDIT but you could change this name.

Note that, because this change makes the monitor address space an extension of the global resource serialization address space, the monitor address space's CPU time is added to SMF type 30 records for the global resource serialization address space.

Migration action: In parmlib member SCHEDxx, remove any PRIV|NOPRIV keyword that is specified for the global resource serialization ENQ/RESERVE/DEQ monitor. If you do not remove the keyword, it will be ignored, but not removing it could confuse someone who sees it in the future and does not realize that it is no longer functional. If a SYST keyword was also specified to ensure that the monitor would not time out, specify TIME=NOLIMIT on the JOB statement instead because TIME=NOLIMIT pertains to all tasks in the address space.

Modify GRS ISV-oriented exit ISGNQXITQUEUED1 (Required-IF as of R10)

Required if your software or any of your ISV software uses the ISGNQXITQUEUED1 installation exit.

In z/OS V1R10, the parameter list for installation exit ISGNQXITQUEUED1 was changed. The NqqpRsc_ReturnCode in the NQQP now contains the ISGENQ return code for ISGENQ requests, and the ENQ return code for ENQ/RESERVE requests. Before z/OS V1R10, the field contained an ENQ/RESERVE style return code for both ISGENQ and ENQ/RESERVE requests.

Migration action:

1. Determine whether the ISGNQXITQUEUED1 installation exit is being used on your system by using the DISPLAY PROG,EXIT command. The target module names can help identify who the exploiters are.
2. If you are using the exit, do the following:
 - If any of the exit routines are owned by ISVs that you did not contact for new z/OS V1R10 support, contact those ISVs to ensure that you have the latest updates.
 - If any of the exit routines are owned by your installation, ensure that these exit routines have been modified to enable handling of the parameter list change.

Failure to modify ISGNQXITQUEUED1 could cause a data integrity problem or system failure.

Update automation that handles message IEE174I (Required-IF as of R10)

Required if you have automation routines that examine the message.

The content of message IEE174I has changed. Before z/OS V1R10, the message might contain the following information:

```
ID CPU VF CR SERIAL
cpuid s s s ser_no
cpuid s s s ser_no
```

```
CPC ND = tttttt.nnn.mmm.pp.ssssssssss
CPC SI = tttt.nnnnnnnnnnnnnnnn.mmmmmmmmmmmmmmmmm.pppp.ssssssssssssssss
```

Starting in z/OS V1R10, the message might contain an additional line as follows:

```
ID CPU VF CR SERIAL
cpuid s s s ser_no
cpuid s s s ser_no
```

```
CPC ND = tttttt.nnn.mmm.pp.ssssssssss
CPC SI = tttt.nnnnnnnnnnnnnnnn.mmmmmmmmmmmmmmmmm.pppp.ssssssssssssssss
      Model: oooooooooooooooooo
```

Use the default for MEMLIMIT (Recommended, as of R10)

Not required, but recommended so that there is some nonzero MEMLIMIT, by default, on the amount of virtual storage above the 2 GB bar for an address space.

Your MEMLIMIT value affects the virtual storage limit for a single address space above the 2 GB bar. Before z/OS V1R10, the MEMLIMIT default value was zero (meaning that the MEMLIMIT default was no virtual storage available above the 2 GB bar). As of z/OS V1R10, the MEMLIMIT default value has changed to 2 GB. This changed default provides a more usable value.

Migration action:

- To accept this new behavior, do nothing.
- If you wish to revert back to the prior MEMLIMIT default of zero, override the MEMLIMIT default to specify 0. But realize that a nonzero MEMLIMIT value is recommended, which is why the default value was changed in z/OS V1R10.

Tip: Use IBM Health Checker for z/OS to help determine whether your virtual storage limits are set properly. The check RSM_MEMPLIMIT checks the current setting for the MEMLIMIT parameter in SMFPRMxx, which affects the amount of virtual storage above the 2 GB bar that is available to jobs. This check verifies that a nonzero MEMLIMIT value is in use.



Provide an appropriate HZSPDATA data set on fallback (Required-IF as of R10)

Required if you use IBM Health Checker for z/OS and you fall back from z/OS V1R10 to z/OS V1R9 or z/OS V1R8.

In z/OS V1R10, the format of data written to the HZSPDATA data set used by IBM Health Checker for z/OS has changed. Data written to HZSPDATA in z/OS V1R10 cannot be read by z/OS V1R9 or z/OS V1R8. Thus, if you fall back from z/OS V1R10 to z/OS V1R9 or z/OS V1R8 after you have started IBM Health Checker for z/OS on z/OS V1R10, you have to provide an appropriate HZSPDATA data set before you can use it on z/OS V1R9 or z/OS V1R8 because z/OS V1R10 formats HZSPDATA from the older format to the newer (z/OS V1R10) format.

Migrating to z/OS 1.10 - Part 2

Note that the z/OS V1R10 version of IBM Health Checker for z/OS has no problem reading HZSPDATA data that was written on a z/OS V1R9 or z/OS V1R8 system.

Once you use the HZSPDATA data set on z/OS V1R10, you cannot use it on z/OS V1R9 or z/OS V1R8. You must either provide a different data set (presumably a copy of the original z/OS V1R9 or z/OS V1R8 HZSPDATA) or reallocate (and lose the data in) the original data set.

Migration action: Make a copy of your current z/OS V1R9 or z/OS V1R8 HZSPDATA data set. Upon fallback to z/OS V1R9 or z/OS V1R8, specify that copy of HZSPDATA on your IBM Health Checker for z/OS started procedure. If you do not provide an appropriate HZSPDATA data set, message HZS0015E or HZS0013A is issued to indicate that the format of the current HZSPDATA data set is invalid. You will be prompted for specification of an HZSPDATA data set so that a check can write its persistent data.

Modify JCL to be consistent with a new coding rule for the system symbols (Required-IF as of R9)

Required if you depend on JCL coding that is affected.

You can use system symbols in started task JCL (for both jobs and procedures) and in TSO logon procedures. As of z/OS V1R9, support was expanded to allow substrings of system symbols. However, regardless of implementing the new substrings functionality, if you code a system symbol in a data set name with a relative generation number, and the system symbol immediately precedes the generation number and the generation number is not preceded by a plus or minus sign, you might need to change the JCL. For example:

- Case 1: DD DSN=TSOUSER.EREP&SYSNAME(+0),DISP=SHR
- Case 2: DD DSN=TSOUSER.EREP&SYSNAME(0),DISP=SHR

In Case 1, JCL substitution continues to be successful even in z/OS V1R9. On the other hand, in Case 2, resolution to an expected data set name will not occur, resulting in a JCL error.

Migration action: For coding a system symbol in a data set name with a relative generation number, if the system symbol immediately precedes the generation number and the generation number is not preceded by a plus or minus sign, place a period between the system symbol and the generation number. For example, if &SYSNAME were to resolve to SY1, code DSN=PROD.&SYSNAME.(0) when the desired data set name is PROD.SY1(0). The period is optional if a plus or minus sign is included: DSN=PROD.&SYSNAME.(+0) or DSN=PROD.&SYSNAME(+0).

Update automation that handles message IOS551I (Required-IF as of R9, and rollback via OA19445)

Required if you have automation that examines the message text.

If a VARY dddd,ONLINE command is issued on one system while the device is reserved on another system, the VARY command fails and the following commands are issued:

```
IEE103I UNIT dddd NOT BROUGHT ONLINE
IEE763I NAME= IECVIOPM CODE= 0000000400000000
IOS551I NO PATHS PHYSICALLY AVAILABLE
IEE764I END OF IEE103I RELATED MESSAGES
```

Message IOS551I can be misleading because it suggests there is something physically wrong with the device or paths. As of z/OS V1R9, and APAR OA19445 on z/OS V1R8 and z/OS V1R7, path validation processing is changed to issue the following message instead of IOS551I if all paths fail and at least one path failure was caused by I/O timeout: IOS558I FUNCTION NOT PERFORMED, I/O TIMED OUT.

Migration action: Change your automation to expect the new message text. Also, tell operators about the improved message text. **Note:** One possible cause for message IOS558I is that the device is shared and is reserved by a system other than the processing system. As an operator response, look for a reserve held on the device by a sharing system. If one exists, wait for the reserve to be released on the sharing system before trying the VARY command again.

Accommodate the changed DAE interval for suppressing duplicate dumps (Required-IF as of R9 with OA20972)

Required if you want to retain the historical information recorded in the SYS1.DAE data set for 180 days, which was the previous default.

The dump analysis and elimination (DAE) interval for suppressing duplicate dumps was changed from 180 days to 60 days to improve first failure data capture. This allows duplicate dumps for recurring problems to be suppressed within that interval, and allows another dump to be taken if the problem persists or if a different problem occurs. The change of interval was introduced by APAR OW51284 in 2001 and was integrated in z/OS V1R3. However, the command ADYUPDAT CLEANUP nnn, which could be used to delete entries older than some period, was not

changed, and its default value (*nnn*) remained at 180 days. APAR OA20972 on z/OS V1R9 changes the default value for DAE CLIST ADYUPDAT to 60 days to match the DAE interval.

Migration action: The default value in the ADYUPDAT CLEANUP CLIST does not need to match the DAE interval after which entries are ignored. So, you might want to retain the historical information recorded in the SYS1.DAE data set for longer than 60 days (new default). In such a situation, if you would like to continue using the previous default value (180 days), you need to specify it explicitly, as follows: ADYUPDAT CLEANUP 180.

Ensure that CLOCKxx is specified properly for LOCAL timing mode (Required-IF as of R9 and via OA13344)

Required if you are running in LOCAL timing mode, that is, ETRMODE NO or ETRMODE YES but a Sysplex Timer is not available.

A hardware feature of z10, z9, z990, and z890 servers known as Server Time Protocol (STP), a follow-on to the Sysplex Timer (9037-002), can be enabled using the new STPMODE parameter in the CLOCKxx member of parmlib. APAR OA13344 on z/OS V1R8 and V1R7 enables the STP functionality, with a default value of STPMODE YES. Systems for which ETRMODE NO is specified, without an explicit specification of STPMODE, receive the default of STPMODE YES. Because the server is not in STP timing mode but in LOCAL timing mode in this case, messages IEA888A and IEA381I are issued during system initialization. (IEA381I is THE STP FACILITY IS NOT USABLE. SYSTEM CONTINUES IN LOCAL TIMING MODE.) Note that operator prompt IEA888A is issued regardless of whether OPERATOR PROMPT or NOPROMPT is specified in CLOCKxx.

Migration action: If you are not using STP timing mode, that is, if you are running in LOCAL timing mode (which means ETRMODE NO or ETRMODE YES but a Sysplex Timer is not available), and you do not want to reply to the clock prompt (IEA888A) at each IPL, specify STPMODE NO explicitly in the CLOCKxx parmlib member. This action applies to z/OS systems running on servers that are not STP-capable (z900 and z800) and also servers that are STP-capable (z10, z9, z990, and z890) but not STP configured.

Note: If the z/OS system image is running with SIMETRID in the CLOCKxx parmlib member, this migration action is not relevant because ETRMODE and STPMODE are ignored when SIMETRID is specified.

Modify applications to accommodate the changed MEMLIMIT value (Required-IF as of R9 and rolled back in OA14391)

Required if you use IEFUSI and set REGION but not MEMLIMIT, and have applications that are affected by the way MEMLIMIT is now set.

Before APAR OA14391 for z/OS V1R8 and V1R7, if a job specified MEMLIMIT and REGION=0 in the JCL, and if IEFUSI was used to limit REGION but not MEMLIMIT, the MEMLIMIT value specified in the JCL was not used. Instead, the MEMLIMIT for the job was set equal to the REGION size. Beginning with z/OS V1R9 (and z/OS V1R8 and V1R7 with APAR OA14391), this scenario results in a setting of MEMLIMIT equal to the value specified in the JOB or EXEC statement of the job's JCL.

Migration action: Examine applications that depend on the MEMLIMIT setting and, if the new behavior is unacceptable, code an appropriate MEMLIMIT value for these applications, either in your JCL for the job or in your IEFUSI exit.

Note: If no explicit value for MEMLIMIT is specified in the JCL but REGION=0 is specified, and if IEFUSI is subsequently used to limit REGION but not establish a MEMLIMIT, both of the following occur:

- The MEMLIMIT for the job is set equal to the IEFUSI-specified REGION value. (This behavior has been in effect since z/OS V1R7.)
- A system SMF-specified value for MEMLIMIT is ignored and, if the PTF for APAR OA14391 is applied, SMF30MES (the source of MEMLIMIT in the SMF30 record) is set to X'0A'.



Ensure that the GRS ISV-oriented exit routines run in a cross-memory environment (Required-IF, as of R9)

Required if your software or any of your ISV software uses the ISGNQXITBATCH, ISGNQXITBATCHCND, ISGNQXITPREBATCH, or ISGNQXITQUEUED1 installation exit.

Before z/OS V1R9, the global resource serialization exit routines ISGNQXITBATCH, ISGNQXITBATCHCND, ISGNQXITPREBATCH, and ISGNQXITQUEUED1, used by independent software vendors (ISVs), were not invoked in cross-memory mode. Starting in z/OS V1R9, these exit routines are invoked in cross-memory mode. If your installation uses any of these exit routines or if any of your ISV software uses these exit routines, you must ensure that the

exit routines can run in a cross-memory environment prior to implementing z/OS V1R9. A failure in any of these exit routines could cause a data integrity problem or system failure.

Migration action: Determine whether any of the affected exit routines are currently in use on your system by using the DISPLAY PROG,EXIT command. The target module names can help identify who the exploiters are.

If you are using any of the affected exit routines, do the following:

- If any of the exit routines are owned by ISVs that you did not contact for new z/OS V1R9 support, contact those ISVs to ensure that you have the latest updates.
- If any of the exit routines are owned by your installation, ensure that these exit routines have been modified to enable execution in a cross-memory environment.



Use the new default for the VSM ALLOWUSERKEYCSA parameter of DIAGxx

(Recommended)

Not required, but recommended for better security.

The VSM ALLOWUSERKEYCSA parameter of parmlib member DIAGxx, which was introduced in z/OS V1R8, either prevents (NO) or allows (YES) the allocation of storage in the user key (8-F) common storage area (CSA). Before z/OS V1R9, the default value was YES. Beginning with z/OS V1R9, the default value is NO. Specifying or defaulting to NO is recommended. YES allows user key CSA to be allocated, which can be a security risk because any unauthorized program can modify it.

When VSM ALLOWUSERKEYCSA is NO, any attempt to obtain user key from a CSA subpool (through GETMAIN or STORAGE OBTAIN) will result in a failure with a B04-5C, B0A-5C, or B78-5C abend.

Migration action:

1. If you use the default value for the VSM ALLOWUSERKEYCSA parameter, accept the new default value of NO because it prevents unauthorized programs from modifying user key CSA. However, if you prefer the previous default value of YES, specify it on VSM ALLOWUSERKEYCSA in parmlib member DIAGxx.
2. Investigate any required software updates on your target system to support the default value of NO.

Tip: Use IBM Health Checker for z/OS to check the value of the VSM ALLOWUSERKEYCSA parameter. The check VSM_ALLOWUSERKEYCSA examines the setting of the ALLOWUSERKEYCSA(YES|NO) DIAGxx option and compares it to the IBM recommended setting of ALLOWUSERKEYCSA(NO). A warning is issued if the setting is YES. Note that APAR OA22166 on z/OS V1R9 changed IBM Health Checker for z/OS so that the VSM_ALLOWUSERKEYCSA check is active.

Use the new default for the SRM invocation interval (RMPTTOM) with very fast processors **(Recommended, as of R9 and in OA18452)**

Not required, but recommended so that SRM timed algorithms are not called more often than necessary on systems with a uniprocessor speed of greater than 100 MIPS.

The RMPTTOM parameter in parmlib member IEAOPTxx specifies the system resources manager (SRM) invocation interval. Before z/OS V1R9 (and APAR OA18452 on prior releases), the RMPTTOM default value was 1000 (milliseconds) for all processors. Beginning with z/OS V1R9 (or V1R8 and V1R7 with APAR OA18452), the default value depends on processor speed. For systems with a uniprocessor speed of 100 MIPS or less, the default value remains 1000. But for systems with a uniprocessor speed of greater than 100 MIPS, the default value is now 3000.

The default value of 3000 with very fast processors is more reasonable than the previous default value of 1000. 1000 could cause some SRM timed algorithms to be called more often than necessary. For example, determining the mean time to wait and the resulting dispatching priority of an address space on the order of every 1000 milliseconds has little value compared to doing the same every 3000 milliseconds.

Migration action: If you use the default value for the RMPTTOM parameter, and your processor speed is greater than 100 MIPS, accept the new default value of 3000 because it is more reasonable with very fast processors. However, if you prefer the previous default value of 1000 (or any other value), specify it on RMPTTOM in parmlib member IEAOPTxx.



Discontinue use of the DIAGxx ISGERQA parameter (Required, as of R9)

Before z/OS V1R9, ISGERQA was a parameter of parmlib member DIAGxx that could be used under the direction of IBM Service to change the amount of private virtual storage below the 2 GB bar that global resource serialization would reserve for the extended resource queue area (ERQA). The ERQA was used mostly for ENQ-related control blocks and it still exists in z/OS V1R9.

Beginning with z/OS V1R9, ISGERQA is not to be used anymore because of the enhanced ENQ capacity of global resource serialization. Most of the storage that was in the ERQA is now above the 2 GB bar in the global resource serialization private area and is not controlled by the ISGERQA parameter. z/OS V1R9 can place higher demands on other parts of global resource serialization private storage below the 2 GB bar, and having an unnecessarily high ERQA can reduce the amount of storage available to it.

Migration action: Do not use the ISGERQA parameter of parmlib member DIAGxx.

BCP Migration Actions Post-First IPL

Do not specify volume information on allocations intended to uncatalog a data set (Required-IF as of R10)

Required if you have allocation requests that uncatalog data sets.

Before z/OS V1R10, a data set defined with the parameters VOLUME and DISP=UNCATLG was uncataloged when a step or job terminated (either normally or abnormally), whether or not the data set was on the volume specified by VOLUME. Consequently, the wrong data set could be uncataloged. Starting with z/OS V1R10, a data set defined with the parameters VOLUME and DISP=UNCATLG is not uncataloged on step or job termination. Instead, message IEF287I *dsn* NOT UNCATLGD 13 is issued. This change prevents a user from accidentally uncataloging the wrong data set.

Note that this change applies to DISP processing in dynamic allocation as well as batch allocation, that is, the data set will not be uncataloged if the dynamic or batch request to uncatalog specifies VOLUME information.

Migration action: Remove the volume (VOLUME) specification from data set allocation requests if the data set's disposition is uncatalog (DISP=UNCATLG). If the system issues message IEF287I *dsn* NOT UNCATLGD 13, and you need to uncatalog a cataloged data set, update the request and run it to uncatalog the data set.



Update callers of CNZMXURF that do not hold appropriate serialization (Required-IF, as of R10)

Required if you have applications that use the CNZMXURF macro and access the UCME that CNZMXURF returns, and you want to ensure that application errors do not occur because of a failure to hold serialization.

The CNZMXURF macro locates the console control block (UCME) that contains a specific console ID. Applications should hold serialization while using CNZMXURF and accessing the UCME that CNZMXURF returns.

Before z/OS V1R10, when console services shared mode was the only mode available, the UCME contents did not change, so applications that did not hold serialization ran without error.

Beginning with z/OS V1R10, console services distributed mode is added, and it is now more important that applications hold serialization. When the system is in distributed mode (the default is shared mode) and a UCME is accessed without serialization, the contents of UCME storage could become cleared while being examined by the application. In addition, if the application is examining a UCME for an active console (when the system is in distributed mode) and the console becomes inactive, the UCME is cleared. If serialization is not held, the application can fail.

With z/OS V1R10 (and APAR OA19948 on z/OS V1R9 and prior releases), the Tracking Facility was enhanced so that it reports applications that call the CNZMXURF service without holding appropriate serialization. You should run the Tracking Facility to find the applications that need to be updated. In Tracking Facility output, look for violations that start with "UCME-CMS".

Migration action:

1. Obtain the Tracking Facility exclusion list at <http://www.ibm.com/eserver/zseries/zos/downloads/>.
2. Install the PTF for APAR OA19948 on z/OS V1R9 or z/OS V1R8 so that you can use the Tracking Facility to detect inappropriate serialization.
3. Activate the Tracking Facility. Use the exclusion list you obtained in the step above.
4. Periodically issue the DISPLAY OPDATA, TRACKING command to obtain the results of the tracking.
5. Forward the output to consoles@us.ibm.com so that the exclusion list can be updated for future users of the Tracking Facility.
6. Obtain necessary modifications for the affected programs from the program owners.

Modify applications that receive IOS ENF signal 10, 31, 32, or 33 for PPRC devices (Required-IF as of R10)

Required if you receive ENF event code 10, 31, 32, or 33.

As of z/OS V1R10, peer-to-peer remote copy (PPRC) primary and secondary devices can be defined in subchannel set 1. Listeners for IOS ENF signals that include either device numbers or UCB addresses for PPRC primary or secondary devices need to qualify the device numbers for these devices with a subchannel set ID. The ENF event codes may contain these devices in their parameter lists. Any code directly using a device number from these ENF signal parameter lists or using UCBCHAN from the UCB addresses in the parameter lists to identify a device will also need to refer to the subchannel set ID supplied in the parameter list or UCBSSID from the UCB address to correctly identify the device.

Migration action: Look for a parameter list received by an ENF listen exit that is established by an ENFREQ ACTION=LISTEN for ENF event code 10, 31, 32, or 33. Based on the ENF event code, do the following:

- **10** If the UCBCHAN from the UCB addresses in the parameter list is used to identify a device, the UCBSSID is required to be used to correctly identify the device. The UCBSSID can be referenced using the IOSDUPI mapping macro based on the UCBIEXT pointer.
- **31 or 32** If the DCCD parameter list is being processed, and the DCCDEYYP=DCCDDEV entries are used, use the DCCDEYYP=DCCDSDEV as well. Use the DCCDDEVN from the DCCDSDEV entry and DCCDSSID from the DCCDDEV entry in combination to correctly identify the device. The DCCDSDEV entries are the same as the DCCDDEV entries but they have a DCCDEYYP=DCCDDEV entry immediately following them.
- **33** When processing the DACH parameter list, if the DACHQN=DACHPAV entries are being used, the DACHQN=DACHPAV_AS entries should also be used, and if the DACHQN = DACHIORA entries are being used, the DACHQN=DACHIORA_AS entries should also be used. Use the DACH_PAV_DEVN from the DACHPAV_AS entries and DACH_PAV_SSIDBase in combination to correctly identify the device. The DACH_IORA_DEVN from the DACHIORA_AS entries should be used in combination with DACH_IORA_SSID to correctly identify the device.

DACHTYPED	DACHQN if device is in subchannel set 0	DACHQN if device is not in subchannel set 0	Meaning
LPE	DACHIORA	DACHIORA_AS	I/O resource available
PAVS	DACHPAV	DACHPAV_AS	Change in PAV UCB

For any listeners of ENF signal 10, 31, 32, or 33, use the new SPECIAL(YES) keyword and the SCHSET keyword on UCBLLOOK or UCBSKAN invocations.



Reallocate the WLM couple data set **(Required-IF as of R10)**

Required if the current WLM couple data set is not significantly larger than required for the currently used WLM service definition so that the data set can accommodate the increased size of the WLM policy structures.

In z/OS V1R10, you might need a larger workload management (WLM) couple data set because the WLM service class and workload records in the WLM policy are longer, as shown below:

IWMSPOL entry	Pre-z/OS V1R10 entry length	z/OS V1R10 entry length
Policy information	168	360
Service class information	64	144
Workload information	40	112

Migration action: Before you activate a WLM policy on z/OS V1R10, do the following:

1. Allocate a new WLM couple data set using the IXCL1DSU utility, as described in "Allocate a WLM Couple Data Set" in *z/OS MVS Planning: Workload Management*. Use the current NUMBER specification for each section (POLICY, WORKLOAD, SRVCLASS, and so forth).

Note: By using z/OS V1R10 to allocate your WLM couple data set, the space allocated will be sufficient for policy activation on z/OS V1R10. Ensure that the values provided for number of policies, workloads, and service classes are the current values because these values will be used by WLM to calculate the space required. An overspecification to allow for growth is acceptable.
2. Switch to the new WLM couple data set or sets as described in "Make a Newly Formatted Couple Data Set Available to the Sysplex" in *z/OS MVS Planning: Workload Management*.
3. Update the COUPLExx parmlib member to specify the new WLM couple data set or sets.

During policy activation, if WLM determines that the WLM couple data set is too small, the following message is issued: IWM003I VARY WLM FAILED, WLM COUPLE DATA SET TOO SMALL. You would then have to allocate and active a new, larger WLM couple data set.

Accommodate the change of directory information for PDSE program objects with authorization code zero (Required-IF, as of OA12579 on R7/R8/R9)

Required if you use IEHLIST LISTPDS output or examine the directory entries to determine the authorization requirement of PDSE program objects.

If the authorization code for a program object (PDSE) is not specified or is specified as zero during bind, the program management binder builds a directory entry with the bit set indicating that the authorization code is not present. However, this is not consistent with load module (PDS) processing and the IEHLIST LISTPDS command shows inconsistent information for PDS modules and PDSE program objects where the authorization code is either not specified or is specified as zero (using the binder AC option or SETCODE control statement). For PDSEs, the information is blank unless a nonzero authorization code is specified. For PDSs, the Auth Req column contains NO if the authorization code is zero or unspecified. As of z/OS V1R9 and APAR OA21759, the binder always sets a valid authorization code in the directory for a program object (PDSE) with the value specified by the user on the AC option or SETCODE control statement, or the default value if none is specified.


Migration action: Ensure that the change of directory information for the PDSE program object with authorization code zero does not affect your existing processing. For example, if your program reads the Auth Req column listed by the IEHLIST LISTPDS command for the PDSE program object, it needs to be adjusted to handle the new display of NO, which means AC=00.

Notes: 1. To update the directory information and achieve the improved IEHLIST LISTPDS command output (new display of NO), existing program objects in PDSE data sets must be relinked with the program management binder after applying the PTF for APAR OA21759.





2. If there is no APF code stored in the directory entry for a program object, ISPF displays 00 in the APF authorization code column on the member list panel. This functionality was introduced by ISPF APAR OW52063, which was integrated in z/OS V1R5 and does not depend on this enhancement to the program management binder.

z/OS 1.10 Migration Actions

Communications Server Migration Actions for z/OS R10 from z/OS R8




Migration Actions You Can Do NOW:

- 
 - **IP Services: Migrate from Network Database function (Recommended)**
 - R10 is planned to be the last release to support the **NDB** function.
 - Change your processes to use the distributed data facility (**DDF**) provided by **DB2** for z/OS, and the **DB2 Runtime Client**.
- 
 - **IP Services: Migrate from the Dynamic Host Configuration Protocol server (Recommended)**
 - R10 is planned to be the last release to support the **DHCP** server function.
 - Investigate using a **DHCP** server on **Linux** for **System z**.
- 
 - **IP Services: Migrate from the Boot Information Negotiation Layer function (Recommended)**
 - R10 is planned to be the last release to support the **BINL** function.
 - Consider using the product **IBM Tivoli Provisioning Manager for OS Deployment V5**.
- 
 - **IP Services: Migrate from BIND DNS 4.9.3 to 9.2.0 (Recommended)**
 - R10 is planned to be the last release to support **BIND DNS 4.9.3**.
 - Use **BIND DNS 9.2.0** as a replacement (introduced in **z/OS R4**).

16 © 2008 IBM Corporation

z/OS 1.10 Migration Actions

Communications Server Migration Actions for z/OS R10 from z/OS R8

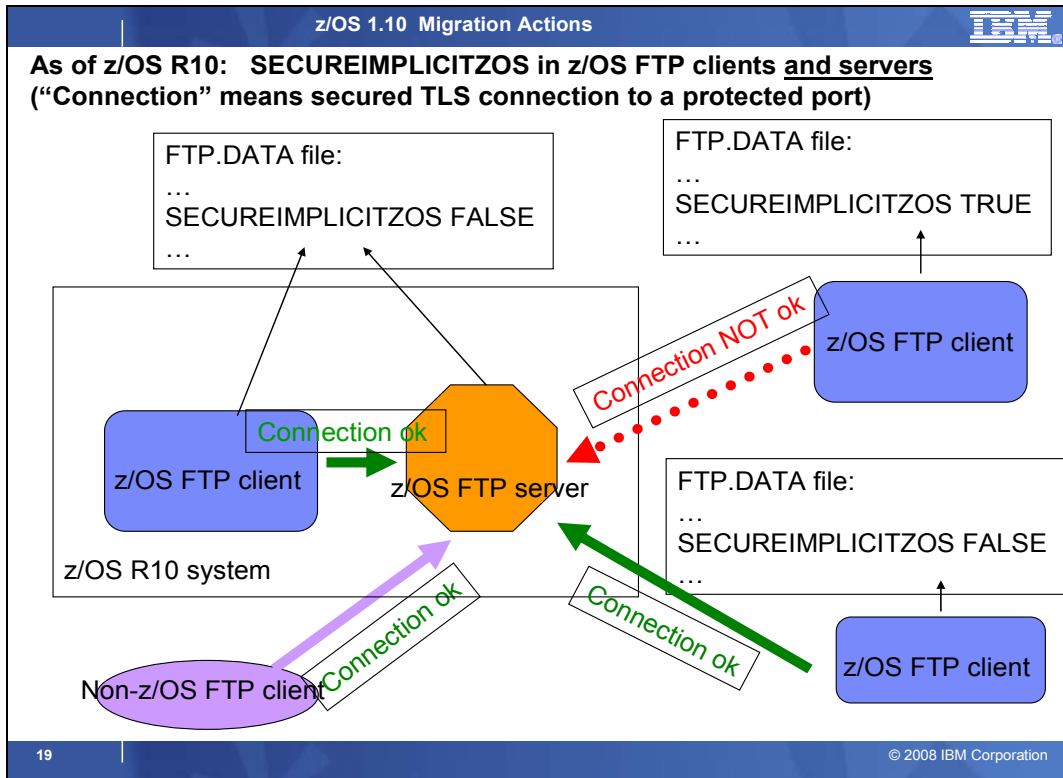
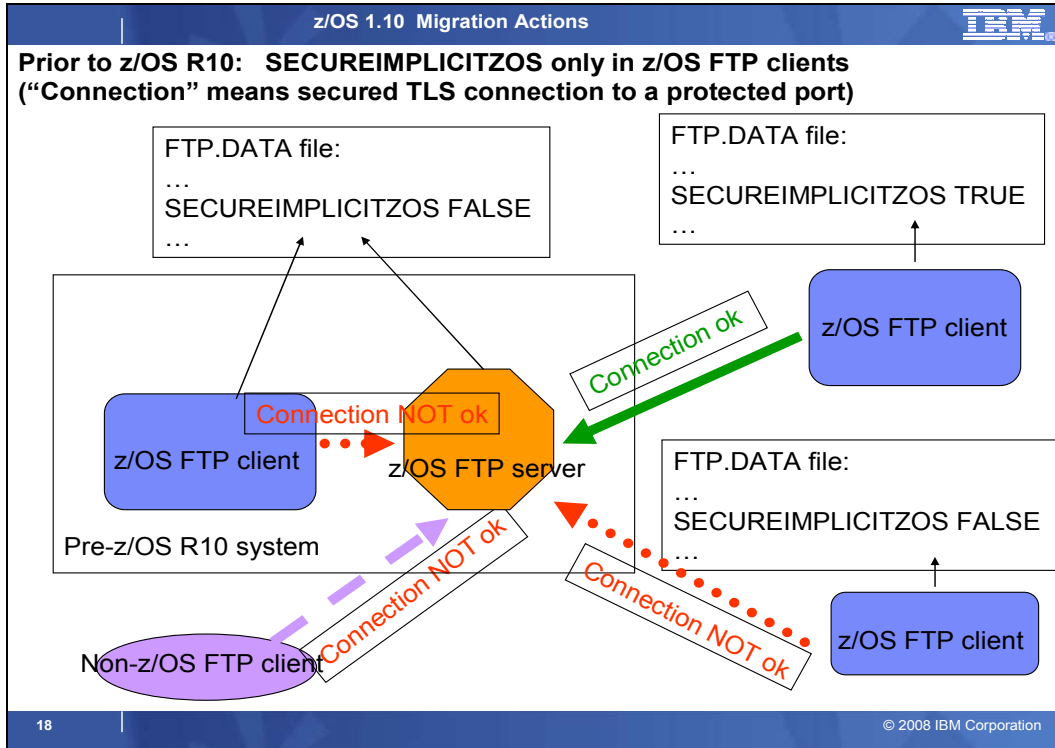


Migration Actions You Can Do NOW:

- **IP Services: Migrate FTP servers sharing FTP.DATA with FTP clients (Required-IF, as of R10)**
 - Before R10, only FTP client supported the **SECUREIMPLICITZOS** statement.
 - This statement specified when the FTP client should negotiate the security handshake with the FTP server for **TLSPORT**, implicit connections. **TRUE** is the default.
 - As of R10, FTP server enhanced to support implicitly secured **TLS** connections from non-z/OS FTP clients. Thus, **SECUREIMPLICITZOS** is now supported in the FTP server.
 - Sharing (or not sharing) the **FTP.DATA** between z/OS FTP client and server may have different behaviors....see *next foils*.
 - **Know new rules for behavior!**
 - Same value for successful connection.
 - Non-z/OS FTP clients need **SECUREIMPLICIT FALSE**.


17 © 2008 IBM Corporation

Migrating to z/OS 1.10 - Part 2





z/OS 1.10 Migration Actions

Communications Server Migration Actions for z/OS R10 from z/OS R8



Migration Actions You Can Do NOW:

- **IP Services: Migrate to the TN3270E Telnet server that runs in its own address space (Required-IF, as of R9)**





- Before R9, could run TN3270E Telnet server as part of the TCP/IP stack.
- As of R9, must run the TN3270E Telnet server in its own separate address space.
 - This function was introduced in z/OS R6, and provides better visibility and control over TN3270E functions.
 - Perform security customization.
 - Move TN3270E server profile statements from TCP/IP profile into a new data set.
 - Modify sample JCL as necessary.
 - Update any automation, as necessary.

20 © 2008 IBM Corporation


z/OS 1.10 Migration Actions

Communications Server Migration Actions for z/OS R10 from z/OS R8



Migration Actions You Can Do NOW:

- **SNA Services: Ensure compatible levels of VTAM for HPR sessions (Required-IF, as of R10)**




- To run R10 as an HPR-capable interchange node in a mixed subarea and APPN network, you must ensure that all HPR-capable VTAMs in your APPN network (and in attached APPN networks) are running R8 or later. Sessions established with or through pre-R8 VTAMs may fail.
 - Why? R8 and later provides additional information on APPN session establishment flows to identify when sessions cross from APPN into subarea (or vice versa) through an interchange node. This additional information is used by R10 to separate interchange node sessions from APPN-only sessions by placing them on different RTP pipes. R10 interchange nodes might incorrectly place interchange node sessions onto the wrong RTP pipe, which could result in session setup failures.
 - May see that interchange nodes create more RTP pipes to adjacent APPNs than before, because separate RTP pipes are now used for interchange node sessions vs. APPN-only sessions.
- **SNA Services: Remove all apps use of APPC Application Suite (Req-IF, in R9)**
 - Note that this is not the z/OS BCP APPC support!

21 © 2008 IBM Corporation

z/OS 1.10 Migration Actions

Communications Server Migration Actions for z/OS R10 from z/OS R8



Migration Actions You Can Do NOW:

- **IP and SNA Services: Increase region size for Telnet (Required-IF, as of R10)**
 - As of R10, ECSA usage for Telnet is reduced. However, Telnet private storage usage increased. The private storage increase depends upon the number of Telnet sessions per LPAR.
 - Determine the number of Telnet sessions that will exist on the LPAR.
 - Increase the REGION size on the Telnet started task by 1 MB for each 25,000 Telnet sessions.
 - Examine the values for your CRPLBUF pool defined in the ATCSTRxx VTAM start list, particularly the base size. If the base size is large because of a large number of Telnet sessions, you can reduce it. Failure to reduce it can minimize the amount of ECSA savings realized.
- **IP Services: Recompile customer-written listeners (Required-IF, as of R9)**
 - IP CICS Socket listener control area (LCA) control block has changed in R9.
 - If you wrote your own listener to use the IP CICS Socket Interface management transaction, EZAO, you must recompile those listeners.

22 © 2008 IBM Corporation

Communications Server Migration Actions Between z/OS V1 R8 and z/OS V1 R10

These migration actions were taken from *z/OS Migration*. Many descriptions and actions have been severely shortened for inclusion in this presentation. For the complete descriptions and actions, refer to *z/OS Migration*.

Communications Server Migration Actions You Can Do Now



IP Services: Migrate from Network Database function (Recommended)

Not required, but recommended because if you use the NDB function because the action is planned to become a requirement in the release following z/OS V1R10.

z/OS V1R10 is planned to be the last release in which z/OS Communications Server supports the Network Database (NDB) function. After z/OS V1R10, the function will be removed from z/OS.

Migration action: Change your processes to use the distributed data facility (DDF) provided by DB2 for z/OS, and the DB2 Runtime Client. DDF allows client applications running in an environment that supports Distributed Relational Database Architecture™ (DRDA®) to access data at DB2 servers.

Tip: Use IBM Health Checker for z/OS to determine if the NDB function is in use. The check is named ZOSMIGV1R10_CS_NDB



IP Services: Migrate from the Dynamic Host Configuration Protocol server (Recommended)

Not required, but recommended because if you use the DHCP server function because the action is planned to become a requirement in the release following z/OS V1R10.

z/OS V1R10 is planned to be the last release in which z/OS Communications Server supports the Dynamic Host Configuration Protocol (DHCP) server function. After z/OS V1R10, the function will be removed from z/OS.

Migration action: You should implement the DHCP server on a system other than your z/OS system. If you want to continue to run the DHCP server on your System z hardware, you can implement a DHCP server inside a Linux for System z image. DHCP servers are widely available on a variety of platforms.

Tip: Use IBM Health Checker for z/OS to determine if the DHCP function is in use. The check is named ZOSMIGV1R10_CS_DHCP.



IP Services: Migrate from the Boot Information Negotiation Layer function (Recommended)

Not required, but recommended because if you use the BINL function because the action is planned to become a requirement in the release following z/OS V1R10.

z/OS V1R10 is planned to be the last release in which z/OS Communications Server supports the Boot Information Negotiation Layer (BINL). After z/OS V1R10, the function will be removed from z/OS.

Migration action: Consider using the product IBM Tivoli® Provisioning Manager for OS Deployment V5 (5724-Q99) for network-based operating system installation services.

Tip: Use IBM Health Checker for z/OS to determine if the BINL function is in use. The check is named ZOSMIGV1R10_CS_BINL.



IP Services: Migrate from BIND DNS 4.9.3 (Recommended)

Recommended if you use BIND DNS 4.9.3 function because it is planned to become a requirement in the release following z/OS V1R10.

z/OS V1R10 is planned to be the last release in which z/OS Communications Server supports BIND DNS 4.9.3. After z/OS V1R10, the function will be removed from z/OS.

Migration action: You should implement BIND DNS 9.2.0 as a replacement. BIND DNS 9.2.0 is included in z/OS beginning with V1R4. If you exploit the Connection Optimization (DNS/WLM) feature of BIND 4.9.3, you should investigate alternative solutions. One alternative would be using the sysplex distributor function, which is implemented using distributed dynamic VIPAs (VIPADISTRIBUTE statement in the TCPIP PROFILE). Another alternative is the z/OS Load Balancing Advisor in combination with the Automated Domain Name Registration application (ADNR), which provides the ability to automatically update DNS with the availability status of sysplex resources.

Tip: Use IBM Health Checker for z/OS to determine if the BIND DNS 4.9.3 function is in use. The check is named ZOSMIGV1R10_CS_BIND4.

IP Services: Update IP filter policy to filter IP fragments correctly (Required-IF, as of R10)

Required if you enable RFC 4301 compliance and your IP filter policy selectively matches routed traffic based on TCP port, UDP port, ICMP type and code, ICMPv6 type and code, or OSPF type.

IP filter policy support for filtering fragments is improved in z/OS V1R10. Before z/OS V1R10, Communications Server filtered all IP fragments using a policy of first possible filter match, and filtered IPv6 fragments as protocol IPv6Frag. Beginning in z/OS V1R10, Communications Server follows rules and restrictions established by RFC 4301 (<http://www.ietf.org/rfc/rfc4301.txt>) to ensure proper classification of fragments. Communications Server does not implement stateful fragment checking; therefore, restrictions are added as required by RFC 4301 to optionally require that IP filter rules applying to routed traffic not apply to specific ports, types, or codes. These restrictions are globally configured as to whether or not they are applied. Additionally, support is added for use of OPAQUE protocol selector values for IPv6 fragments in which the upper layer protocol value is not known.

Migration action: If you configure the RFC4301Compliance parameter on the IpFilterPolicy statement as No, or use the default value of No, no further action is needed at this time. If you configure the RFC4301Compliance parameter on the IpFilterPolicy statement as Yes, perform the steps that are appropriate for you as described in *z/OS Migration*.

IP Services: Migrate FTP servers sharing FTP.DATA with FTP clients (Required-IF, as of R10)

Required if your FTP server and FTP client use a common FTP.DATA file or data set, and you have coded SECUREIMPLICITZOS FALSE in FTP.DATA.

Before z/OS V1R10, only the FTP client supported the SECUREIMPLICITZOS statement. In z/OS V1R10, the FTP server has been enhanced to support implicitly secured TLS connections from non-z/OS FTP clients. Thus, the server will start supporting the SECUREIMPLICITZOS statement. If you want to keep the behavior of past releases, you must take action.

When SECUREIMPLICITZOS FALSE is coded in FTP.DATA, the server will support implicitly secured TLS connections when non-z/OS clients connect to the server's protected port as specified by the TLSPOrt statement in FTP.DATA. The client that shares FTP.DATA with the server will now be able to log into the server's protected port. z/OS FTP clients that do not share FTP.DATA with the server could have configured SECUREIMPLICITZOS TRUE, the default. These clients will no longer be able to log into a z/OS FTP server using the protected port when the server shares FTP.DATA with an FTP client that has specified SECUREIMPLICITZOS FALSE in FTP.DATA.

Migration action: Do one of the following:

- Create a separate FTP.DATA file or data set for the FTP server, and code SECUREIMPLICITZOS TRUE (the default) in the server's FTP.DATA. This preserves the behavior of past releases.

- For clients that share FTP.DATA with the server, change the SECUREIMPLICITZOS setting after starting the client but prior to logging into the server's protected port, as follows:
 1. Start the FTP client with the -n option to suppress logging into the server.
 2. Use the **locsite** subcommand with the **secureimplicitzos** parameter to set SECUREIMPLICITZOS to TRUE.
 3. Use the **open** subcommand to log into the FTP server's protected port. **Tip:** Use the **locstat** subcommand to display the client's current SECUREIMPLICITZOS setting.

Rules:

- To enable a z/OS FTP client to log into a z/OS FTP server's protected port, the client and server must be configured with the same value for SECUREIMPLICITZOS.
- To enable a non-z/OS FTP client to log into a z/OS FTP server's protected port, SECUREIMPLICITZOS FALSE must be coded in the server's FTP.DATA.

IP Services: Update network management applications for SNMP support of RFC versions of networking MIB modules (Required-IF, as of R10)

Required if you have network management applications that use the changed IP-MIB MIB objects, or if you have network management applications that are using the internet draft sample files for the MIB modules.

In z/OS V1R5 and V1R6, in order to support IPv6 network management data, the SNMP TCP/IP subagent was enhanced to support IETF Internet drafts of the standard networking MIB modules, which are: IP-MIB, IP-FORWARD-MIB, TCP-MIB, UDP-MIB, and INET-ADDRESS-MIB.

The Internet drafts provided support for IPv6 network management data in the form of version-neutral tables, that is, tables that supported both IPv4 and IPv6 data.

As of z/OS V1R10, the SNMP TCP/IP subagent has been changed to support the RFC version of these same MIB modules, as follows (RFC text is at the Web addresses in parentheses):

- RFC 4293: IP-MIB (<http://www.ietf.org/rfc/rfc4293.txt>)
- RFC 4292: IP-FORWARD-MIB (<http://www.ietf.org/rfc/rfc4292.txt>)
- RFC 4022: TCP-MIB (<http://www.ietf.org/rfc/rfc4022.txt>)
- RFC 4113: UDP-MIB (<http://www.ietf.org/rfc/rfc4113.txt>)
- RFC 4001: INET-ADDRESS-MIB (<http://www.ietf.org/rfc/rfc4001.txt>)

The RFC version of the IP-MIB renames and renumbers some of the MIB objects in the earlier Internet draft, so if you have network management applications that use the changed IP-MIB MIB objects, you might have to change the applications so they continue to access the same data. Also, the Internet draft sample files for the MIB modules are no longer shipped, so if you copied these samples for use by a network management application, you should replace them with the RFC version of the MIB modules.

IP Services: Specify at least one valid ZIIP subparameter on GLOBALCONFIG ZIIP statements (Required-IF, as of R10)

Required if you are currently specifying a TCP/IP configuration statement of GLOBALCONFIG with a ZIIP parameter and no associated subparameters, and you do not want your ZIIP parameter to be ignored.

Before z/OS V1R10, specification of a GLOBALCONFIG statement with the ZIIP parameter but without a valid ZIIP subparameter would cause the NOIPSECURITY subparameter to be assumed. Beginning with z/OS V1R10, at least one valid ZIIP subparameter is required. Lack of a valid ZIIP subparameter on a GLOBALCONFIG statement causes the ZIIP parameter to be ignored, and informational message EZZ0318I is issued.



IP Services: Migrate from QoS TR policy to IDS TR policy (Required-IF, as of R10)

Required if TR policies are configured as part of the QoS policy type (PolicyScope TR).

z/OS V1R9 was the last release that supported the Traffic Regulation (TR) policy as part of the Quality of Service (QoS) policy type. The TR policy function is still available but only as part of the Intrusion Detection Services (IDS) policy type. Note that this change is only for the TR policy configuration. The TR policy functions themselves remain unaffected and continue to run.

Migration action:

1. Convert QoS policies that specify the PolicyScope TR parameter on the PolicyAction statement to IDS policies. All PolicyRule statements that reference any such PolicyAction statement must be converted to IDSRule statement with the ConditionType TR parameter. The PolicyAction statements must be converted to IDSAction statements with the ActionType TR parameter. The new IDS policies must be configured in an existing or new IDS configuration file.

2. If you were not using IDS policies prior to the conversion, specify the IDSCONFIG statement to point to the new image-specific IDS configuration file. Optionally, configure the IDS policies in both common and image-specific configuration files, and also specify the CommonIDSCONFIG statement.
3. Refresh the policies by issuing the MODIFY proname,UPDATE command, waiting for the refresh interval to expire, or restarting the Policy Agent. If you are using UNIX files and you previously started the Policy Agent using the -i startup option, no action is necessary; the new policies will be refreshed as soon as the configuration files are saved.

IP Services: Delete duplicate named inline statements from Policy Agent config files (Required-IF, as of R9)

Required if you have duplicate named inline statements within the same statement and you export only the last duplicate name to be included.

Starting in z/OS V1R9, a nonpersistent system name is generated for each named inline statement where the name is optional, using the named portion of the statement name with a unique identifier. This is to avoid the named inline statement from being reused as a reference name. Therefore, for Policy Agent IDS, AT-TLS, and IPSec flat-file configuration files, you must delete any duplicate inline statement names for any statements where the name is optional.

Migration action:

1. In Policy Agent configuration files, delete all but the last entry of any duplicate named inline statement.
2. Refresh Policy Agent to pick up the updated configuration files.



IP Services: Migrate from LDAP protocol version 2 to LDAP protocol version 3 for QoS and IDS policies in Policy Agent (Required IF, as of R9)

Required if you use LDAP protocol version 2 for QoS or IDS policies.

z/OS V1R8 was the last release that supported the LDAP protocol version 2 for QoS and IDS policies in Policy Agent. Starting with z/OS V1R9, you have to use the LDAP protocol version 3 for QoS and IDS policies in Policy Agent. The LDAP protocol version 3 has improvements in internationalization, authentication, referral, and deployment. Failure to move to LDAP protocol version 3 will result in Policy Agent failing on the ReadFromDirectory statement with the following message: EZZ8439I PAGENT READFROMDIRECTORY STATEMENT CONTAINS ERRORS.

Migration action:

1. Install the LDAP protocol version 3 schema definition files on the LDAP server instead of the LDAP protocol version 2 schema definition files.
2. Change the protocol version (LDAP_ProtocolVersion parameter) on the Policy Agent ReadFromDirectory configuration statement to use protocol version 3.



IP Services: Migrate to the TN3270E Telnet server that runs in its own address space (Required-IF, as of R9)

Required if the TN3270E Telnet server is running as part of the TCP/IP stack.

Before z/OS V1R9, the TN3270E Telnet server could run as part of the TCP/IP stack or in its own separate address space. Starting in z/OS V1R9, the TN3270E Telnet server is no longer available to run as part of the TCP/IP stack. You must run the TN3270E Telnet server in its own separate address space.

The TN3270E Telnet server has been able to run in its own address space since z/OS V1R6. This enhancement provides visibility and control over the TN3270E function separate from the TCP/IP stack. For example, users can run the TN3270E Telnet server at a different priority than the TCP/IP stack. This enhancement also provides the ability to stop and restart the TN3270E Telnet server without stopping the TCP/IP stack. This makes it easier to reset the server or install maintenance. Overall, problem diagnosis is easier and better when the TN3270E Telnet server and the TCP/IP stack are separate.

Migration action:

1. In z/OS V1R6, the default program properties table (PPT) was updated for EZBTNINI, the TN3270E Telnet server program. Ensure that your SCHEDxx parmlib member does not specify EZBTNINI or modify the IBM defaults for EZBTNINI (NOCANCEL, KEY(6), PRIV, NOSWAP, and SYST).
2. If you want to change the priority of Telnet, do so by assigning Telnet to a service class other than the default SYSSTC class in the STC subsystem. EZBTNINI would be put in the SYSSTC service class if it were not explicitly classified in the workload manager (WLM) classification rules.
3. Define security for the procedure name and the associated user ID.

4. Move all TN3270E Telnet server profile statements from the TCP/IP profile into a new data set to be accessed by Telnet. If the Telnet server port is reserved using INTCLIEN, change the job name from INTCLIEN or the TCPIP stack name to the job name Telnet is now running under.
5. Modify the sample JCL found in SEZAINST(EZBTNPRC) as necessary for your installation. Specify the new profile data set on the PROFILE DD entry of the JCL.
6. Update any automated operator scripts to specify the name of the TN3270E Telnet server address space as the PROCNAME field in any DISPLAY or VARY commands for TELNET options.

Note: In z/OS V1R9 and V1R8 without the PTF for TCP/IP APAR PK64947 installed, if you run the TN3270E Telnet server without a UID value of 0, the server is not able to increase the MAXFILEPROC value. This could result in the TN3270 Telnet server refusing connections. In z/OS V1R10 (and also in V1R9 and V1R8 with the PTF for APAR PK64947 installed), if you run the TN3270E Telnet server without a UID value of 0, the server attempts to set the effective UID to zero. If the request fails, the user ID probably was not permitted to the BPX.SUPERUSER FACILITY class. This could result in the server refusing connections.



IP Services: Remove customization of SNMP sysObjectID MIB object in OSNMPD.DATA File

(Recommended)

Recommended because the ability to customize the sysObjectID value will not be supported in a future release.

The SNMP agent allows you to provide some initial settings for a small set of MIB objects by using the OSNMPD.DATA file. One of the objects for which an initial value can be provided is sysObjectID.0. The sysObjectID.0 object is the vendor's authoritative identification of the network management subsystem contained in the entity. That is, it is intended to uniquely identify the SNMP agent. Changing this value is not recommended and the ability to change it will be disabled in a future release. As of z/OS V1R4, warning message EZZ6317I is written to the syslog daemon if the object is set by using the OSNMPD.DATA file. As of z/OS V1R8, message EZZ6317I is also written to the console.

Migration action: Review the statements in your OSNMPD.DATA configuration file. If this file contains a statement for the sysObjectID object, remove the statement from the file.

IP Services: Use the new OSAENTA trace default (Recommended, as of R8 APAR PK36947)

Recommended if you are currently using the OSA-Express Network Traffic Analyzer and do not have the PTF for APAR PK36947 installed, on z10, z9 EC/BC and OSA-Express2 feature in QDIO mode, with OSA-Express Network Traffic Analyzer support.

In z/OS V1R8, serviceability for the OSA-Express2 feature was enhanced by the introduction of an OSA-Express network traffic analyzer (OSAENTA) trace. This function allows Communications Server to control and format the tracing of frames collected in the OSA-Express2 feature at the network port.

Before APAR PK36947 on z/OS V1R8, the default behavior of the OSAENTA trace when it was enabled was to collect all frames collected at the OSA-Express network port. APAR PK36947 introduced two new parameters, NOFILTER and DISCARD, which default to NOFILTER=NONE and DISCARD=EXCEPTION. These two parameters change the default behavior to only collect frames discarded by the OSA-Express for unusual conditions when the trace is enabled.

Migration action:

To obtain the benefit of improved serviceability, do not adjust the default behavior of the OSAENTA trace introduced in z/OS V1R8 by APAR PK36947. This default behavior is to collect only frames discarded by the OSA-Express for unusual conditions when the trace is enabled. If you want the prior behavior, specify NOFILTER=ALL and DISCARD=NONE on the OSAENTA statement or command that enables the OSAENTA trace. This overrides the default behavior when the trace is enabled to collecting all frames received by the OSA and not collecting discarded packets, as was done before APAR PK36947.



SNA Services: Ensure compatible levels of VTAM for HPR sessions (Required-IF, as of R10)

Required if you plan to enable APPN and HPR support in VTAM using the NODETYPE and HPR start options.

In order to run z/OS V1R10 Communications Server as an HPR-capable interchange node in a mixed subarea and APPN network, you must ensure that all HPR-capable VTAMs in your APPN network (and in attached APPN networks) are running z/OS V1R8 or later Communications Server, or sessions established with or through these earlier VTAMs might fail. z/OS V1R8 and later Communications Server provides additional information on APPN session establishment flows to identify when sessions cross from APPN into subarea (or vice versa) through an interchange node. This additional information is used by z/OS V1R10 Communications Server to separate interchange node sessions from APPN-only sessions by placing them on different RTP pipes. If any of the VTAMs in your network (or in attached APPN networks) are not running z/OS V1R8 or later, then z/OS V1R10 interchange nodes might incorrectly place interchange node sessions onto the wrong RTP pipe, which could result in session

setup failures. (If any of the HPR-capable VTAMs in your APPN network or in attached APPN networks are running an earlier release of z/OS Communications Server, contact your IBM representative to find out what alternatives are available.)

You should also be aware that this new function might result in interchange nodes creating more RTP pipes to adjacent APPN nodes than prior releases because separate RTP pipes are now used for interchange node sessions versus APPN-only sessions.

As a result of separating interchange node sessions from APPN-only sessions by placing them onto separate RTP pipes, there may be an increase in the number of RTP pipes created by interchange nodes to adjacent APPN nodes and a corresponding increase in the amount of storage used to represent these RTP pipes.

Migration action: Ensure that all HPR-capable VTAMs in your network (and any APPN-attached networks) are running z/OS V1R8 or later Communications Server. All HPR-capable VTAMs must be upgraded to this level before IPLing the first z/OS V1R10 Communications Server interchange node.

SNA Services: Update applications and user exits that use the VTAM version and release level in algebraic expressions (Required-IF, as of R10)

Required if any user-written applications or exits use the VTAM release level in algebraic comparisons to determine release support.

The VTAM version and release level is available to applications and user exits to allow the applications and exits to react based on the level of VTAM that is active. Before z/OS V1R10, applications and exits could use the version and release fields algebraically to determine the level of support. Beginning with z/OS V1R10, the fields cannot be used algebraically.

For example, after a VTAM application program issues an OPEN macro instruction with an access method control block (ACB) to indicate it desires VTAM services, the VTAM version and release level is available to it in the access-method-support vector list. The application cannot use the two bytes containing the version and release in algebraic expressions, such as checking whether the two bytes are greater than the characters 1 and 9. The reason is that, with z/OS V1R10, the version and release characters are 1 and A, respectively, and the character A cannot be used reliably in algebraic comparisons.

SNA Services: Remove all applications that use Communications Server APPC Application Suite (Required-IF, as of R9)

Required if you have applications that use the APPC Application Suite APIs.

The APPC Application Suite is a set of common applications originally designed to enhance the value of SNA networks for end users. Because more full-featured alternative applications exist in modern integrated SNA/IP networks, z/OS V1R8 is the last release that includes the APPC Application Suite. As of z/OS V1R9, the APPC Application Suite is no longer shipped with the product and is not supported. But note that APPC itself remains an integral part of z/OS Communications Server's SNA functions, and there are no plans to remove APPC from z/OS.

Migration action: Remove the following:

1. Any applications that use APPC Application Suite AFTP API macros.
2. Any applications that use APPC Application Suite ANAME API macros.
3. The AFTP client REXX exec.
4. The ACOPY client REXX exec.
5. The AFTP server REXX exec.
6. The transaction program (TP) profile of the AFTP server (AFTPD). This is the profile that defines AFTPD to MVS/APPC.
7. The APING client REXX exec.
8. The transaction program (TP) profile of the APING server (APINGD). This is the profile that defines APINGD to MVS/APPC.
9. The ANAME client REXX exec.
10. The transaction program (TP) profile of the ANAME server (ANAMED). This is the profile that defines ANAMED to MVS/APPC.
11. The A3270 application major node definition from VTAMLST.
12. The initialization files of AFTP programs (AFTP, AFTPD, and ACOPY), the ANAME server, and the A3270 server.
13. The VSAM data set with the APPC Application Suite messages. This data set is named VSAM.ASUITE.MMSENU.

Consider the following alternatives to the APPC Application Suite:

- For A3270, consider migrating to TN3270. TN3270 provides a much richer capability assuming IP connectivity exists between the client and server.

Migrating to z/OS 1.10 - Part 2

- For APING, use the DISPLAY APING command that has been provided as a native VTAM command for many years.
- A number of other IBM and vendor products provide SNA file transfer capability (such as NetView FTP) and can be used to replace AFTP. TCP/IP's FTP capability is also a good alternative if an IP infrastructure is in place between the client and server.

Communications Server Migration Actions Pre-First IPL

IP Services: Update network management applications for SNMP interface data changes (Req-IF, as of R10)

Required if the changes affect network management applications that retrieve the affected SNMP data..

Starting in z/OS V1R10, the SNMP TCP/IP subagent interface data support is changed as follows:

- Changes to the standard interface tables, ifTable and ifXTable, which are defined in the IF-MIB from RFC 2233 (<http://www.ietf.org/rfc/rfc2233.txt>), are as follows:
 - The tables will support a dynamically-generated OSA-Express QDIO port entry for IPAQENET and IPAQENET6 interfaces defined only by the INTERFACE profile statement. The interface counters for the OSA-Express QDIO port entry will reflect the sum of the counters for all the IPAQENET and IPAQENET6 interfaces defined for the same port.
 - When defining interfaces with DEVICE and LINK profile statements, a device layer entry is created in the tables, with the link entry as a lower interface layer under it. When defining interfaces with INTERFACE profile statements (other than IPAQENET or IPAQENET6), only an interface entry is created in the interface tables. There is no higher-layer device entry associated with the interface.
- The standard linkUp and linkDown notifications from the IF-MIB will no longer be created for VIPA interfaces.
- The proprietary ibmTcpiMvsDeviceTable (defined in the IBM TCP/IP MVS Enterprise-specific MIB module) will only support entries for IPv4 interfaces either defined with a DEVICE profile statement or dynamically defined due to dynamic XCF or dynamic VIPA configuration.
- The indexes of the proprietary osaexpEthPortTable and osaexpEthSnaTable (defined in the IBM TCP/IP MVS Enterprise-specific MIB module) are changed from the interface index of an interface defined with the LINK profile statement to the interface index of a device or OSA-Express QDIO port.
- The index of the standard dot3StatsTable, which is defined in the EtherLike-MIB from RFC 2665 (<http://www.ietf.org/rfc/rfc2665.txt>), is changed from the interface index of an interface defined with the LINK profile statement to the interface index of a device or OSA-Express QDIO port.

These changes might affect network management applications that retrieve the above SNMP data.

IP Services: Adjust for a new client error code for the FTP client (Required-IF, as of R10)

Required if your FTP client application is expecting a specific client error code when it attempts a client subcommand that needs a connection to the server while the client has no connection to the server..

Starting in z/OS V1R10, the FTP client returns a new client error code, FTP_NEEDS_CONNECTION, when an interactive user or FTP client API program attempts a subcommand that needs a connection to the server. This client error code replaces more general client error codes returned by earlier releases.

IP Services: Update automation that handles FTP job informational messages for MVS data set transfers (Required-IF, as of R10)

Required if your installation uses automation or software to capture FTP client messages or FTP server replies, and you use FTP to transfer MVS data sets.

Before z/OS V1R10, when a file transfer failed because the MVS data set was in use, the FTP server reply and the FTP client message indicated that the data set was in use by another process but did not identify the job or jobs holding the MVS data set. The FTP server would send a 450 reply and the FTP client would send messages EZA2562W and EZA2799W. Beginning with z/OS V1R10, in the above situation:

- The following additional FTP client messages are issued to identify the job or jobs holding the MVS data set when the file transfer fails:
 - A sequence of EZZ9819I messages identifying the jobs that are holding the MVS data set
 - Message EZZ9820I to end the sequence of messages.
- The following additional FTP server replies are issued:
 - A sequence of 125- replies identifying the jobs that are holding the MVS data set
 - A single 125- reply to end the sequence of replies for this sequence

IP Services: Update automation for changed OMPROUTE messages (Required-IF, as of R9)

Required if you have automation that scans for the name "OMPROUTE" in messages.

Migrating to z/OS 1.10 - Part 2

Starting in z/OS V1R9, several of the OMPROUTE messages are changed so that the constant string "OMPROUTE" is replaced with the OMPROUTE job name. Automation might require updates if the message text is examined by the automation software.

Migration action: Do one of the following:

- Change automation that refers to the constant string "OMPROUTE" in OMPROUTE messages (messages in the range EZZ7800 - EZZ8200) to refer to the OMPROUTE job name instead.
- Automate on message numbers instead of constant text in the message.

Non-OMPROUTE messages that contain the constant string "OMPROUTE", such as EZZ9672E, are not affected by this change.

IP Services: Disable dropping of idle connections associated with noncurrent profiles (Recommended at R9)

Recommended for installations that have a need for Telnet clients to remain connected to the server for extended periods of time while not actively in a SNA session and associated with a noncurrent profile.

Starting in z/OS V1R9, the new PROFILEINACTIVE statement in the TELNETGLOBALS, TELNETPARMS, or PARMSGROUP block can be set to cause the cleanup of idle connections that do not have active SNA sessions and are associated with noncurrent profiles. A connection is considered idle when it has a USSMSG or Solicitor panel displayed. When no more connections are associated with a noncurrent profile, all storage used for that profile is returned to the system. The default value for PROFILEINACTIVE is 1800 seconds (30 minutes). The message that reports the cleanup of the idle connections is EZZ6034I TELNET CONN 00000055 LU luname CONN DROP INACT-PF IP..PORT: nn.nn.nn.nn..nnnn.

If you want the pre-z/OS V1R9 behavior, which is for Telnet clients to remain connected to the server for extended periods of time while not actively in a SNA session and associated with a noncurrent profile, you must specify PROFILEINACTIVE 0 in the TELNETGLOBALS, TELNETPARMS, or PARMSGROUP block.

Migration action: To obtain the pre-z/OS V1R9 behavior, which is to disable the dropping of idle connections, specify PROFILEINACTIVE 0 in the TELNETGLOBALS, TELNETPARMS, or PARMSGROUP block.

But if you want to obtain the new behavior, specify PROFILEINACTIVE *sec* in the TELNETGLOBALS, TELNETPARMS, or PARMSGROUP block, where *sec* is the number of seconds you want the function to wait before starting the cleanup of idle sessions.

IP Services: Update applications to handle scope information on Getnameinfo calls (Required-IF, as of R9)

Required if you use the host name value returned by Getnameinfo for displays or as input to other socket API invocations.

Before z/OS V1R9, resolver Getnameinfo processing did not use the zone index value in the input sockaddr_in6 structure. Starting in z/OS V1R9, Getnameinfo processing is enhanced so that scope information is appended to output host names when a nonzero zone index value is present in the input sockaddr_in6 structure and the input IPv6 address being resolved is a link-local address. The scope information is presented to the application in the form *hostname%scope information*. The scope information returned, by default, is the interface name associated with the zone index value. You should verify that your application can handle the scope information properly.

Migration action:

1. Be aware of the potential that Getnameinfo calls may result in host name output of the form *hostname%scope information*.
2. If the host name output is used for calls to socket API calls other than Getaddrinfo, remove the *%scope information* prior to passing the host name on the socket API call.
3. If the host name output is used for display purposes, verify that the displays are not affected by the presence of scope information. If the displays are affected, remove the *%scope information* prior to using the host name in the displays.
4. If the host name output is used for diagnostic purposes, verify that the diagnostics are not affected by the presence of scope information. If the diagnostics are affected, remove the *%scope information* prior to using the host name in the diagnostics.

To avoid steps 2-4 and to prevent the resolver from ever returning scope information as part of the host name output, set the zone index value in the sockaddr_in6 structure to zero prior to calling Getnameinfo. Upon return from Getnameinfo processing, restore the zone index value to its previous setting.

IP Services: Adjust to a change in how AT-TLS policies are deleted from the stack (Required-IF, as of R9)

Required if you use AT-TLS policies.

Before z/OS V1R9, the Policy Agent did not delete AT-TLS policies from the stack if the TTLSSConfig statement was not configured, FLUSH was configured on the Tcplmage statement, and the Policy Agent was started or a MODIFY REFRESH command was entered. If AT-TLS policies were previously configured to the Policy Agent, and the

NOPURGE option was configured on the TTLSSConfig or Tcplmage statement, then the AT-TLS policies were not deleted from the stack when the Policy Agent was shut down. If the TTLSSConfig statement was then removed, the AT-TLS policies would not be deleted from the stack when the Policy Agent was restarted or a MODIFY REFRESH command was entered.

Now, in the previous scenario, AT-TLS policies are deleted from the stack if the Policy Agent is started or a MODIFY REFRESH command is entered without the TTLSSConfig statement configured.

Depending on how the changed behavior affects your system, you might need to modify your Tcplmage statement.

Migration action: If you do not want AT-TLS policies to be deleted from the stack when you delete the TTLSSConfig statement, specify NOFLUSH and NOPURGE on the Tcplmage statement. Be aware that using NOFLUSH also affects other policy types.

If you want AT-TLS policies to be deleted from the stack when you delete the TTLSSConfig statement, specify FLUSH on the Tcplmage statement.

IP Services: Make Configure GLOBALCONFIG SEGMENTATIONOFFLOAD if TCP segmentation offload support is desired (Required-IF, as of R9 with PK47376)

Required if TCP segmentation offload is desired.

Before APAR PK47376 on z/OS V1R9 and APAR PK46334 on z/OS V1R8, V1R7, and V1R6, TCP segmentation was offloaded to the OSA-Express2 feature by default. Beginning with APAR PK47376 on z/OS V1R9 and APAR PK46334 on z/OS V1R8, V1R7, and V1R6, the default behavior is to not offload TCP segmentation to the OSA-Express2 feature. If you want to continue to offload TCP segmentation to the OSA-Express2 feature, you must configure SEGMENTATIONOFFLOAD on the GLOBALCONFIG statement.

Migration action: To obtain the prior behavior, configure SEGMENTATIONOFFLOAD on the GLOBALCONFIG statement in the TCP/IP profile.

IP Services: Make changes for Netstat enhancements (Required-IF)

Required if the changed or removed settings affect automation running off Netstat or front-end programs to Netstat. The Netstat command displays the status of a local host. Each release, the Netstat reports are changed in ways that can affect automation or front-end programs.

Migration action: Accommodate Netstat changes in your automation and front-end programs. You can begin planning your changes by reviewing the ways in which the displays are updated each release. However, you will have to execute the commands to know with certainty what changes to make.

For details about Netstat report changes, see *z/OS Summary of Message and Interface Changes*



IP Services: Update /etc configuration files (Required-IF)

Required if you have customized a configuration file that IBM has changed.

Some utilities provided by Communications Server require the use of certain configuration files. You are responsible for providing these files if you expect to use the utilities. IBM provides default configuration files as samples in the /usr/lpp/tcpip/samples directory. Before the first use of any of these utilities, you should copy these IBM-provided samples to the /etc directory (in most cases). You can further customize these files to include installation-dependent information. An example is setting up the /etc/osnmpd.data file by copying the sample file from /usr/lpp/tcpip/samples/osnmpd.data to /etc/osnmpd.data and then customizing it for the installation.

If you customized any of the configuration files that have changed, then you must incorporate the customization into the new versions of the configuration files.

Migration action: If you added installation-dependent customization to any of the IBM-provided configuration files listed below, make the same changes in the new versions of the files by copying the IBM-provided samples to the files shown in the table and then customizing the files.

Utility	IBM-provided sample file	Target location	What changed and when
Network File System (NFS)	/usr/lpp/tcpip/samples/rpc	/etc/rpc	In z/OS V1R9, the rpc sample was updated to include RPC information for NFS services.
Network File System (NFS)	/usr/lpp/tcpip/samples/services	/etc/services	In z/OS V1R9, the service names and associated port number entries for NFS have been added to the services sample.

Migrating to z/OS 1.10 - Part 2

Utility	IBM-provided sample file	Target location	What changed and when
Network security services (NSS)	/usr/lpp/tcpip/samples/nssd.conf	Update as needed by your installation and put in user directory.	In z/OS V1R10, the NSS server introduced support for discipline-specific NSS clients. The two disciplines supported in V1R10 are IPSec and XMLAppliance. To account for the configurable support of each discipline, the NSS server's sample configuration file was updated to indicate that support for both disciplines is enabled by default. A new syslog level for the SAF access service within the XMLAppliance discipline was also introduced.
Remote Procedure Call (RPC)	/usr/lpp/tcpip/samples/rpc	/etc/rpc	In z/OS V1R9, a sample is provided that includes RPC program names, numbers, and alias names.
SNMP agent	/usr/lpp/tcpip/samples/osnmpd.data	/etc/osnmpd.data	Every release, the value of the sysName MIB object is updated to the current release. In z/OS V1R10, the SNMP agent was changed to create its default z/OS UNIX connection file in the /var file system directory instead of the /tmp directory. So the value of the dpiPathNameForUnixStream MIB object in this sample was changed from /tmp/dpi_socket to /var/dpi_socket. If you use the updated sample, the z/OS UNIX file will be created in the /var file system directory instead of in the /tmp directory. Because the Agent will no longer use the /tmp/dpi_socket file, this file can be removed.

IP Services: Ensure that IP security messages are logged by syslogd as needed (Required-IF, as of R9)

Required if you wish to continue logging TRMD IP security messages EZD0827I and EZD0811I (reason 8 or 9). Before z/OS V1R9, Traffic Regulation Management Daemon (TRMD) IP security messages were always logged to syslogd with a priority of INFO. In some cases, messages EZD0827I (Remote port translated) and EZD0811I (Decapsulation failed, reason 8 or 9) could flood the syslog. These messages could not be excluded from syslog without excluding all TRMD IP Security messages.

Starting in z/OS V1R9, TRMD logging is enhanced to log EZD0827I and EZD0811I (reason 8 or 9) with a priority of DEBUG. This allows you to exclude these messages from your IP security log file. If you wish to continue logging these

messages, you must ensure that your syslogd configuration file does not exclude TRMD IP security messages with a priority of DEBUG.

Migration action:

- Determine whether your syslogd configuration file excludes TRMD IP security messages with a priority of DEBUG.
- Update your syslogd configuration file to include TRMD IP security messages with a priority of DEBUG. For example, if your configuration file contains the following line:

```
*.TRMD*.local4.info /tmp/trmd.log
```

Change it to the following if you wish to continue logging messages EZD0827I and EZD0811I (reason 8 or

9):

```
*.TRMD*.local4.debug /tmp/trmd.log
```

IP Services: Update automation that handles message EZZ4201I (Required-IF, as PK46221 on R9/R8/R7)

Required if you depend on the message EZZ4201I issued in response to the TCP/IP STOP command..

When TCP/IP is started, the console ID used on the START command is saved. This console ID is used when TCP/IP sends messages to the console. When TCP/IP is stopped, message EZZ4201I TCP/IP TERMINATION COMPLETE FOR jobname uses the console ID from the START command rather than the STOP command, causing EZZ4201I to not be trapped. As of APAR PK46221 on z/OS V1R9, z/OS V1R8, and z/OS V1R7, STOP command processing has been changed to save the WTO CART (command and response token) and console identifier value. As a result, message EZZ4201I issued in response to the STOP command is sent to the correct console, that is, the one that issued the STOP command.

IP and SNA Services: Increase region size for TELNET (Required-IF, as of R10)

Required if you use the TN3270E Telnet server.

Starting in z/OS V1R10, extended common system area (ECSA) usage for Telnet sessions is reduced. This change requires growth in Telnet private storage usage. The amount of Telnet private storage increase depends upon the number of Telnet sessions on the LPAR. If Telnet is using most of its allotted region size, you must increase it.

Migration action:

1. Determine the number of Telnet sessions that will exist on the LPAR.
2. Increase the size specified on the REGION parameter of the Telnet started task by 1 MB for each 25 000 Telnet sessions.
3. Examine the values for your CRPLBUF pool defined in the ATCSTRxx VTAM start list, particularly the base size. If the base size is large because of a large number of Telnet sessions, you can reduce it; failure to reduce it can minimize the amount of ECSA savings realized.

IP and SNA Services: Disable path MTU discovery for IPv4 and IPv6 Enterprise Extender connections (Required-IF, as of R10)

Required if you do not want path MTU discovery enable for IPv4 and IPv6 Enterprise Extender connections.

Starting in z/OS V1R10, path MTU discovery is enabled for Enterprise Extender (EE) connections. It is automatically enabled for IPv6 Enterprise Extender connections, and is enabled for IPv4 when the PATHMTUDISCOVERY keyword is specified on the IPCONFIG profile statement for the EE associated TCP/IP stack.

Path MTU discovery uses ICMP fragmentation-needed errors to detect the path MTU for a path. For path MTU discovery to work, you must permit ICMP errors to flow at all hosts along the path of a connection. Path MTU discovery does not function if a firewall blocks ICMP errors.

Migration action: To disable path MTU discovery for IPv4 and IPv6 Enterprise Extender connections, perform one of the following actions:

- Specify PMTUD=NO in the appropriate ATCSTRxx VTAM start list or on the VTAM START command.
- When VTAM is active, issue the MODIFY *procname*,VTAMOPTS, PMTUD=NO command.

SNA Services: Define generic resource resolution preferences using a generic resource preference table (Required-IF, as of R9)

Required if you have modified the default generic resource exit.

Before z/OS V1R9, global generic resource (GR) resolution preferences could be set in the generic resource exit ISTEEXGR. Beginning in V1R9, this is not allowed. Instead, VTAM is initiated with default GR preferences, and the only way to override these default GR preferences is by defining a generic resource preference table.

Action is necessary if you have modified the default generic resource exit. The generic resource exit parameter list flags in GRRFLAG1 are no longer interrogated by VTAM. These flags affected generic resource resolution.

Specifically, they

controlled whether the generic resource exit was called, whether the workload manager was called, and whether VTAM would prefer generic resource instances at the same host as the application or local LU initiating the session. You must define a generic resource preference table with a default entry with the desired generic resource preferences. A default entry in the generic resource preference table is created by adding a nameless GRPREF entry. The generic resource exit will not be called unless the generic resource preference table entry associated with the generic resource indicates to call the generic resource exit.

If you have modified the exit to potentially change the generic resource resolution flags across every call to the GR exit, you no longer have that capability using the GR exit flags. Instead, you can modify the GR exit to logically perform the same function described by the flags, or you can manually (or through automation) modify and activate the generic resource preference table to use new generic resource preferences.

If you do not currently modify the default values, then there is no migration action required because the VTAM default values in z/OS V1R9 are the same as the default values before z/OS V1R9.

Migration action: If you currently modify the default generic resource preferences in the generic resource exit parameter list GRRFLAG1:

1. Define a generic resource preference table with a nameless GRPREF entry and define the GRPREF keywords to correspond to the settings in your existing generic resource exit.
 - a. If the exit sets the GRRFUVX value to 1, set the GRPREF option GREXIT = ON.
 - b. If the exit sets the GRRFWLMX value to 0, set the GRPREF option WLM = OFF.
 - c. If the exit sets the GRRFNPLA value to 1, set the GRPREF option LOCAPPL = OFF.
 - d. If the exit sets the GRRFNPLL value to 1, set the GRPREF option LOCLU = OFF.
2. Modify your generic resource exit to remove the setting of the generic resource preferences in flag GRRFLAG1. Note that if a generic resource exit continues to set the flags in the exit, they will be ignored by VTAM.

SNA Services: Override MPC group activation suspension if manual reactivation of MPC groups is desired (Required-IF, as of R9)

Required if you wish to continue reactivating manually.

Before z/OS V1R9, activation of a multipath channel (MPC) group of CTC connections that lacked a sufficient number of online devices (such as one read device and one write device) failed and needed to be reactivated manually after the required minimum number came online. Starting in z/OS V1R9, the behavior of such activations is enhanced to automatically suspend until the required minimum number has come online, then resume as normal. This is accomplished with a new VTAM start option, MPCACT, which defaults to a value of WAIT, indicating that VTAM is to suspend activations of MPC subchannel groups if the minimum number of read and write subchannels is not available.

If you wish to continue reactivating manually, you must take action to override the new, default behavior.

In addition, in z/OS V1R9, a display of an MPC subarea line while it is suspended contains a new message indicating the wait condition. The new message is IST2219I *resource* ACTIVATION WAITING FOR MINIMUM NUMBER OF DEVICES. During the entire suspension, the MPC subarea line state displays as PALNK.

Migration action: To obtain the prior behavior, which requires manual reactivation, code MPCACT=NOWAIT as a VTAM start option. This specifies that VTAM is to fail activations of MPC subchannel groups if the minimum number of read and write subchannels is not available.

But if you want to obtain the new behavior, which is automatic reactivation, code MPCACT=WAIT (or allow it to default).

SNA Services: Update automation that handles message IST2139I (Required-IF, as of R9)

Required if you have automation routines that examine the message text.

Before z/OS V1R9, the text for message IST2139I was CONNECTIVITY TEST RESULTS DISPLAYED FOR *count* INTERFACES. Starting with z/OS V1R9, the text is CONNECTIVITY TEST RESULTS DISPLAYED FOR *tested_routes* OF *total_routes* ROUTES. The specific changes are:

- In the message text, INTERFACES has been changed to ROUTES.
- Because of the new MAXROUTE parameter on the DISPLAY EEDIAG command, the message text is modified to provide the number of routes tested out of the total number of routes found. The number of routes tested is determined by the MAXROUTE value that is either specified or defaulted to on the DISPLAY EEDIAG command. The default value of the MAXROUTE parameter is 16.

If you are automating on the message text (not the message number), your automation might be affected.

Migration action: Change your automation to look for ROUTES instead of INTERFACES. Be aware that the maximum number of routes out of the number of tested routes is also displayed in the message text of IST2139I.

Communications Server Migration Actions Post-First IPL

IP Services: Use ETHERNET instead of TOKENRING with IPCS CTRACE (Recommended, as of R10)

Not required, but recommended if you use the TOKENRING keyword of the IPCS CTRACE subcommand because someone could be confused by ETHERNET taking effect when they see that TOKENRING was specified.

Before z/OS V1R10, the TOKENRING keyword of the SNIFFER option of the IPCS CTRACE subcommand would format packet trace output for token-ring analysis. Because of the phaseout of token-ring technology, beginning with z/OS V1R10, the TOKENRING keyword is no longer supported.

Migration action: Use the ETHERNET keyword instead of the TOKENRING keyword when formatting the SNIFFER data set trace. ETHERNET formats output for Ethernet analysis and is the default. If you specify TOKENRING, the keyword is ignored and ETHERNET is in effect.

IP Services: Modify FTPOSTR exit routines to handle a new parameter (Required-IF, as of R10)

Required if you have installed an FTPOSTR exit routine and it expects 18 or fewer parameters.

The FTPOSTPR exit routine is called upon completion of the FTP commands RETR, STOR, STOU, APPE, DELE, and RNTD. In z/OS V1R10, the exit routine has a new parameter: a pointer to a buffer containing the complete text of the server reply sent to the client. If you have an FTPOSTPR exit routine that cannot tolerate this new parameter, you must modify the routine.

Migration action: Modify your FTPOSTPR exit routine to tolerate 19 parameters.

IP Services: Update the timestamp of a user-customized FTP message or reply catalog (Req-IF, as of R9)

Required if you are using a modified ftpdrply.cat or ftpdmsg.cat message catalog for FTP messages and replies.

Starting in z/OS V1R9, the FTP application checks to verify that the time stamp included in the catalog is the same as the time stamp that is used for its default messages. If these do not match, FTP will use its default messages. Before

z/OS V1R9, FTP used messages from the catalog without verifying that the catalog time stamp matched the time stamp of the catalog used to build FTP.

Migration action: Follow this procedure to ensure that the time stamp in your locally modified catalog matches the time stamp in the FTP product catalog:

1. Convert the product FTP reply or message catalog to a format that can be edited.
2. Update the time stamp in the catalog.
3. Make local changes.
4. Rebuild the catalog.

IP Services: Allow the IP CICS Socket Listener to remain active when its stack is not available (Recommended)

Recommended for improved availability.

Before z/OS V1R9, the IP CICS Socket Listener would end when its TCP/IP stack was either initially down or recycled.

In z/OS V1R9, a new configuration option, RTYTIME, allows the Listener to remain active when the TCP/IP job with which it has affinity is initially down or was recycled. If the RTYTIME configuration option is not specified, the Listener defaults to an RTYTIME value of 15 seconds, meaning that the Listener will retry connecting to its TCP/IP stack after a delay of 15 seconds has gone by.

If you want the Listener to end when its stack is not available, as it did before z/OS V1R9, you have to specify an RTYTIME of 0.

Migration action: To cause the IP CICS Socket Listener to end when its stack is initially not active or ends, which is the pre-z/OS V1R9 behavior, specify RTYTIME=0 in the Listener configuration using either the EZACICD assembler macro or the EZAC configuration transaction.

To accept the new behavior, do nothing, which allows RTYTIME to default to 15 seconds, allowing the Listener to retry connecting to its TCP/IP stack after a delay of 15 seconds has gone by. Of course, you can specify a value other than 15 for RTYTIME.

Note: Use of a pre-z/OS V1R9 Listener configuration file in z/OS V1R9 causes the Listener to continue when its connection with its stack is initially not active or the connection ends. A value of 15 seconds is used for RTYTIME.


IP Services: Recompile customer-written listeners (Required-IF, as of R9)

Required if your IP CICS Socket-enabled applications use any of the control blocks contained in the EZACICA macro.



The IP CICS Socket listener control area (LCA) control block changed. The changes are particular to the management of the IP CICS Socket listener and they include needed expansion for future additions.

Migration action: If you wrote your own listeners to use the IP CICS Socket interface management transaction, EZAO, you must recompile those listeners.

z/OS 1.10 Migration Actions




DFSMS Migration Actions for z/OS R10 from z/OS R8

- **Migration Actions You Can Do NOW:**
 - 
Redefine existing VSAM data sets that contain the IMBED, REPLICATE, and KEYRANGE attributes (Recommended)
 - IMBED and REPLICATE support are planned to be removed in a future release. KEYRANGE isn't recommended to use.
 - Determine which VSAM data sets are affected, then schedule a time to redefine them.
 - Use tools (IMBDSHIP and Migration Checker).
 - In a future release of z/OS, when DFSMSshm or DFSMSdss recalls or restores a VSAM data set with either IMBED or REPLICATE attribute or both, the attributes will be removed.
 - 
Specify a control data set ID for DFSMSrmm (Required, as of R9)
 - A control data set ID is now required, where it was optional previously.
 - Define it using the CDSID operand of the OPTION command in EDGRMMxx, and then set it in the data set using the utility or restarting RMM.

23
© 2008 IBM Corporation

z/OS 1.10 Migration Actions




DFSMS Migration Actions for z/OS R10 from z/OS R8

- **Migration Actions Pre-First IPL:**
 - **Accommodate changes to DEVSERV command output (Required-IF, as of R10)**
 - DEVSERV QDASD output is changed: cylinder column expanded, new column heading ECK and is reduced to three characters.
 - DEVSERV PATHS output is changed: adds the number of cylinders in a new column called CYL, ALT column has been replaced with the CYL column.
 - Both outputs have been shifted one position to the right.
 - Change any dependency you have on this changed output.
 - **Accommodate changes in the CA sizes assigned to newly allocated VSAM data sets (Required-IF, as of R10)**
 - The CA size selection is done differently on z/OS VIR10 for all new VSAM data sets (not just those on EAVs).
 - The CA size is 1, 3, 5, 7, 9 or 15 tracks and is determined by the primary and secondary space allocation amounts.
 - As before, you cannot explicitly specify a CA size. Before z/OS VIR10, other possible CA sizes existed. Adjust any primary and secondary requests to accommodate the CA size you rely upon.

24
© 2008 IBM Corporation

z/OS 1.10 Migration Actions




DFSMS Migration Actions for z/OS R10 from z/OS R8

- **Migration Actions Pre-First IPL:**
 - **Allow the dynamic volume count (data class attribute) to exceed 1 when allocating SMS-managed data sets with partitioned organization (Required-IF, as of OA22738)**
 - As of R8 and R9 base level, dynamic volume count could not be greater than 1 when allocating an SMS-managed data set with PO. This was an unpopular change, so it was reversed.
 - Make sure OA22738 is installed to keep this behavior as it was before.
 - **Update automation for changed messages (Required-IF, as of R10)**
 - Many many messages have changed: message text, messages have been added, changed, and removed.
 - Make sure automation can handle these changes.

25
© 2008 IBM Corporation

z/OS 1.10 Migration Actions




DFSMS Migration Actions for z/OS R10 from z/OS R8

- **Migration Actions Post-First IPL:**
 - **Update programs or procedures that depend on output in reports from the LISTDATA PINNED command (Required-IF, as of R10)**
 - Track addresses in output of IDCAMS LISTDATA PINNED are in a different format, regardless of volume size.
 - As of R10: a range is identified, and the track address is printed in 8 hex digits.
 - **Accommodate the change of IDCAMS DEFINE for RECORDS involving VSAM linear data set (Required-IF, as of OA21369 on R9)**
 - Before OA21369: when space for linear data sets was calculated for the DEFINE command with the RECORDS parameter, an implied CI size of 4096 was used when the CI size was larger than 4096.
 - As of OA21369: in the same scenario, the actual larger CI size is used instead of 4096.
 - Consider adjusting the value of RECORDS when a linear data set is defined with CISIZE greater than 4096. Using the same IDCAMS DEFINE CLUSTER job will result in the allocated space being larger than before because the actual CISIZE (instead of 4096) has an influence on the amount of space allocated to the data set. You need to decrease the value of RECORDS to achieve an allocation amount similar to before.
 - Install DB2 PTF PK42840 to avoid DB2 side effects from this change.

26
© 2008 IBM Corporation

z/OS 1.10 Migration Actions



DFSMS Migration Actions for z/OS R10 from z/OS R8

- **Migration Actions Post-First IPL:**
 - **Use the new version of the DEFrag command in DFSMSdss (Recommended)**
 - Before R10: DEFrag command contained a CONSOLIDATE keyword.
 - As of R10: a new DEFrag command is introduced (for EAV usage), which drops the CONSOLIDATE keyword. Replacement for the CONSOLIDATE keyword on the new DEFrag command is the CONSOLIDATE command. Need facility class profile access to STGADMIN.ADR.CONSolid and STGADMIN.ADR.DEFRag to use.
 - Old DEFrag command is still available, when specifying the VERSIONI keyword (which isn't for EAV usage).
 - Old JCL that uses CONSOLIDATE with DEFrag (and without VERSIONI) will continue to run, however CONSOLIDATE command will be executed first followed by the DEFrag command.
 - **Define facility class names to protect (H)BACKDS NEWNAME commands (Required-IF, as of R10)**
 - As of R10: new hsm facility class names protect the BACKDS NEWNAME and HBACKDS NEWNAME commands. Also, authorization for data set name on NEWNAME is checked. New facility class names must be defined to protect from unintended use of this support by unauthorized users.
 - Define new facility class profiles to authorize to the (H)BACKDS commands:
 - STGADMIN.ARC.BACKDS.NEWNAME
 - STGADMIN.ARC.ENDUSER.HBACKDS.NEWNAME

27
© 2008 IBM Corporation

DFSMS Migration Actions Between z/OS V1 R8 and z/OS V1 R10

These migration actions were taken from z/OS *Migration*. Some descriptions and actions have been shortened for inclusion in this presentation. For the complete descriptions and actions, refer to z/OS *Migration*.

DFSMS Migration Actions You Can Do Now



DFSMSdftp: Back up SMS control data sets (Recommended)

Recommended to ensure data integrity.

Migration action: Do the following on your pre-z/OS V1R10 systems:

1. Back up SMS control data sets according to established procedures in the event that fallback is required. The control data set format is VSAM linear.
2. Install all coexistence PTFs defined for z/OS V1R10.

In addition, if you modified and activated a higher-level policy on a pre-z/OS V1R10 system, do the following to ensure that the ACDS can be accessed on z/OS V1R10:

1. On the pre-z/OS V1R10 system, save the active ACDS as an SCDS with the SETSMS SAVESCDs command.
2. On z/OS V1R10, update, translate, validate, and activate the saved SMS policy.

Note: With z/OS V1R10 (and prior releases by APAR OA22373), CDS uplevel checking only checks the formatted release against the updating release level. The construct size change no longer causes warning messages..



DFSMSdftp: Redefine existing VSAM data sets that contain the IMBED, REPLICATE, and KEYRANGE attributes (Recommended)

Recommended to avoid degraded performance and wasted DASD space.

No supported release of z/OS honors the IMBED, REPLICATE, and KEYRANGE attributes for new VSAM data sets. In fact, using these attributes can waste DASD space and often degrades performance. Servicing these VSAM data sets has become increasingly difficult. In some cases, unplanned outages have occurred. For these reasons,

IBM recommends that you stop using IMBED and REPLICATE, and that you minimize or eliminate your use of KEYRANGE. IMBED and REPLICATE were intended as performance improvements and have been obsoleted by newer, cached DASD devices. Striped data sets provide much better performance than KEYRANGE and should be viewed as a candidate for any existing KEYRANGE data sets.

Migration action:

1. Determine which VSAM data sets were defined with the IMBED, REPLICATE, or KEYRANGE attribute. To help you perform this task, you can get a tool that reads existing VSAM data sets and reports which ones have these attributes. The tool is available from the software server (<ftp://software.ibm.com>) in the `s390/mvs/tools` directory as `IMBDSHIP.JCL.TRSD`. Download the file in binary format and uncompress it on your z/OS system using `AMATERSE` or `TRSMMAIN`. Instructions for using the tool are included in the downloaded `JCL`. See APAR II13894 for additional information.

Notes:

- The tool only checks data sets that are on DASD. Data sets that were migrated by DFSMSHsm or backed up by DFSMSdss continue to retain their IMBED, REPLICATE, and KEYRANGE attributes when recalled or restored. In a future release of z/OS, when support for IMBED and REPLICATE is removed and data sets are recalled or restored by DFSMSHsm or DFSMSdss, the IMBED and REPLICATE attributes will be removed during the recall or restore and the data sets will be in a usable state. Because the data sets will be in a usable state, there is no need for a tool to identify data sets that were backed up or migrated at a time when the data sets had the IMBED or REPLICATE attribute and support for IMBED and REPLICATE still existed.
 - “AMATERSE” and “TRSMMAIN” are names for a service aid that compresses and decompresses data exchanged with IBM. “AMATERSE” is the preferred program name since its integration into z/OS V1R9. (AMATERSE is also available for z/OS V1R7 and V1R8 by PTFs UA36927 and UA36928 for APAR OA19194.) “TRSMMAIN” is the original program name and is now shipped as an alias entry point to AMATERSE. For more information about AMATERSE, including several differences with TRSMMAIN, see *z/OS MVS Diagnosis: Tools and Service Aids*.
2. Schedule a time for the affected VSAM data sets to be unavailable, and redefine them.

Please note that while no announcement has been made of the specific date or release that data sets with these attributes will no longer be able to be opened, it would be prudent to identify the data sets in your installation and begin converting them to versions that do NOT contain these attributes.



DFSMSrmm: Specify a DFSMSrmm control data set ID (Required, as of R9)

Before z/OS V1R9, a control data set ID was optional. If you did not define one in z/OS V1R8 (with the PTF for DFSMSrmm APAR OA17965 installed), warning message EDG0229E NO CDSID WAS SPECIFIED FOR OPTION IN PARMLIB was issued but DFSMSrmm initialization did not fail.

Starting with z/OS V1R9, a control data set ID is required. You must define it using the CDSID operand of the OPTION command in parmlib member EDGRMMxx, and then set it in the control data set either by using the EDGUTIL utility or by restarting DFSMSrmm. If you do not define a control data set ID, you cannot start DFSMSrmm, and warning message EDG0237E MISSING IDENTIFIER FOR THE CONTROL DATA SET is issued.

Migration action:

1. If you have entries in GRSRNLxx for QNAME(SYSZRMM) RNAME(MASTER.RESERVE), complete the steps in “DFSMSrmm: Update GRSRNLxx” before continuing here.
2. Define a control data set ID in parmlib member EDGRMMxx by using the CDSID operand of the OPTION command. Do this on each system in the RMMplex. Use a different CDSID for each control data set.
3. Set the control data set ID in the control data set by using the EDGUTIL utility or by restarting DFSMSrmm on one system.

DFSMSHsm: Define the BCDS with a maximum record size of 6544 for the FRBACKUP command (Required-IF, as of R9)

Required if you plan to issue an FRBACKUP command.

DFSMSHsm control data set records created for the FRBACKUP function can be up to 6544 bytes in length. In order for an FRBACKUP command to complete successfully, the maximum record size of the BCDS must be defined as 6544. While an FRBACKUP command that processes only a limited number of volumes may process successfully with a smaller record size, this environment is not recommended. In z/OS V1R9, if the BCDS maximum record size

Migrating to z/OS 1.10 - Part 2

is less than 6544, DFSMSshsm will fail FRBACKUP commands with message ARC1843I FAST REPLICATION BACKUP FUNCTIONS WILL NOT BE PERFORMED. BCDS MAXIMUM RECORD SIZE MUST BE 6544.

Migration action: Follow the standard procedures for reorganizing the DFSMSshsm BCDS. Define all BCDS clusters with RECORDSIZE(334 6544) and CISIZE(12288).

DFSMSrmm: Use the new default for the EDGHSKP XREPTTEXT DD statement (Recommended, as of R10)

Recommended because the new default requires less DASD space and fewer resources to create the extract.

The default processing for the EDGHSKP XREPTTEXT DD statement is changed so that only extended records are created. Before z/OS V1R10, DFSMSrmm report extract processing used allocated ddnames to determine whether extended records were required. If the ddname was REPTTEXT, DFSMSrmm created all records except the extended records in the extract. If the ddname was XREPTTEXT, DFSMSrmm created all records in the extract and then used the D and V records to create extended records.

Starting in z/OS V1R10, by default, the XREPTTEXT DD statement receives only extended (X) records. To tailor the contents of the REPTTEXT and XREPTTEXT DD statements, DFSMSrmm report extract processing now checks for the existence of the RPTTEXT command in the SYSIN file, and if the RPTTEXT command is found, DFSMSrmm writes the selected extract records to the data set specified by either the REPTTEXT or XREPTTEXT DD statement.

The new default of only X records matches what most customers want. In addition, the new default requires less DASD space and fewer resources to create the extract.

Migration action: To obtain the new behavior, which is recommended, do nothing.

If you want the prior behavior, which is for the XREPTTEXT DD statement to contain all records, not just X records, update any report extract JCL used to create a report extract file containing extended records (run to create records other than the extended records) by adding the following statements to your existing JCL:

```
//SYSIN DD *  
RPTTEXT RECORDS(X,O,R,B,S,P)  
/*
```

This change must be made to your JCL before IPLing z/OS V1R10 and running extract, but it can be made on your current (pre-z/OS V1R10) release (with the coexistence PTF installed if z/OS V1R9) because it will only be used by DFSMSrmm on z/OS V1R10.

DFSMSrmm: Update GRSRNLxx (Required, as of R9)

Starting with z/OS V1R9, an additional ENQ name is used by DFSMSrmm for CDS serialization. The new ENQ name now includes the CDSID, which is now a required parmlib option. If you have a GRSRNLxx entry for DFSMSrmm, you must either convert it to a generic RNL entry or add an additional specific RNL entry. Furthermore, if you add an additional specific RNL entry, the RNAME value specified in that entry must be of the correct length. If you do not update existing GRSRNLxx entries, or do not specify the RNAME value correctly, there could be performance or serialization problems related to the CDS volume.

DFSMSrmm uses both the existing and the new ENQ names, so if you have existing GRSRNLxx entries for DFSMSrmm you must update them to handle the serialization the same way for both ENQ name cases. The new serialization is enabled automatically the first time z/OS V1R9 or later DFSMSrmm is started.

Migration action:

1. If you have entries in GRSRNLxx for QNAME(SYSZRMM) RNAME(MASTER.RESERVE), update those entries as shown in the following step. Do this on all coexisting and fallback systems and on your planned z/OS V1R10 system before the first IPL. If you do **not** have entries in GRSRNLxx for QNAME(SYSZRMM) RNAME(MASTER.RESERVE), you have no action to take; DFSMSrmm CDS serialization will be based on both a hardware reserve and SYSTEMS ENQs.
2. Update existing entries in GRSRNLxx for QNAME(SYSZRMM) RNAME(MASTER.RESERVE). The simplest way to do this is by using a generic RNAME entry, that is, by changing the existing definition from TYPE(SPECIFIC) to TYPE(GENERIC). Here is an example for an exclusion RNL:

```
RNLDEF RNL(EXCL) TYPE(GENERIC) QNAME(SYSZRMM) RNAME(MASTER.RESERVE)
```

The same change is required for any conversion RNLs you might have for DFSMSrmm. For example:

```
RNLDEF RNL(CON) TYPE(GENERIC) QNAME(SYSZRMM) RNAME(MASTER.RESERVE)
```

Updating the entries in this way ensures that, regardless of your CDSID, the correct exclusion or conversion action is always taken by global resource serialization and both the original and the new ENQ names are handled the same way.

An alternative to creating generic RNL entries is to add a second specific RNL entry that includes the CDSID, as shown in the following RNL(EXCL) example. In this case, you would replace *cdsid* in RNAME with your selected CDSID as specified in the EDGRMMxx parmlib member OPTION statement:

```
RNLDEF RNL(EXCL) TYPE(SPECIFIC) QNAME(SYSZRMM) RNAME(MASTER.RESERVE)  
RNLDEF RNL(EXCL) TYPE(SPECIFIC) QNAME(SYSZRMM) RNAME(MASTER.RESERVE.cdsid)
```

Requirement: The length of RNAME must be exactly 23 characters. If the length of your CDSID is less than eight characters, you should use a generic entry as described above. If you do not use a generic entry, you must specify RNAME as the hexadecimal constant RNAME('X'D4C1E2E3C5D94BD9C5E2C5D9E5C54Bxx..xx4040'), where X'D4C1E2E3C5D94BD9C5E2C5D9E5C54B' represents MASTER.RESERVE and xx..xx4040 represents your CDSID appended with blanks to eight characters in length. Update the GRSRNLxx entries on each system before implementing a control data set ID and before starting DFSMSrmm on z/OS V1R9 or later.

DFSMS Migration Actions Pre-First IPL

DFSMSdfp: Ensure that the Language Environment run-time library is available for DLLs (Required-IF)

Required if your installation builds or references DLLs.

Language Environment provides common services and language-specific routines in a single run-time environment. You can use Language Environment to build and use dynamic link libraries (DLLs) for applications.

Migration action: If your installation builds or references DLLs, either you must set up the system link list to refer to the Language Environment run-time libraries (SCEERUN and SCEERUN2), or each job that creates or uses a DLL must include a STEPLIB DD statement referencing these libraries.

DFSMSdss: Build the IPLable stand-alone DFSMSdss image (Required-IF)

Required if you intend to use the Stand-Alone Services provided by DFSMSdss.

If you intend to use the Stand-Alone Services provided by DFSMSdss, you must use the DFSMSdss BUILDSA function to create the Stand-Alone Services IPL-capable core image

Migration action:

1. Prepare for Stand-Alone Services by creating a Stand-Alone Services IPL-able core image with the BUILDSA command. With the BUILDSA command you can specify the device (card reader, tape drive, or DASD volume) from which Stand-Alone Services will be IPLed. You can also specify the operator console to be used for Stand-Alone Services. The BUILDSA function builds the IPL-able core image under the current operating system and determines a record size based on whether the IPL is from card, tape, or DASD.
2. Use RACF or another external security system to protect the SYS1.ADR.SAIPLD.Vvolser data set and the Stand-Alone Services modules.
3. If you haven't done so already, make a backup copy of your system that can be restored by this function. For information about backing up volumes, see *z/OS DFSMSdss Storage Administration Guide*.

DFSMSdfp: Ensure your SYS1.IMAGELIB is customized for your printing environment (Required-IF)

Required if you are not using your old SYS1.IMAGELIB, you are installing with ServerPac or SystemPac, and you are using line mode printers such as the 3800 or 3900.

If you use line mode printers such as the IBM 3800 or the IBM 3900 running in line mode (not page mode), you must install library character sets, graphic character modification modules, and character arrangement tables in SYS1.IMAGELIB. This migration action does not apply if you are using IBM 3900 printers that are driven by PSF.

Migration action:

1. Run the LCSBLD1 job from the samplib data set to create character sets, graphic character modification modules, and character arrangement tables in SYS1.IMAGELIB.
2. Copy customized or locally-written FCBs and UCS images from your old system's SYS1.IMAGELIB data set to the new system's SYS1.IMAGELIB data set.

DFSMSdss: Replace the INDD and INDYNAM keywords in COPY DATASET and DUMP DATASET jobs (Recommended)

Recommended if your jobs issue the COPY DATASET or DUMP DATASET commands and you do not want to receive informational messages stating that DFSMSdss has replaced INDD and INDYNAM with the correct keywords.

The INDD or INDYNAM keyword on a COPY DATASET or DUMP DATASET command is used to perform either a logical data set copy or a physical data set dump, depending on the command.

Starting in z/OS V1R8, the INDD and INDYNAM keywords are being replaced as follows:

- For uses of the COPY DATASET command, the INDD and INDYNAM keywords are replaced with the LOGINDD and LOGINDYNAM keywords, respectively.

- For uses of the DUMP DATASET command, the INDD and INDYNAM keywords are replaced with the PHYSINDD and PHYSINDYNAM keywords, respectively.

You can still use INDD and INDYNAM on z/OS V1R8 and higher. However, if you do, you will receive an informational message stating that DFSMSdss has replaced INDD or INDYNAM with the correct keywords. To avoid receiving such messages, review and update any DFSMSdss jobs that issue these commands to use the correct keywords.

Note: If the PTF for APAR OA20242 (for R8 and R9) is not installed, message ADR146I is not issued for COPY DATASET with INDD, COPY DATASET with INDYNAM, or DUMP DATASET with INDYNAM. (The PTF does not affect DUMP DATASET with INDD.)

Migration action: Check your DFSMSdss jobs for usage of the COPY DATASET and DUMP DATASET commands. Update the JCL, as follows:

- For uses of the COPY DATASET command, replace any occurrences of the INDD or INDYNAM keywords with the LOGINDD or LOGINDDYNAM keywords, respectively.
- For uses of the DUMP DATASET command, replace any occurrences of the INDD or INDYNAM keywords with the PHYSINDD or PHYSINDYNAM keywords, respectively.

DFSMSdfp: Provide a 140-byte work area when using OBTAIN (Required-IF, as of R10)

Required if you have programs that do not specify at least a 103-byte DSCB return area.

Programs must pass the address of a 140-byte return area when they search for a data set name using the OBTAIN macro with the CAMLST SEARCH option or the OBTAIN SVC 27 with the SEARCH for DSNNAME operation code. In releases prior to z/OS V1R10, the system returned data in only the first 101 bytes of the return area. Beginning with z/OS V1R10, the system will return data in the first 103 bytes. Therefore, you should upgrade your programs to provide the entire 140-byte return area; otherwise, you might encounter storage overlay problems or program interruptions.

Migration action: Upgrade programs to provide the entire 140-byte return area as stated in the reference information below.

DFSMSdfp: Accommodate new defaults for z/OS Global Mirror (XRC) parameter values (Req-IF, as of R10)

Required if you rely on the previous XRC default values.

Default values have changed for two parameters of the SHADOW statement in the XRC parmlib member. Before z/OS V1R10, the RequireUtility parameter had a default value of NO, and the UtilityDevice parameter had a default value of FLOAT. In z/OS V1R10, these default values have changed as follows:

- The default value of the RequireUtility parameter is now YES, which prevents XADDPAIR of a non-XRCUTL volume when an active storage control session does not exist for the specified (or default) SCSESSION. The previous default of NO enables XADDPAIR of a non-XRCUTL volume to proceed when an active storage control session does not exist.
- The default value of the UtilityDevice parameter is now FIX, which changes the method used for selecting a utility device. The FIX value causes XRC to use the most recently added XRCUTL volume as the utility device for the storage control session. The previous default of FLOAT allows the storage control to dynamically pick the primary volume to be used as the utility device for the storage control session. The nonreserved device with lowest I/O activity will typically be selected.

Migration action: If your XRC sessions rely on the previous default values, you must add statements in your XRC parmlib to specify those values, thus overriding the new defaults.

DFSMSdfp: Update programs that process SMF record type 19 (Required-IF, as of R10)

Required if you rely on the previous SMF 19 record length.

In z/OS V1R10, the length of SMF type 19 records has increased. A type 19 record is created for each DASD volume that is online when IPL occurs, when a HALT EOD or SWITCH SMF command is issued, and when a DASD is varied offline. The type 19 record describes the space usage on the volume, and new fields have been added to the record for track-managed free space statistics and volume size information. Programs that process this record type might need to be modified to support the expanded length of the record or to utilize the new fields. SMF type 19 records are recorded by LSPACE processing when SMF=YES or SMF=TEST is specified on the LSPACE macro invocation. YES specifies that the caller wants an SMF record containing volume information to be written. TEST specifies that LSPACE is to test for an active SMF system that has been defined to request volume accounting before writing the SMF record.

Migration action: Check any programs that process SMF type 19 records to determine if you must make any changes. If necessary, modify the programs to use the new record length.

DFSMSdftp: Accommodate changes to DEVSERV command output (Required-IF, as of R10)

Required if you have a dependency on the output of the DEVSERV QDASD or DEVSERV PATHS command.

z/OS V1R10 includes several changes to output from the DEVSERV command:

- DEVSERV QDASD output is changed as follows: the CYL column was expanded to accommodate a nine-digit number for cylinders. In addition, the extended function consistency check column EF-CHK is reduced to three characters. The new column heading is ECK.
- DEVSERV PATHS output is changed to add the number of cylinders, in a new column labeled CYL. To accommodate the largest possible number of cylinders for a device (nine digits) in this new column, the original ALT column is replaced with this new CYL column and unnecessary spaces around the PIN and DDC columns are removed.
- The output for both DEVSERV QDASD and PATHS has been shifted one position to the right, leaving the first character position blank.

Migration action: Change any dependency you have on the output of DEVSERV QDASD or DEVSERV PATHS. Examples of the output changes can be found in *z/OS Migration*.

DFSMSdftp: Update programs that use DCOLLECT (Required-IF, as of R10)

Required if you use DCOLLECT to process data set (type D) records to determine whether data sets are overallocated.

In a DCOLLECT type D record, the DCDOVERA field is a 4-byte field that reports on the amount of overallocated space for a data set. Before z/OS V1R10, the field was defined as a 31-bit signed number. Beginning with z/OS V1R10, DCDOVERA is defined as a 32-bit unsigned number in order to accommodate larger values.

Migration action: If you use the IDCDOUT macro to map DCOLLECT records, reassemble your programs to pick up the current mapping.

If you do not use the IDCDOUT macro to map DCOLLECT records, be aware that the field definition has changed for DCDOVERA from signed to unsigned. Use the new field definition when you create DCOLLECT reports.

DFSMSdftp: Accommodate changes in the CA sizes assigned to newly allocated VSAM data sets (Required-IF, as of R10)

Required if you have a dependency on the CA size selected for VSAM data set and the corresponding primary or secondary space amounts.

z/OS V1R10 includes a change in control area (CA) size selection for all new allocations of VSAM data sets on any volume type (not only extended address volumes). The system selects a control area size of 1, 3, 5, 7, 9, or 15 tracks. Previously, the system selected from a wider variety of possible CA sizes. As previously, you cannot explicitly specify control-area size. Generally, the primary and secondary space allocation amounts determine the control-area size. For requests where either the primary or secondary allocation amount is smaller than one cylinder, the system might adjust the primary and/or secondary quantity. The system might also select a CA size that is different from what was selected in a prior release. For example, a TRK(24,4) request results in a control area of 5 tracks, and primary and secondary amounts of 25 and 5 tracks, respectively.

Migration action: CA size is derived from the requested primary and secondary allocation. Because the CA size is computed differently in z/OS V1R10, adjust any primary and secondary allocation requests to accommodate the CA size you rely upon.

If you have any programs that depend upon a specific CA size that is no longer supported, change the programs to accept a CA size of 1, 3, 5, 7, 9, or 15 tracks.

DFSMSdftp: Tell users that the ISMF SORTREC line operator is no longer available (Required-IF, as of R10)

Required if you use the SORTREC line operator.

Beginning with z/OS V1R10, the DFSORT interactive storage management facility (ISMF) panels are no longer supported. Consequently, the SORTREC line operator is no longer available.

Migration action: Tell users that the SORTREC line operator is no longer available. Trying to use it will result in the message COMMAND SORTREC NOT FOUND. An alternative to SORTREC is to use DFSORT against the data set from a batch job.

DFSMSdftp: Allow the dynamic volume count (data class attribute) to exceed 1 when allocating SMS-managed data sets with partitioned organization (Required-IF, as of OA22738)

Required if you use data classes that specify dynamic volume counts when allocating data sets with partition organization, and you want to allow the dynamic volume count (data class attribute) to exceed 1 (which is the pre-z/OS R8 situation).

Before z/OS V1R8, when allocating an SMS-managed data set with partitioned organization, you could assign to it a data class that specified a dynamic volume count greater than 1. Then, in z/OS V1R8 and V1R9, the base code was changed so that the dynamic volume count could not be greater than 1. However, APAR OA22738 on z/OS V1R8 and V1R9 restored the original situation, that is, the dynamic volume count could be greater than 1 again. Now, in z/OS V1R10, the original (pre-z/OS V1R8) situation (greater than 1 allowed) has been permanently restored. This migration action is not relevant to non-SMS data sets because the dynamic volume attribute only applies to SMS-managed data sets.

Migration action: If you desire, allow ACS routines to select a data class with a dynamic volume count greater than 1 for SMS-managed data sets with partitioned organization. Note that because PDSs and PDSEs can only exist on one volume (and cannot be extended to a second volume), the dynamic volume count will be ignored.

DFSMSdfp: Update automation for changed multiline WTO messages (Required-IF, as of R9 with APAR)

Required if you have automation which examines the message text of messages described.

In z/OS V1R8 without the PTF for APAR OA15780 installed, and z/OS V1R9 without the PTF for APAR OA21810 installed, some DFSMSdfp IEAxxxx messages are formatted using multiple single-line WTO messages when the length of the message requires more than one line. The second WTO message includes the text “[CONTINUATION OF IEAXXXX”, followed by the text “SNS=xxxxxxxx...].”

Starting with z/OS V1R8 with the PTF for APAR OA15780 installed, and z/OS V1R9 with the PTF for APAR OA21810 installed, the complete message is formatted using a single multiline WTO message, and the text “SENSE=xxxx xxxx ...”, if applicable, is included as part of the multiline message. The text “[CONTINUATION OF IEAXXXX” is no longer provided.

Example in z/OS V1R8 without the PTF for APAR OA15780 installed, and z/OS V1R9 without the PTF for APAR OA21810 installed:

```
IEA442E dev REPORTS DISABLED INTERFACE ON nn -- FAULT CODE = cccc -  
NOTIFY CE  
[CONTINUATION OF IEA442E SNS=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx]
```

Example in z/OS V1R8 with the PTF for APAR OA15780 installed, z/OS V1R9 with the PTF for APAR OA21810 installed, and later:

```
IEA442E dev REPORTS DISABLED INTERFACE ON nn -- FAULT CODE = cccc -  
NOTIFY CE,SENSE=xxxx xxxx xxxx xxxx xxxx xxxx xxxx
```

Note that the word “SENSE” is now spelled out, and the 32 bytes of sense data have a blank character after every four sense bytes.

The specific messages affected are IEA442E, IEA448I, IEA449A, IEA451I, IEA452I, IEA453I, IEA454E, IEA455E, IEA457I, IEA458I, IEA459I, IEA468I, IEA469E, IEA473I, IEA474E, IEA475E, IEA476E, IEA477I, IEA478E, IEA479E, IEA480E, IEA481I, IEA482I, IEA484I, IEA491E, IEA494I, IEA497I, and IEA498I.

Migration action: Change your automation to expect the new message text format.

DFSMSdfp, DFSMSdss, and DFSMSshsm: Update automation for changed messages (Req-IF, as of R10)

Required if you have automation routines that examine the message text of the messages listed below.

In z/OS V1R10, the text for messages with the following IDs has been changed: ADRxxxx (DFSMSdss), ANTxxxx (DFSMSdfp), ARCxxxx (DFSMSshsm), DMOxxxx (DFSMSdfp), IDCxxxx (DFSMSdfp), IEAxxxx (DFSMSdfp), IECxxxx (DFSMSdfp), IEExxxx (DFSMSdfp), and IGDxxxx (DFSMSdfp).

Some messages have had a period (.) removed from the end of the message text. Other messages have had text added, changed, or removed. These changes can affect automation programs that examine the text of the messages.

Migration action: Update your automation to handle the following DFSMSdfp message changes:

- The message text or explanation has been changed in the following messages to support extended address volumes, including new track, cylinder, or return code information:
 - ANTF073E, ANTF0401E, ANTF0443E, ANTX5117E, ANTX8030W, ANTM6004I, ANTM6005I, ANTM6006E, ANTM6007E, ANTM6013E, ANTM6015I, ANTM6016I, ANTM6020I, ANTM6021I, ANTM6022I, ANTM6023I, ANTM6024I, ANTM6025I, and ANTM8809I
 - DMO0011I, DMO0012I, DMO0013E, DMO0031E, DMO0032E, DMO0033I, DMO0040I, and DMO0041I
 - IDC3009I
 - IEA434I, IEA479E, IEA480E, and IEA480E
 - IEC142I, IEC144I, IEC602I, IEC609I, and IEC987I
 - IEE459I
 - IGD002I, IGD031I, IGD17268I, IGD17279I, IGD17269I, IGD17318I, and IGD17389I
- Message IGD17038I now provides a number of volumes (“n volumes”) in place of the words “this request”.

Update your automation to handle the following **DFSMSdss** message changes:

- The period has been removed from the end of the following messages: ADR021I, ADR032W, ADR037E, ADR045T, ADR109I, ADR146I, ADR148I, ADR204W, ADR238I, ADR239I, ADR240W, ADR226W, ADR297I, ADR299E, ADR301E, ADR325E, ADR343E, ADR344I, ADR345D, ADR359W, ADR363E, ADR365E, ADR388W, ADR390I, ADR397E, ADR401W, ADR405E, ADR444E, ADR459E, ADR499I, ADR701E, ADR702E, ADR711I, ADR715W, ADR716W, ADR719E, ADR721I, ADR731W, ADR733I, ADR734I, ADR736E, ADR737W, ADR767I, ADR785E, ADR795E, ADR796E, ADR801I, ADR804W, ADR805E, ADR865E, ADR880I, ADR901E, ADR902I, ADR903I, ADR904I, ADR906I, ADR907I, ADR908E, ADR912I, ADR913E, ADR916W, ADR940I, ADR941E, ADR942E, ADR943E, ADR944E, ADR946E, ADR969E, ADR970E, ADR974I, ADR978W, ADR979E, ADR985W, ADR990E, and ADR993E.
- The message text or explanation has been changed in the following messages to support extended address volumes, including new track, cylinder, or return code information: ADR024E, ADR037E, ADR037E, ADR040I, ADR071I, ADR145I, ADR152E, ADR201E, ADR210E, ADR226W, ADR232E, ADR234I, ADR235I, ADR247W, ADR250I, ADR251W, ADR255E, ADR256E, ADR258I, ADR260I, ADR261I, ADR262I, ADR299E, ADR309E, ADR317I, ADR336I, ADR345D, ADR364W, ADR366W, ADR373E, ADR379E, ADR433E, ADR436E, ADR441W, ADR498E, ADR520I, ADR557I, ADR558I, ADR735W, ADR817W, ADR818E, ADR887I, ADR888I, ADR932W, ADR935W, ADR938E, ADR939E, ADR945W, ADR946I, and ADR973E. (The following messages have also been changed but they will never be analyzed by automation because they are only issued during stand-alone restore when z/OS is not in operation: ADRY0500I, ADRY0700I, ADRY1500W, ADRY1524W, ADRY1711I, ADRY2500I, and ADRY3521I.)
- The ending period has been removed and additional message text has been added to messages ADR037E and ADR345D.
- Message ADR049E has been enhanced to include abend reason code information at the end.
- References to JOBCAT and STEPCAT have been removed from messages ADR380E, ADR415W, ADR470W, ADR472E, and ADR485E.

Update your automation to handle the following **DFSMShsm** message changes:

- The message text or explanation has been changed in the following messages to support extended address volumes, including new track, cylinder, or return code information: ARC0120I, ARC0157I, ARC0185I, ARC0269I, ARC0638I, ARC0742I, ARC1309I, ARC1359I, ARC1520I, and ARC1524I.

DFSMSrmm: Define default partitioning support (Recommended, as of R10)

Recommended if you have no REJECT commands in the DFSMSrmm parmlib so that you get, by default, the desired type of volumes added to the DFSMSrmm control data set when the volumes are not defined to DFSMSrmm.

Before z/OS V1R10, the DFSMSrmm default processing for volumes that were not defined to DFSMSrmm was to add only system-managed volumes, and not non-system-managed volumes, to the DFSMSrmm control data set. Starting in z/OS V1R10, the DFSMSrmm default processing when PRTITION commands are in use, either because they are defined or because there are no REJECT commands, is to add all undefined volumes. If you do not use REJECT commands in parmlib, you must add a PRTITION command if you want to ensure that the DFSMSrmm default processing stays the same.

Migration action:

- To ensure that the DFSMSrmm default processing stays the same, add the following PRTITION command to parmlib: PRTITION VOLUME(*) TYPE(NORMM) NOSMT(IGNORE)
As a result, system-managed volumes are automatically added to the DFSMSrmm control data set during library entry/insert and OPEN processing, but non-system-managed volumes are not added automatically. Note that when adding volumes automatically, DFSMSrmm no longer creates a rack number but will use an empty rack number if it exists.
- To get the new behavior, do one of the following:
 - If you have no REJECT commands in parmlib, take no action. The new default is provided by the default PRTITION command created by DFSMSrmm.
 - If you have REJECT commands in parmlib, migrate to PRTITION and OPENRULE commands to exploit the new capabilities.

DFSMSrmm: Update volume replacement policies (Required-IF, as of R10)

Required if you are using the usermod to the EDGMUPD module to override the shipped value.

If you are using the usermod to the EDGMUPD module to set LIMIT=*n*, where *n* is shipped as X'00000001', you can no longer use this usermod. Before z/OS V1R10, you could use the usermod to override the shipped value. The

Migrating to z/OS 1.10 - Part 2

usermod could disable the volume replacement action setting by setting a value of X'00000000'. The usermod was first available with APAR OW43870 and may be called RMREPRL.

Starting in z/OS V1R10, you must use the function of MEDINF REPLACE in parmlib to implement any override or disablement. The usermod no longer works with the EDGMUPD module, and DFSMSrmm processing is based solely on MEDINF REPLACE.

Migration action:

- To disable the replacement of IBM media, you can specify the following in the EDGRMMxx parmlib member: MEDINF NAME(IBM) REPLACE(PERM(0))
- To override the DFSMSrmm built-in default processing of PERM(1), use any of the available REPLACE operand values.

DFSMSrmm: Use DELETE disposition support for tape data sets (Required-IF, as of R10)

Required if you use JOBNAME(DELETED) or DSNNAME('DELETED') in any data set name VRSEs.

Before z/OS V1R10, DFSMSrmm did not consider the disposition of a tape data set. If a job step abended, and CLOSE was entered because of abend processing, DFSMSrmm tracked this in the data set record and provided VRS management by using the ABEND VRSEs.

Starting in z/OS V1R10, DFSMSrmm considers the normal disposition for a tape data set. All tape data sets, whether using system temporary data set names or not, are considered. If the normal disposition (as coded in the JCL or dynamic allocation) is DELETE, the data set record is updated to track the data set as "deleted" by disposition processing. This is a flag just like the "abend" flag. VRSEL processing is updated to process the new restricted "DELETED" VRSEs.

Migration action: If you use JOBNAME(DELETED) in any data set name VRSEs, you must change the VRS to use JOBNAME(DELETED*) to avoid a conflict with the new 'DELETED' VRS support.

If you use DSNNAME('DELETED') in any data set name VRSEs, you must change the VRS to use DSNNAME('DELETED*') to avoid a conflict with the new 'DELETED' VRS support.

DFSMSrmm: Use the backward-compatible CIM provider (Required-IF, as of R10)

Required if you use a CIM client with the DFSMSrmm CIM agent and decide not to update your code to handle the new key formats.

The keys used for the DFSMSrmm CIM classes have changed with z/OS V1R10. If you have CIM clients that have been working on z/OS V1R9 or z/OS V1R8 and you want to use them against z/OS V1R10, and the clients have some hard-coded or specific processing of the keys of DFSMSrmm CIM classes, you can do either of the following:

- Use the backward-compatible CIM provider.
- Update your code to handle the new key formats and then use the z/OS V1R10 CIM provider. If you choose this path, see migration action "DFSMSrmm: Replace CIM providers and CIM classes".

The following table shows the old keys of DFSMSrmm CIM classes and the new compound keys, which have formats of concatenated strings containing the values of the old keys delimited with "+" and appended with spaces by the fix length if needed.

Class Name	Key Transformation
IBMRMM_Dataset	DataSetName,PhysicalFileSequenceNumber, VolumeSerialNumber, and CdsID keys have been replaced with Name key of format DatasetName+FileSeq+Volser+CdsID.
IBMRMM_Location	LocationName, LocationType, and CdsID keys have been replaced with Tag key of format LocationName+LocationType+CdsID.
IBMRMM_LogicalVolume	VolumeSerialNumber and CdsID keys have been replaced with DeviceID key of format Volser+CdsID.
IBMRMM_Owner	OwnerID and CdsID keys have been replaced with InstanceID key of format OrgID:OwnerID+CdsID.
IBMRMM_PhysicalVolume	VolumeSerialNumber and CdsID keys have been replaced with Tag key of format Volser+CdsID.
IBMRMM_PolicyRule	PolicyRuleType, PolicyRuleName, JobNameMask, and CdsID keys have been replaced with PolicyRuleName key of format RuleType+RuleName+JobNameMask+CdsID.
IBMRMM_Product	ProductNumber and CdsID keys have been replaced with IdentifyingNumber key of format ProductNumber+CdsID.
IBMRMM_ShelfLocation	LocationName, MediaName, ShelfLocationNumber, and CdsID keys have been replaced with Tag key of format LocationName+MediaName+Number+CdsID.

Starting in z/OS V1R10, in order to simplify processing of the compound keys, every changed class now has the KeyWithCdsIdName attribute containing the name of its compound key. Additionally, xxxFormat and xxxMask attributes are provided, where xxx is the name of the compound key. For example, TagFormat property of IBMRMM_PhysicalVolume is set to "Volser+CdsID" and TagMask string contains the consecutive concatenation of six blanks, symbol "+", and eight blanks.

Migration action: The backward-compatible CIM provider is provided as rmmcim19.tar.Z compressed tar archive within the /usr/lpp/dfsms/rmm directory. To use it, go to the Miscellaneous Functions panel provided by the rmmutil.sh script and invoke "Unpack RMM CIM Agent for V1R9". As a result, you will have the old agent unpacked in the "v1r9" path within the current directory. For example, if you use the /usr/lpp/dfsms/rmm deployment directory, the new agent will reside in /usr/lpp/dfsms/rmm/v1r9.

DFSMSrmm: Create or update VRSEs for DFSMSrmm data set names with lowercase or mixed case letters (Recommended)

Recommended to ensure that existing data sets with lowercase or mixed case letters are correctly VRS-retained.

Before z/OS V1R9, data set names using lowercase letters, mixed case letters, or special characters were correctly recorded when created, but the DFSMSrmm TSO subcommands did not support the listing, changing, or deleting of data sets that did not follow MVS data set naming conventions. In addition, you had limited capability to retain such data sets through vital record specifications (VRSEs).

Starting with z/OS V1R9, you can use DFSMSrmm to fully record and manage all data sets names that can be specified in job control statements, including those with lowercase or mixed case letters. You can also use all of the DFSMSrmm TSO subcommands and create VRS data set name masks that allow data sets with lowercase or mixed case letters to be VRS-retained.

Migration action: Following the instructions in *z/OS Migration*:

1. Check whether you have any lowercase or mixed case data set names that are on volumes that are VRS-retained.
2. If you have lowercase or mixed case data set names on volumes that are VRS-retained, identify the VRSEs that retain the data set.
3. If any of the matching DSNAMES VRSEs contain uppercase letters that match the lowercase or mixed case letters in the data set names, the VRSEs will no longer match the data set names in z/OS V1R9. You must create new VRSEs or change the existing VRSEs for the data sets.

Data sets that match generic characters in the VRS data set name masks are not affected. While you run VRSEL vital record processing on releases below z/OS V1R9, you should not delete any VRSEs that retain data sets with lowercase or mixed case letters, or you might lose data.

DFSMSrmm: Customize tape unit names to ensure automatic CBRUXVNL processing (Required-IF, as of R8 with APAR OA19374)

Required if you have not yet customized the list of tape unit names..

Starting with z/OS V1R8, you must customize the list of tape unit names to ensure that DFSMSrmm CBRUXVNL processing is provided automatically for tape data sets.

If you did not install the PTF for APAR OA19374 on z/OS V1R8, if you did not customize the list of tape unit names, and if no match is found in the hardcoded tape unit name table, DFSMSrmm ignores the CBRUXVNL processing and your uncataloged tape data set jobs fail.

Migration action:

- To exploit the new tape unit recognition for uncataloged tape data sets, do the following:
 1. Customize the processing by turning on the new code by changing the setting &TAPEDEC SETC to YES.
 2. Customize the hardcoded list of tape unit names to include those used in your installation for your uncataloged tape data sets. The tape unit names are listed in the shipped table called TAPEUNITS.
- To change the decision to issue message EDG81971 for non-RMM managed volumes, customize the CBRUXVNL source code. The decision to issue message EDG81971 for non-RMM managed volumes is made selectable by an option flag in the EDGLCSUP parameter list. The CBRUXVNL exit shipped by DFSMSrmm sets this option flag depending on the circumstances:
 - When a cataloged tape data set is processed, option flag LCSUP_8197 is set to B'1'.
 - When &TAPEDEC is set to YES and the unit name is a tape unit, option flag LCSUP_8197 is set to B'1'.
 - Otherwise option flag LCSUP_8197 is set to B'0'. To turn off message EDG81971, set option flag LCSUP_8197 to B'0'.

DFSMS Migration Actions Post-First IPL

DFSMSdfp: Run OAM DB2 BIND jobs (Required-IF)

Required if you use OAM object support.

When migrating to any new release of z/OS, you must run OAM DB2 BIND jobs if you are using OAM for object support. The BIND jobs update DB2 with new OAM DB2 code.

Migration action: Run the BIND jobs appropriate to your installation:

1. Update and execute the samplib job CBRPBIND (OAM DB2 Bind Package Job).
2. Do one of the following:
 - If your installation starts OAM, uses optical or tape devices, or uses the OAM storage management component (OSMC), do the following:
 - Update and execute samplib job CBRABIND (OAM DB2 Application Plan Bind for LCS and OSR).
 - Update and execute samplib job CBRHBIND (OAM DB2 Application Plan Bind for OSMC).
 - If your installation does not start OAM, use optical or tape devices, or use OSMC, update and execute samplib job CBRIBIND (OAM DB2 Application Plan Bind for OSR only).
3. For more information, see the topic "Migrating, Installing, and Customizing OAM" in *z/OS DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

Note: The following list itemizes the changes to OAM BIND jobs in z/OS V1R10. If you choose to edit a previous version, you must incorporate any new changes as described in the header of each new samplib BIND job:

- CBRPBIND: CBRIEDBS added
- CBRABIND: CBRIEDBS added to CBRIDBS plan
- CBRHBIND: no changes
- CBRIBIND: CBRIEDBS added to CBRIDBS plan.

DFSMSdfp: Examine the job and hardcopy logs for important Delete/Rename and storage constraint messages (Required-IF as of R10)

Required if these messages are not reaching a storage administrator now.

Before z/OS V1R10, some important DFSMS messages could be suppressed or not issued by some components of the system. An example of such a message is IGD17054I DATA SET NOT FOUND FOR DELETE/RENAME ON VOLUME xxxxxx DATA SET IS aaaa.bbbb.cccc.

In z/OS V1R10, DFSMS writes a number of these existing messages to the job log and hardcopy log for the first time. These include messages for Delete and Rename processing, as well as messages pertaining to storage constraints that need to be addressed to prevent possible storage-related failures. This change is made to help ensure that these important messages reach the attention of a storage administrator.

Note: This change does not alter the functioning of the system in any way beyond externalizing messages that may have been suppressed. Also, it applies to messages for SMS-managed data sets as well as non-SMS-managed data sets.

Migration action: On a regular basis, just as frequently as you would today, examine the job and hardcopy logs for important messages related to Delete, Rename, and storage constraint processing.

DFSMSdfp: Update programs or procedures that depend on output in reports from the LISTDATA PINNED command (Required-IF as of R10)

Required if users or programs depend on output from the LISTDATA PINNED command.

The track addresses in the output of the IDCAMS LISTDATA PINNED command are in a different format, regardless of the volume size. Before z/OS V1R10, the report listed each pinned track and its associated data set name.

Beginning with z/OS V1R10, the report identifies a range of pinned tracks associated with a data set. Each track address is printed in eight hex digits in the native format of the device, using the form CCCCcccH. In addition, instead of one track being listed per line, consecutive tracks for each data set are gathered into one line and the range of track addresses is shown.

The meaning of "CCCCcccH" is as follows: "CCCC" is the low-order 16 bits of the cylinder address, "ccc" is the high-order 12 bits of the cylinder address, and "H" is the 4-bit track address. When a 28-bit cylinder address is greater than 65535 (X'FFFF'), the F column in the report contains an asterisk.

Migration action: SYSPRINT and TSO terminal output from IDCAMS are not intended programming interfaces but the changes are stated here as a convenience to help you identify any changes to a program that uses this output. Change any affected program appropriately to parse the new format.

DFSMSdftp: Accommodate the change of IDCAMS DEFINE for RECORDS involving VSAM linear data sets (Required-IF, as of R9 with OA21369)

Required if you define a linear data set with CISIZE greater than 4096 bytes and specify the RECORDS parameter. Previously, when space for linear data sets was calculated in response to an AMS DEFINE command with the RECORDS parameter, an implied control interval size of 4096 was used when the control interval size was larger than 4096. As of z/OS V1R9 with APAR OA21369, in the same scenario, the actual (larger) control interval size is used instead of 4096.

Migration action: Consider adjusting the value of the RECORDS specification when a linear data set is defined with CISIZE greater than 4096 bytes. If you continue to use the same IDCAMS DEFINE CLUSTER job in such a situation, the allocated space will be larger than in z/OS V1R8 or V1R7 because the actual CISIZE (instead of 4096 bytes) has an influence on the amount of space allocated to the data set. You need to decrease the value of the RECORDS specification to achieve an allocation amount similar to before migration.

Notes: 1. DB2 allocates VSAM linear data sets using the RECORDS parameter, so DB2 allocations would be affected by this change when CISIZE greater than 4 KB is used. To avoid this side effect, apply the PTF for DB2 APAR PK42840. With this PTF, the define and extend code in DB2 has been changed to use KB when issuing the IDCAMS DEFINE command.

2. In z/OS V1R8, SMS added a new code to validate the space quantity for volume selection. The space calculation is based on the number of records multiplied by the actual CISIZE instead of the 4 KB record size. If SMS fails the allocation because the primary quantity requested is larger than the total capacity of the largest available volume, the following error message is issued: IGD17279I (n) VOLUMES WERE REJECTED BECAUSE OF INSUFF TOTAL SPACE.

DFSMSdftp: Increase dump space in a Sysplex for VSAM RLS abends (Required-IF as of R9)

Required if the existing dump space is not large enough to accommodate additional dumps.

Before z/OS V1R9, when a VSAM RLS abend occurred, an abend dump was taken only on the originating system where the abend occurred. In z/OS V1R9, new sysplex-wide dumping support causes other systems running VSAM RLS in the sysplex to take dumps as well as the originating system. This change, might result in more dump space being needed on a given system, if dump space is currently limited.

Migration action: If more space for RLS abend dumps is needed, use the same procedures you currently do to increase dump data set size, but be aware that this might need to be done more frequently than in the past. In addition, you can use DAE to suppress dumps if needed.

DFSMSdftp: Add column SUBLEVEL to the DB2 TAPEVOL table for OAM (Required-IF as of R9)

Required if you use OAM.

Before z/OS V1R9, the OAM storage hierarchy consisted of three levels: disk, optical, and tape. In z/OS V1R9, two sublevels are introduced into the tape level, effectively expanding the OAM storage hierarchy into four levels: disk, optical, tape sublevel 1 (TSL1), and tape sublevel 2 (TSL2). You must modify and run the CBRSMR19 migration job to add the new SUBLEVEL column to the DB2 TAPEVOL table, whether you intend to exploit the new function or not.

Migration action:

- If your installation's DB2 TAPEVOL table does *not* contain a SUBLEVEL column, edit the sample job CBRSMR19 to make the changes specified in the comments section, and then run the job. The job adds the new TSL SUBLEVEL column to the DB2 TAPEVOL table and primes it with default values.
- If you are migrating from z/OS V1R7, and if your installation's DB2 object directory tables do *not* contain an ODLOBFL column, edit the sample job CBRSMR18 to make the changes specified in the comments section, and then run the job. The job adds the ODLOBFL column to the object directory tables, and drops and re-creates the object directory table views.
- Run OAM DB2 BIND and GRANT jobs. To determine which BIND and GRANT jobs you need to run, see *z/OS DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

DFSMSdftp: Accommodate the changed condition code form an IDCAMS DIAGNOSE command against the ICF catalog (Required-IF, as of R9, and on R8 with OA19984)

Required if you depend on the condition code from the IDCAMS DIAGNOSE command.

When using the IDCAMS DIAGNOSE command against an ICF catalog, you might receive the following messages: IDC11374I THESE ADDITIONAL CATALOG REFERENCED VOLUMES WERE ENCOUNTERED: volume-serial-numbers

IDC000II FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 4

In this situation, the listed volume serial numbers are referenced by catalog entries but these volume serial numbers were not found as VVDS entries. This could indicate that entries are referencing volumes for which the catalog has

Migrating to z/OS 1.10 - Part 2

no record. Although there is no impact on processing, DIAGNOSE considers this an attention condition and issues a warning message with processing continued.

As of z/OS V1R8 with APAR OA19984, message IDC11374I has been removed from IDCAMS DIAGNOSE command processing in the above situation. The condition code returned from this message is now zero.

Migration action: Change procedures that depend on the condition code from the IDCAMS DIAGNOSE command against the ICF catalog. The situation previously reported by warning message IDC11374I is no longer detected and condition code 4 is not set for this reason.

Notes: 1. There should be a "SYS1.VVDS.Vvolser" entry for each volume referenced by VSAM entries in the ICF catalog, except where the "SYS1.VVDS.Vvolser" is created by EOV processing.

2. Message IDC11374I was an attention and might signal entries that are missing or damaged. To add an entry for the missing VVDS to the catalog, use IDCAMS DEFINE RECATALOG on the VVDS. If a VSAM data set incorrectly references the volume, remove the reference from the data set entry with IDCAMS ALTER REMOVEVOLUMES.

DFSMSdss: Use the new version of the DEFRAG command (Recommended, as of R10)

Recommended for improved efficiency, improved debugging, and compatibility with extended address volumes.

Before z/OS V1R10, the DFSMSdss DEFRAG command contained a CONSOLIDATE keyword. In z/OS V1R10, a new version of the DEFRAG command is added that is compatible with extended address volumes. This new version of the command drops the CONSOLIDATE keyword and adds other keywords to improve efficiency and debugging. To replace the CONSOLIDATE keyword, a new DFSMSdss CONSOLIDATE command is added for use in conjunction with the new version of the DEFRAG command. This new CONSOLIDATE command allows the consolidate function to be used without any additional DEFRAG processing of a volume.

The pre-z/OS V1R10 version of the DEFRAG command with its CONSOLIDATE keyword is still available if you specify it with the new VERSION1 keyword. Note that the previous version is not compatible with extended address volumes.

Migration action:

- To use the z/OS V1R10 version of the DEFRAG command, specify the DEFRAG command without specifying the CONSOLIDATE keyword, and use the new, separate CONSOLIDATE command. Using both the CONSOLIDATE and DEFRAG commands consecutively in the same job will give the same information and do essentially the same thing as the old DEFRAG command with the CONSOLIDATE keyword. But in addition, the new commands are compatible with extended address volumes, you can use the new MAXTIME and MMOVPCT keywords to limit the amount of time a job runs, and the new CONSOLIDATE command reduces the number of extents of a data set as much as possible rather than only when the extents can be reduced to one.

To use the z/OS V1R10 CONSOLIDATE command, you must have access to the STGADMIN.ADR.CONOLID profile in the FACILITY class. (Using the z/OS V1R10 DEFRAG command continues to require access to the existing STGADMIN.ADR.DEFRAG profile.)

- To use the pre-z/OS V1R10 version of the DEFRAG command with its CONSOLIDATE parameter intact, specify the new VERSION1 keyword. Note that this previous version is not compatible with extended address volumes, and data sets will only have extent reduction when they can be reduced to one extent.

Existing (created on z/OS V1R9 or earlier) JCL that specifies the DEFRAG command with the CONSOLIDATE keyword on z/OS V1R10 (without VERSION1) will cause the CONSOLIDATE command to be executed first, followed by the DEFRAG command using the new version of DEFRAG.

DFSMSHsm: Define facility class names to protect (H)BACKDS NEWNAME commands (Req-IF as of R10)

Required if you use the BACKDS NEWNAME and HBACKDS NEWNAME commands.

Starting in z/OS V1R10, new DFSMSHsm facility class names protect the BACKDS NEWNAME and HBACKDS NEWNAME commands. DFSMSHsm also checks for RACF authorization to the data set name specified as the NEWNAME parameter. The new facility class names must be defined to protect from unintended use of this support by unauthorized users. This support is only available on z/OS V1R10. Earlier-level systems will be able to recover backup versions created with the NEWNAME keyword.

The NEWNAME keyword is added to the DFSMSHsm (H)BACKDS command. When users specify the (H)BACKDS NEWNAME command, DFSMSHsm creates a backup version of the data set specified on the command and makes the backup version look like it is a backup of the data set specified with the NEWNAME keyword.

Migration action: Define new facility class name STGADMIN.ARC.BACKDS.NEWNAME to authorize any BACKDS NEWNAME command and STGADMIN.ARC.ENDUSER.HBACKDS.NEWNAME to authorize any HBACKDS NEWNAME command.

DFSMSHsm: Use the generic rather than the esoteric unit name for duplex generated tape copies **(Required-IF as of R10)**

Required if you need to use the generic unit name.

Before z/OS V1R10, when a duplex failure generated a tape copy, DFSMSHsm passed the generic equivalent of a SETSYS unit name (such as 3590-1) to the tape copy. Beginning with z/OS V1R10, DFSMSHsm instead passes a SETSYS-specified unit name (such as SETSYS MIGUNITNAME (*esoteric-unitname*)).

This change ensures that the same output devices and unit name that are targeted for the failed duplex copy are targeted for the tape copy. This change affects the DFSMSHsm backup, migration, and recycle functions. The ABARS function is not changed.

Migration action: To change the behavior back to always passing the generic unit for the tape copy, use the DFSMSHsm PATCH command to set the MCVTF_GENERIC_TCN_UNIT flag on. For example:

```
PATCH .MCVT.+196 BITS(...1...)
```

To return to the default behavior of using the SETSYS specified unit name, use the following command:

```
PATCH .MCVT.+196 BITS(...0...)
```



DFSMSrmm: Replace DFSMSrmm CIM providers and CIM classes **(Required-IF as of R10)**

Required if migrating from z/OS V1R9: Yes, if you have chosen to use the z/OS V1R10 CIM provider (not the backward-compatible CIM provider) as described in migration action “DFSMSrmm: Use the backward-compatible CIM provider”.

Required If migrating from z/OS V1R8: Yes, if you have chosen to use the z/OS V1R10 CIM provider (not the backward-compatible CIM provider) as described in migration action “DFSMSrmm: Use the backward-compatible CIM provider”, or if you previously set up a CIM server and agent for DFSMSrmm and registered the DFSMSrmm CIM classes.

In z/OS V1R10, the keys used for the DFSMSrmm CIM classes were changed. In z/OS V1R9, DFSMSrmm CIM providers and CIM classes were changed. You must unregister the previous CIM providers and CIM classes, and register the z/OS V1R10 CIM providers and CIM classes.

Migration action:

1. If you are planning to use the z/OS V1R10 CIM provider rather than the backward-compatible CIM provider, update your code to handle the new key formats in z/OS V1R10. For the new formats, see migration action “DFSMSrmm: Use the backward-compatible CIM provider”.
2. Using the rmmutil.sh tool, unregister all the pre-z/OS V1R10 CIM providers and unload all the pre-z/OS V1R10 CIM classes.
3. Using the same rmmutil.sh tool, register the complete set of z/OS V1R10 CIM providers and load the z/OS V1R10 CIM classes.

z/OS 1.10 Migration Actions

JES2 Migration Actions for z/OS R10 from z/OS R8

Migration Actions Pre-First IPL:

- Accommodate JES2 dynamic exits function (Required-IF, as of OA23561/OA21346)
 - Review (and change if necessary) any JES2 exit routines that use the following data areas and macros:
 - \$XRT and \$XIT data areas: \$XRT structure is altered. Some fields moved from \$XIT to \$XRT. Reassemblies will, in general, fail for field reference changes.
 - \$LMT data area: two new states may have to be considered.
 - \$MODLOAD and \$MODCHECK macros: message processing has changed.
 - Also, review your installation load modules (those loaded with the LOADMOD initialization statement) to determine if they can support dynamic loading and deletion. May want to use DYNAMIC=NO option on the \$MODULE macro to prevent unwanted processing.
- Update automation for command responses to \$D NODE, \$D EXIT, and \$D LOADmod (Required-IF, as of R10, and as of OA21346)
- Update exit routines that examine the \$JCT, and exits 8, 31, 42, and 45 (Required-IF, as of R9)

28
© 2008 IBM Corporation

JES2 Actions Between z/OS V1 R8 and z/OS V1 R10

These migration actions were taken from *z/OS Migration*. Some descriptions and actions have been shortened for inclusion in this presentation. For the complete descriptions and actions, refer to *z/OS Migration*.

JES2 Migration Actions You Can Do Now

<none>

JES2 Migration Actions Pre-First IPL



Accommodate JES2 dynamic exits function (Required-IF, as of R10)

Required if you use JES2 exit routines or load installation load modules using the JES2 LOADMOD (or LOAD) initialization statement.

The dynamic exits function of JES2, integrated in z/OS V1R10 and rolled back to prior releases by APARs, makes the following possible:

- Installation load modules (exit modules) can be dynamically loaded, refreshed, or deleted. Note that this function does not apply to modules that are part of the base JES2 product, meaning that you cannot use this function to apply IBM service.
- The list of routines associated with an exit can be dynamically altered.

Because of this new function, you should review and change (if necessary) any JES2 exit routines that use the following data areas and macros:

- The \$XRT and \$XIT data areas. The structure of the \$XRT has been significantly altered. In addition, some fields in the \$XIT have been moved to the \$XRT. Reassembly of your installation code will, in general, fail if it references any fields in the \$XRT or \$XIT that have changed.
- The \$LMT data area. Before z/OS V1R10, a \$LMT can be active (LMT1INVD bit off) or invalid/free (LMT1INVD bit on). Beginning with z/OS V1R10, a \$LMT can be in two new states: it can represent a module that has been logically deleted (LMT2DELT bit on) or a module whose storage has been freed (LMT3FREE bit on). Code that examines \$LMTs might need to take these two new states into consideration.

- The \$MODLOAD and \$MODCHK macros. Message processing for these macros has changed. Beginning with z/OS V1R10, if MESSAGE=NO is coded or defaulted on the macro invocation, the message area returned in register 1 must be freed by the caller (using \$FREMAIN with SP=0, KEY=1, and LV=128). If the caller does not require the message to be returned, a new MESSAGE=SUPPRESS option can be used. In addition, not all exit routines can tolerate the fact that installation load modules (ones loaded using the JES2 LOADMOD initialization statement) can now be deleted and refreshed (a new copy loaded and the old copy deleted) using an operator command. You can use the new DYNAMIC=NO option on the \$MODULE macro to prevent unwanted processing.

Migration action:

1. **Update exit routines that refer to the JES2 \$XRT and \$XIT data areas, if necessary.** The \$XIT is a 256-element array that represents all possible JES2 exits. \$XRTs are pointed to by the \$XIT. Each \$XRT is an array in which each entry represents a routine associated with a JES2 exit. Before z/OS V1R10, the \$XRT was a simple array of elements, each one representing a routine. With the dynamic exits function in z/OS V1R10, the \$XRT starts with a header that describes the list, followed by an array of XRTes in which each entry represents a routine.

The change was needed to ensure predictability when routines are invoked by the JES2 \$EXIT facility. When the \$EXIT facility is invoked, it selects the appropriate \$XRT and increments the use count in that \$XRT. The list of routines in that instance of the \$XRT are used by the \$EXIT facility. Operator changes to the list of routines while the \$EXIT facility is in control do not affect the currently active exit environments. When the list of routines for an exit is altered, a new \$XRT is built and, if the existing one is in use, the new one is chained in ahead of the existing \$XRT and the old \$XRT is marked as logically deleted. When the old \$XRT is no longer in use, it is deleted.

Note that the first \$XRT associated with an exit (pointed to by XITXRTAD) can be logically deleted (XRTFDEL bit on) if there are no longer any routines associated with the exit. If there is a chain of \$XRTs, the second and beyond are always logically deleted. (Only the first \$XRT pointed to by XITXRTAD can be active.) One consequence of this change is that some fields that were in the \$XIT had to be moved to the \$XRT header. In particular, the fields XITBSPL (SPLEVEL check bypassed), XIT#RTNS (number of routines associated with this exit), and XITUSCNT (exit use count) have been moved to the \$XRT.

2. **Update exit routines that refer to the JES2 \$LMT data area, if necessary.** The \$LMT is used to track modules loaded by the JES2 \$MODLOAD service, including modules loaded by the LOADMOD initialization statement and the new \$ADD LOADMOD command. Code that searches \$LMTs needs to consider the state of the \$LMT before examining fields (such as the module address) in the \$LMT. Before z/OS V1R10, the only state needing examination was invalid/free (bit LMT1INVD on). However, the dynamic exits function in z/OS V1R10 introduces two new states that need to be considered: logically deleted \$LMTs and force freed \$LMTs.

Logically deleted \$LMTs (bit LMT2DELTD on) represent deleted modules that are not being used for new requests (they are logically deleted) but are in use by existing processing. These \$LMTs should not be examined when looking for the current version of a module but can be used if trying to resolve what module a particular address is located in.

Force freed \$LMTs (bit LMT3FREE on) represent load modules where the actual module is freed before JES2 is done using it. This can only occur if dynamic LPA services is used to delete the module from LPA (something IBM does not recommend doing). Code cannot access the actual load module because the storage is no longer valid. Force freed \$LMTs will eventually be logically deleted (LMT2DELTD set on), although the LMT3FREE bit will always be on.

If an exit routine examines \$LMTs to find a particular routine address by name or to find the load module name associated with an address, there are JES2 services to do this that isolate the code from the internal JES2 data structures. In the JES2 environment, the LOCLMOD service obtains the module name associated with an address (including searching for MVS module names). Outside the JES2 environment, FINDLMOD performs the same function. To locate the address of a routine within a JES2 load module, you can use the LOCENTRY service. However, this is only available within the JES2 environment.

3. **Update exit routines that use the \$MODLOAD and \$MODCHK macros, if necessary.** Message processing for the \$MODLOAD and \$MODCHK macros has changed. Before z/OS V1R10, if MESSAGE=NO was coded or defaulted on the macro invocation, an error message could be returned to the caller in a data area pointed to by register 1. The caller could choose to use or ignore the returned message. The message text was in a work area chained off the HCT and did not need to be freed. However, with the introduction of the dynamic exits function, the data area is obtained by the service and must be freed by the caller using \$FREMAIN with SP=0, KEY=1, and LV=128. If the caller does not need the message text, change the macro invocation to specify MESSAGE=YES (message is issued via WTO) or MESSAGE=SUPPRESS (no message is issued or returned).

4. **Enable load modules to support dynamic processing, if necessary.** With the dynamic exits function in z/OS V1R10, JES2 supports loading, refreshing, and deleting installation load modules using an operator command. JES2 will process all dynamic tables (TYPE=DYNAMIC) that exist in the module just as they were processed before z/OS V1R10 by initialization. Exit routine addresses are cleared when a module is deleted and replaced when a module is refreshed. (\$ADD LOAD will not affect any exit routine address.) However, even with this support, not all load modules will function properly if loaded, deleted, or refreshed after initialization. You must review all your installation load modules (ones loaded with the LOADMOD initialization statements) to determine whether they can be dynamically loaded or deleted and, if they cannot be dynamically loaded or dynamically deleted, decide what needs to be done.

Examples of code that cannot be dynamically loaded or deleted without changes are:

- Modules that have assembled tables that are pointed to by JES2 data areas, for example, a \$UCT data area pointed to by the \$UCT field in the \$HCT. Other tables can be pointed to by other user pointers in JES2 data areas, or the tables can be pointed to by a named token (created using the \$TOKENSR macro).
- Modules that front-end JES2 services.
- Modules that implement their own \$PCE or \$DTE.
- Modules with \$BERTTABS that were not loaded at initialization.
- Modules that define installation \$WSTABs.

Many of these modules will have an exit 24 (post initialization) associated with them. The module was loaded as part of normal initialization processing and then hooked to JES2 data structures using exit 24. Alternatively, they might have code that tests a user field for zero and, if it is zero, they do one-time initialization processing. However, if the initialization code involves obtaining storage and saving that storage address in a JES2 data area, it might not have a problem.

These exit routines might not function properly if they are processed with an operator command. In some cases there will be residual pointers in JES2 data areas that do not get updated. In other cases (such as user defined PCEs or DTEs) there might be no way to stop code that is running in the module. Modules with \$BERTTABS need special considerations. At JES2 initialization, the \$BERTTABS are processed into a table that is used while JES2 is running. Updating this table while JES2 is active could result in unexpected side effects. As a result, JES2 does not process new \$BERTTABS when a module is dynamically loaded. Only \$BERTTABS in modules that were loaded at JES2 initialization (including hot starts) are used. There is no restriction on refreshing modules with \$BERTTABS except that new \$BERTTABS will not be processed. Be careful with any module that contains \$WSTABs. For performance reasons, the address of the \$WSTAB is stored in various JES2 data areas. These addresses are only updated when the WS= parameter associated with a device is updated. New \$WSTAB entries are not processed until the WS= values on any device that use the new work selection criteria are updated (in general by operator command).

There are two ways to address modules that cannot be dynamically updated:

- Prohibit all dynamic processes for these modules. This can be done by setting the DYNAMIC=NO parameter on the \$MODULE macro in the modules.
- Use the new \$\$\$\$LOAD and \$\$\$\$DEL service to perform the initialization processing normally done by exit 24. This generally eliminates the need for exit 24 even for the case when the module is loaded by initialization processing. The \$\$\$\$LOAD and \$\$\$\$DEL routines can also be used to dynamically create and detach PCEs and DTEs. Furthermore, for the case where you want new code loaded but you also want the table that exists in the old module to remain, a function in \$\$\$\$DEL prevents JES2 from ever deleting the load module.

Update automation that handles \$D NODE, \$D EXIT, and \$D LOADmod command responses **(Required-IF, as of OA21346, and as of R10)**

Required if you have automation that examines these command responses.

For \$D NODE: Before z/OS V1R10, the command response to \$D NODE would return all parameters. Beginning with z/OS V1R10, the command response is changed to display only an abbreviated list of parameters. For the complete list, you can issue the \$D NODE, LONG command. For more details about the parameters in the abbreviated list, see the description of the LONG parameter on the \$D NODE command.

For \$D EXIT: Before z/OS V1R10 (and APAR OA21346 on z/OS V1R9 and z/OS V1R8), the command response to \$D EXIT would return all parameters. Beginning with z/OS V1R10, the command response is changed to display only an abbreviated list of parameters. For the complete list, you can issue the \$D EXIT, LONG command. For more details about the parameters in the abbreviated list of \$D EXIT command parameters, see the description of the LONG parameter on the \$D EXIT command.

For \$D LOADmod: Before z/OS V1R10 (and APAR OA21346 on z/OS V1R9 and z/OS V1R8), the command response to \$D LOADmod would return all parameters. Beginning with z/OS V1R10, the command response is changed to display only an abbreviated list of parameters. For the complete list, you can issue the \$D LOADmod, LONG command. For more details about the parameters in the abbreviated list of \$D LOADmod command parameters, see the description of the LONG parameter on the \$D LOADmod command.

Migration action: Review and update automation that examines these commands responses.

Update JES2 exit routines that examine the \$JCT (Required-IF, as of R9)

Required if you have any exit routines that examine the \$JCT eye-catcher field.

Before z/OS V1R9 JES2, the field name for the \$JCT eye-catcher was JCTID. Starting with z/OS V1R9 JES2, the field name is JCTIDENT. If you have any exit routines that examine the \$JCT eye-catcher, change them to use the new field name.



Update JES2 exit 8 routines (Required-IF, as of R9)

Required if you use exit 8 and that exit routine examines the CBMTTR field.

Before z/OS V1R9 JES2, exit 8 used the MTTR parameter to specify a track address. Starting with z/OS V1R9 JES2, exit 8 uses the MQTR parameter instead. If you have exit routines that examine field CBMTTR, change them to examine field CBMQTR instead of field CBMTTR.



Update JES2 exit 31 routines (Required-IF, as of R9)

Required if you use exit 31.

Before z/OS V1R9 JES2, there was no mapping macro for the exit 31 parameter list. There was just an area pointed to by a register, with the contents at all applicable offsets described in the documentation. Starting with z/OS V1R9 JES2, exit 31 uses the \$XPL data area instead. If you use exit 31, you will have to change your exit routine.



Update JES2 exit 42 routines (Required-IF, as of R9)

Required if you use exit 42.

To accommodate unauthorized SSI callers, the following changes have been made to the notify user exit (exit 42) to provide more input/output fields in its \$XPL data area:

- The pointer to the SSNU control block has been renamed to alert programmers about the changes.
- Additional fields are provided in the \$XPL to allow an exit to obtain functionality similar to that provided by the SSNU without having to reference the SSNU when modifying the notify user message. If an exit still must reference data in the SSNU, the SSI caller's key is provided.
- Condition indicator X042NOST in condition byte X042COND has been deleted. When storage cannot be obtained, the SSI no longer calls exit 42. Instead, it returns with SSNUERR set in the SSOBRETN field.



Update JES2 exit 45 routines (Required-IF, as of R9)


Required if you use exit 45 and it references any IAZSSSF fields.

To accommodate unauthorized SSI callers, input and output to the SJF services user exit (exit 45) has been updated as follows:

- The SSI caller's storage key is in the X045CKEY field so that the exit can correctly reference SSSF fields if necessary.
- Additional \$XPL fields have been added, which correspond to SSSF fields. The benefit of using these new fields instead of the SSSF fields is that the new fields are in the JES key, so no keyed references are necessary.
- The address of the SSSF control block has been renamed to X045SSFA to alert programmers about the changes made to the \$XPL.
- You can influence the type of security authorization checking that is to be done by setting flag byte X045FLG1 as desired. Note, however, that if the exit is entered as a result of an unauthorized SSI 70 call, this value is not honored and X045JSSP is the default.
- Condition indicator X045NOST in condition byte X045COND has been deleted. When storage cannot be obtained, the SSI no longer calls exit 45. Instead, it returns with SSSFNOST set in the SSOBRETN field.

JES2 Migration Actions Post-First IPL <none>

z/OS 1.10 Migration Actions



Language Environment Migration Actions for z/OS R10 from z/OS R8

Migration Actions Pre-First IPL:

- **Determine the impact of added and changed runtime options (Required-IF, as of R10)**
 - As of R10, and rolled back: HEAPPOOLS option was added to the CELQDOPT options group (was already valid in CEEDOPT and CEECOPT) As of R9: CEEDUMP option was added, to control the processing of the dump report CEEDUMP.
 - If you use CEEPRMxx: **no migration action** since options added don't change default behavior!
 - If you use assembler language modules: update and reinstall your usermods.
- **Use CEEPRMxx's CEEROPT instead of making copies of Language Environment initialization routines (Required-IF, as of R10)**
 - Many installations have copied LE init routines (making updates to the copies for specific runtime options), and STEPLIB'd to the copies for different applications. **This method will no longer work as of R10!**
 - Now in CEEPRMxx, CEEROPT(ALL) indicates that CEEROPT should be loaded for all application execution environments, and that the options found in that module should be used. **This method does not required any updates to applications!**
 1. Turn on CEEROPT(ALL) in CEEPRMxx, which is not the default.
 2. Create a separate CEEROPT module with your customization similar to how you made the changes before, and put in a data set. Use sample CEEWROPT.
 3. STEPLIB to that CEEROPT data set you customized for your application.
 4. Ensure you remove, and no longer have the LE init modules in the STEPLIB concat.

29
© 2008 IBM Corporation

Language Environment Migration Actions Between z/OS V1 R8 and z/OS V1 R10

Some migration actions from specific elements were selected from *z/OS Migration*. Many descriptions and actions have been shortened for inclusion in this presentation. For the complete descriptions and actions, refer to *z/OS Migration*.

Language Environment Migration Actions You Can Do Now

<none>

Language Environment Migration Actions Pre-First IPL

Update the CICS CSD based on the newest CEECCSD sample (Required)

Each release, Language Environment adds or deletes load modules in the CICS system definition (CSD) file. Thus, you should update the file each release using the program definitions found in member CEECCSD and, if using CICS Transaction Server (TS) for z/OS V3 (5655-M15), in member CEECCSDX..

Migration action: Update the CSD file using the program definitions in member CEECCSD (and member CEECCSDX if using CICS TS V3) found in the *hlq.SCEESAMP* data set.

Note:

1. The group containing the Language Environment runtime routines must be in the group list used during CICS startup.
2. The size of the CEECCSD decreased dramatically in z/OS V1R9. The decrease was intentional; it occurred because iconv() started using the Unicode converters instead of the converters supplied with Language Environment.

Review Language Environment load modules in LPA (Required-IF)

Required if you need to make modules accessible through the link pack area (LPA).

Each release you must update the Language Environment load modules that you make accessible through the link pack area (LPA). In addition, each release you should review your list of load modules in the LPA to determine if it's still suitable.

Migration action: Review Language Environment load modules in the LPA.

- To move load modules into the LPA, use sample members are provided in the CEE.SCEESAMP data set.
- To see which modules are eligible for the LPA, refer to *z/OS Language Environment Customization*. The modules listed there can be put in the LPA or extended LPA (ELPA) depending on their RMODE value:
- If you are considering placing the modules listed in *z/OS Language Environment Customization* in the LPA or the ELPA, then IBM highly recommends that you place the SCEELPA data set in the LPA list (LPALST xx). SCEELPA contains Language Environment load modules that are reentrant, that reside above the 16 MB line, and that are heavily used by z/OS.
- In *z/OS Language Environment Customization* you will also see tables of modules eligible for the LPA and the ELPA above and beyond what is found in the SCEELPA data set. You will need to use the dynamic LPA or MLPA approach to move these modules into the LPA or ELPA. You do not need to include recommended modules if they contain functions your installation does not use. Language Environment modules not listed in these tables can be moved into the LPA or ELPA at your discretion.



Determine the impact of added and changed run-time options (Required-IF, as of R10)

Required if you are using assembler modules to specify your installation-wide run-time options (CEEDOPT, CEECOPT, or CELQDOPT). If you are using CEEPRMxx (introduced in z/OS V1R7) and the default specification for the newly added option is acceptable to you, then no migration action is required.

Periodically, Language Environment introduces new run-time options, adds new suboptions to existing run-time options, and changes the defaults of run-time options.

In z/OS V1R10:

- One runtime option, HEAPPOOLS, was added to the CELQDOPT options group. (HEAPPOOLS was already valid in both CEEDOPT and CEECOPT.) This support is integrated in z/OS V1R10; it was initially made available on z/OS V1R9 by APAR PK49427 and on z/OS V1R8 and V1R7 by APAR PK57579.
- No suboptions were added and no default changes were made to existing runtime options.

In z/OS V1R9:

- One run-time option, CEEDUMP, was added. CEEDUMP is used to specify options to control the processing of the Language Environment dump report CEEDUMP.
- No suboptions were added and no default changes were made to existing run-time options.

Because of these changes, you might have to perform migration actions.

Migration action:

- If you are using CEEPRMxx to specify your installation-wide runtime options:
 - The new z/OS V1R10 runtime option HEAPPOOLS that was added to CELQDOPT does not change the default behavior. You have no migration action.
 - The new z/OS V1R9 CEEDUMP runtime option does not change the default behavior. You have no migration action.

Tip: Use IBM Health Checker for z/OS to check that default Language Environment runtime options are set within a CEEPRMxx parmlib member. The check is named CEE_USING_LE_PARMLIB.

- If you are using the assembler modules to specify your installation-wide runtime options, compare your existing source for the installation-wide runtime options CSECT, CEEDOPT (non-CICS environment), CEECOPT (CICS environment), or CELQDOPT (AMODE 64) with the new samples in *hlq.SCEESAMP* to determine whether you need to change the defaults. Then, update (and reinstall) your usermods with the following new options:
 - The new z/OS V1R10 runtime option HEAPPOOLS that was added to CELQDOPT. But note that this function, which is integrated in z/OS V1R10, was rolled back to prior releases (APAR PK49427 on z/OS V1R9 and APAR PK57579 on z/OS V1R8 and V1R7), so you might have already updated this usermod.
 - The new z/OS V1R9 CEEDUMP runtime option.



Use CEEPRMxx's CEEROPT instead of making copies of Language Environment initialization modules (Required-IF, as of R10)

*Required if you have copied any Language Environment initialization module (CEEBINIT, CEEBINSS, CEEPIPI, or CEEBPIC) for the purpose of manually linking in a separate installation-wide runtime option CSECT CEEDOPT. **Note:** If you have used **only** SMP/E to maintain a customized CEEDOPT CSECT (by using a sample usermod CEEWD01), this migration action does not apply to you. Using the CEEWD01 usermod to specify your runtime options in CEEDOPT will still result in Language*

Migrating to z/OS 1.10 - Part 2

Environment correctly using your specified runtime options. This migration action is applicable only if you have linked CEEDOPT outside of SMP/E into separate libraries that were not SMP/E-controlled (and were STEPLIBed to by the applications).

Many installations have made copies of Language Environment's initialization modules CEEBINIT, CEEBINSS, CEEPIPI, and CEEBPICI (updating each copy with specific runtime options specified in the CEEDOPT CSECT), and then used STEPLIB to access those modules for different applications. This method allows applications to individually have their own runtime options while not requiring any updates to the applications themselves. However, this method has the disadvantages of being difficult to manage when service updates are made in Language Environment, having many copies of Language Environment modules to keep track of, and creating synchronization problems between the initialization modules and other Language Environment modules, which are hard to debug.

This method of customizing Language Environment's initialization modules no longer works as of z/OS V1R10. As of z/OS V1R10, the CEEDOPT CSECT is no longer part of the Language Environment's initialization modules CEEBINIT, CEEBINSS, CEEPIPI, and CEEBPICI. Therefore, any customization for runtime options added to these initialization modules will not be used by Language Environment for the application's runtime environment. To easily allow applications to continue to use customized runtime options, and to eliminate the need to make separate copies with updates to the initialization modules, support was added to the CEEPRMxx parmlib member. The CEEPRMxx CEEROPT statement (for non-AMODE 64 applications and CELQROPT for AMODE 64 applications) controls this support and CEEROPT(ALL) indicates to Language Environment that CEEROPT should be loaded for all application execution environments, and that the options found in that module should be used. Using CEEPRMxx's CEEROPT statement does not require any updates to the applications themselves. Because of these changes, you might have to perform migration actions.

Migration action:


1. CEEROPT processing can only be specified in your CEEPRMxx parmlib member (as CEEROPT is not an option in the CEEDOPT assembler language module). Turn on CEEROPT processing for all non-AMODE 64 environments by setting the keyword CEEROPT(ALL) in your active CEEPRMxx member. (CEEROPT(COMPAT) is the default, which indicates CEEROPT is to be loaded and used only for CICS and LRR environments. This is the behavior prior to z/OS V1R10.)
2. Create a separate CEEROPT module with runtime option overrides similar to those you are currently manually linking into a copy of one of the Language Environment initialization modules. You can use sample CEE.SCEESAMP(CEEWROPT) for the assembly and link-edit. Place each CEEROPT module in a separate data set.
Note: You only need to set those runtime options that are different from the ones set in your installation defaults or system-wide defaults.
3. Place this CEEROPT data set into the STEPLIB concatenation for the application.
4. Review the other modules you have in the STEPLIB concatenation. If you had any Language Environment initialization modules in the concatenation for the sole purpose of customizing the CEEDOPT CSECT, you must remove those modules from the concatenation.

Language Environment Migration Actions Post-First IPL



<none>

z/OS 1.10 Migration Actions

z/OS UNIX Migration Actions for z/OS R10 from z/OS R8



Migration Actions You Can Do Now:

-  **Migrate from HFS to zFS file systems (Recommended)**
 - HFS has been stabilized, and zFS is strategic.
 - Convert your file systems from HFS to zFS.
-  **Use ServerPac (during install, for the shipped file systems), BPXWH2Z (for after install, and other file systems you have), and look at R10's FOMVS,NEWROOT= support for the sysplex root.**


Migration Actions Pre-First IPL:

- Update z/OS UNIX configuration files (Required-IF, as of R10)**
 - /etc/rc was changed to clear man page cache for R10. /etc/magic was changed for R9. /etc/ohelp.ENU changes every release.
- Increase dump space to accommodate dumps of the zFS address space (Required-IF, as of R9)**
 - Before R10: system dumps didn't include dumps from the zFS address space.
 - As of R9: zFS data is captured, as long as the address space is active and the dump cause is related to a file system problem.
 - Can be larger, may need to increase your dump data sets.

30 © 2008 IBM Corporation

z/OS 1.10 Migration Actions

z/OS UNIX Migration Actions for z/OS R10 from z/OS R8



Migration Actions Post-First IPL:

- Use the Shell and Utilities version of the submit command (Recommended, as of R10)**
 - Before R10: submit command was a Tools and Toys download.
 - As of R10: it is part of the Shell and Utilities support in z/OS.
 - Start using the R10 submit command, remove the "old" submit.
- Update ln invocations to accommodate changes made to the ln shell command (Required-IF, as of R10)**
 - Before R10: on the ln command, the -e option with -s, -r, or -R had invalid results.
 - As of R10: -e is mutually exclusive with -s, -r, or -R. If you try to specify -e with those options or with an existing target directory, you will receive an error.
 - Remove any -e option from ln invocations, so it is not used with -s, -r, or -R.

31 © 2008 IBM Corporation

z/OS UNIX Migration Actions Between z/OS V1 R8 and z/OS V1 R10

Some migration actions from specific elements were selected from *z/OS Migration*. Many descriptions and actions have been shortened for inclusion in this presentation. For the complete descriptions and actions, refer to *z/OS Migration*.

z/OS UNIX System Services Migration Actions You Can Do Now



Migrate from HFS file systems to zFS file systems (Recommended)

Not required, recommended because the action is planned to become a requirement in a future release. zFS is the strategic file system for z/OS UNIX and continues to be enhanced to provide superior performance, reliability, and data integrity. IBM has not announced removal of support for the HFS file system.

Before z/OS V1R7, the HFS file system was the primary hierarchical file system. As of z/OS V1R7, you can use any combination of HFS and zFS file systems. Because zFS has higher performance characteristics than HFS and is the strategic file system, you should migrate your HFS file systems to zFS.

The HFS and zFS file system types in mount statements and command operands are now generic file system types that can mean either HFS or zFS. Based on the data set type, the system will determine which is appropriate. But note that you must still specify a type (HFS or zFS and it cannot be defaulted), and if the type you specify is not correct for the file system being mounted, any associated parameter string setting in the mount statement or command is ignored, even though the system sets the type correctly and processes the mount.

Migration action:

1. Before beginning the migration, do the following:
 - Establish backout procedures.
 - Decide on naming conventions.
 - Decide on unavailability.
 - Understand any cloning or deployment changes required by zFS systems being linear data sets. Considerations would include any copy utility invocations, BPXPRMxx specifications for symbolics, and placement of zFS file systems on system volumes.
2. Perform the conversion from an HFS to zFS file system.

Tip: Use the BPXWH2Z tool to perform the conversion. It is an ISPF-based tool that migrates HFS file systems to zFS file systems. Using its panel interface, you can alter the space allocation, placement, SMS classes, and data set names. A HELP panel is provided. With this tool, you can:

 - Migrate HFS file systems (both mounted and unmounted) to zFS file systems. If the HFS being migrated is mounted, the tool automatically unmounts it and then mounts the new zFS file system on its current mount point.
 - Define zFS aggregates by default to be approximately the same size as the HFS. The new allocation size can also be increased or decreased.
 - Have the migration run in TSO foreground or UNIX background.

You can use the JCL sample ISPBTCH in SYS1.SAMPLIB to invoke BPXWH2Z as an ISPF batch job. Before you run the job, be sure to read the Notes section. When you run BPXWH2Z on your z/OS V1R10 system, it uses the z/OS V1R10 level of the **pax** command. This level was enhanced in z/OS V1R7 for sparse file support and other characteristics that are of concern when migrating from an HFS to zFS file system. You can manually migrate from an HFS to zFS file system without using the tool. However, you would need to allocate and format the target zFS file systems.

Requirement: The BPXWH2Z tool requires the zFS address space to be operational. Therefore, before attempting to migrate existing HFS to zFS file systems using BPXWH2Z, make sure that the zFS address space has been successfully configured and initialized.

Tip: Consider migrating the sysplex root file system to zFS after you IPL z/OS V1R10. You can now dynamically migrate the HFS sysplex root in a shared file system configuration to zFS while the root is in use, without disrupting workloads. Before z/OS V1R10, you could not migrate without disrupting active workloads. Although the shared file system configuration is required, the sysplex can be a single system. For the steps involved in migrating the sysplex root file system from HFS to zFS, see *z/OS Migration*.

3. Change policies and scripts, and so forth, to reflect the change from the HFS file system to zFS file system.

Tip: Use the RMF Monitor III option to report on zFS activity.

Use the new default of 10 for the automount delay keyword (Recommended, as of R9)

Recommended to prevent the system from hanging as it continually tries to unmount file systems and fails.
 Before z/OS V1R9, the default for the delay keyword for the **automount** shell command was 0. In z/OS V1R9, the default has been changed to 10. This change brings the default in line with the already-established best practice. A delay value lower than 10 minutes can cause the system to hang as it continually tries to unmount file systems and fails.

Migration action: If you are using the default of 0 on your pre-z/OS V1R9 system, determine whether the default of 10 is acceptable on your z/OS V1R9 system. If it is, you do not need to update the delay field in the MapName file. However, if the default of 10 is not acceptable, you can override the value by changing the 10 to 0.

Tip: If your system has IBM Health Checker for z/OS set up, the USS_AUTOMOUNT_DELAY check looks at the delay configuration values in a sysplex to ensure they are at least 10 minutes.

Update BPXPRMxx if more than 50 MKDIR keywords are specified (Required-IF, as of R9)

Required if more than 50 MKDIR keywords are specified on a ROOT or MOUNT statement in BPXPRMxx.
 Before z/OS V1R9, you could specify more than 50 MKDIR keywords on each ROOT or MOUNT statement in the BPXPRMxx member of SYS1.PARMLIB. In z/OS V1R9, the maximum number of MKDIR keywords allowed for each ROOT or MOUNT statement is 50. If you specify more than 50, syntax errors occur and initialization stops when parsing completes. The syntax error message that is issued is BPXI075I ERROR IN PARMLIB MEMBER=member ON LINE xxxxx. TOO MANY MKDIR STATEMENTS.

Migration action: Make sure that no more than 50 MKDIR keywords are specified on ROOT and MOUNT statements in BPXPRMxx

z/OS UNIX System Services Migration Actions Pre-First IPL



Update z/OS UNIX configuration files changed by IBM (Required)

Some utilities provided by z/OS UNIX require the use of certain configuration files. You are responsible for providing these files if you expect to use the utilities. IBM provides default configuration files as samples in the /samples directory. Before the first use of any of these utilities, you must copy these IBM-provided samples to the /etc directory (in most cases). You can further customize these files to include installation-dependent information. An example is setting up the /etc/rc file by copying the sample file from /samples/rc to /etc/rc and then customizing it for the installation.

Whenever IBM ships a changed sample configuration file, you must copy the changed sample to its target location and, if you customized the previous version of the sample, incorporate the customization into the new version.

Migration action: For each of the files listed in the table below, copy the IBM-provided sample file to the listed target location. If you added installation-dependent customization to any of the IBM-provided files, make the same changes in the new versions of the files.

Utility	IBM-provided sample file	Target location	What changed and when
file	/samples/magic	/etc/magic	In z/OS V1R9, the system default magic file was changed to support the SUSv3 format. The V1R9 file command supports old format magic files without change. The new /samples/magic file has been changed to comply with SUSv3.
initialization	/samples/rc	/etc/rc	In z/OS V1R10, the initialization script was changed to clear out the man page cache. This eliminates the migration action "Update the date and time of book files used for man pages".
OHELP	/samples/ohelp.ENU	/etc/ohelp.ENU	Every release, bookshelf and book data set names are updated. See "Install books and a bookshelf for the OHELP command" on page 306.



Ensure that _UNIX03 is not set in system profiles (Required-IF, as of R9)

Required if you want shell commands to maintain compatibility with prior releases.

Before z/OS V1R9, the `_UNIX03` variable may have been set in system profiles. The variable was introduced in z/OS V1R8 as a means of controlling whether certain shell commands conform to Single UNIX Specification, Version 3 (SUSv3). `_UNIX03=YES` is only needed when an option or behavior conflicts with an existing z/OS implementation, and the SUSv3 behavior is desired. In z/OS V1R9, additional shell commands have behavior conflicts controlled by the `_UNIX03` variable. If `_UNIX03` is not set (or is set to "NO"), then the z/OS V1R9 commands will maintain compatibility with prior releases.

Migration action: To maintain shell command compatibility with prior releases, ensure that the `_UNIX03` variable is not set (or is set to "NO") in system-wide profiles or setup scripts, including the following:

- `/etc/profile`
- `/etc/csh.cshrc`
- `/etc/csh.login`
- `/etc/rc`

If you have set `_UNIX03=YES` in a system profile, then changing that setting may affect users. If users want behavior that conforms to SUSv3, they can set `_UNIX03=YES` in their profile files or in specific scripts or command invocations where it is needed.

z/OS UNIX System Services Migration Actions Post-First IPL



Install books and a bookshelf for the OHELP command (Required-IF)

Required if you use the OHELP command.

The TSO/E OHELP command uses the BookManager READ base element to display online reference information about shell commands, TSO/E commands, C functions, callable services, and messages issued by the shell and dbx. Before this online help facility can be used, you must set it up by installing books and a bookshelf.

Migration action: Copy the IBM-supplied sample file `/samples/ohelp.ENU` to `/etc/ohelp.ENU`. If you choose not to use the IBM-supplied default file, you can create your own file called `/etc/ohelp.language_id`, where `language_id` is the TSO/E user's primary language code.

Use the Shell and Utilities version of the submit command (Recommended, as of R10)

Recommended because the new supported version of the submit command is enhanced compared to the Tools and Toys version.

Before z/OS V1R10, the `submit` command was available from the Tools and Toys section of the z/OS UNIX Web site. As of z/OS V1R10, Shell and Utilities support of the `submit` command has been added. The supported version differs from the Tools and Toys version in a number of ways:

1. Multiple files can be used as input.
2. After the `submit` command completes, a message that indicates success or failure is sent. If the command succeeds, the resulting message contains the job IDs of the submitted jobs.
3. Formatting for submitting jobs from a partitioned data set (PDS) is different. Because the supported `submit` command is now consistent with the other subcommands, the `//` characters are required as a prefix to the PDS name. For example: `"/' full.pds.name(member)'`
4. The Tools and Toys version of `submit` had the following conventions, which have been changed:
 - `?` was used as a way to display usage but is now treated as a valid file name.
 - `-` was used as a file name but is now used to specify standard input.

Migration action:

1. If you are using the Tools and Toys version of the `submit` command, delete it or move it to a location that is not in the system's search path. If you leave it in the search path, it might be picked up instead of the supported version that is in the `/bin` directory.
2. Begin using the supported version in the `/bin` directory.

Update cksum and sum invocations (Required-IF, as of R10)

Required if any automated tasks rely on cksum or sum exiting when an automatic conversion error occurs.

Before z/OS V1R10, the `cksum` and `sum` utilities exited if they could not turn off automatic conversion on a file stream. As of z/OS V1R10, if this error is encountered, the file stream is skipped and the remaining files are processed. The message associated with this error is FSUMF182. This change should not affect most users.

Migration action: Modify any **sum** or **cksum** invocations (for example, in shell scripts) that rely on the utility exiting when automatic conversion errors occur.

Update In invocations to accommodate changes made to the In shell command (Req-IF, as of R10)

Required if In -e is used in either of the following ways: 1) in conjunction with the -s, -r, or -R option, or 2) with an existing target directory.

Before z/OS V1R10, if you specified the **-e** option with the **-s**, **-r**, or **R** option when issuing the **In** shell command, the results were not valid.

- If *source* was a directory containing symbolic links or external links and *target* was an existing directory, and you issued **In -e** with **-s**, **-r**, or **-R**, one of the following events occurred:
 - Message FSUM6245 was issued.
 - One or more incorrect symbolic links was created.
 - One or more incorrect external links was created.
- If *source* was a file or a PDS or PDSE member, **-e** was specified with **-s**, and *target* was not a directory, **In** created a symbolic link.
- If **-e** was specified with an existing target directory, **In** created an incorrect external link.

As of z/OS V1R10, the **-e** option is mutually exclusive with the **-s**, **-r**, and **-R** options. You will receive an error message if you use **-e** in conjunction with **-s**, **-r**, or **-R**, or with an existing target directory.

Migration action:

- Modify any **In** invocations (for example, in shell scripts) so that **-e** is not used in conjunction with the **-s**, **-r**, and **-R** options.

If you want to create . . .	Use . . .
A symbolic link	-s
An external link	-e
Symbolic links or external links recursively	-s with -r or -R

- Modify **In** invocations so that **-e** is not used if the target specified on the command line is an existing directory.

Update scripts to accommodate changes made to the locale shell command (Req-IF, as of R10)

Required if automated scripts evaluate strings based on error messages FSUM6701 or the default locale output.

Before z/OS V1R10, the locale shell command did not support the LC_TOD category and associated keywords. The default locale command (specified without any options or operands) did not display the LC_TOD environment variable, and using LC_TOD (or associated keywords) with or without options generated error message FSUM6701. Beginning with z/OS V1R10, the default locale command displays LC_TOD information along with other LC_* environment variables. Also, using LC_TOD or associated LC_TOD keywords is honored.

Migration action: Modify automated scripts if they evaluate strings based on error message FSUM6701 or the default locale output.

Update automation that handles message BPXF0341I (Required-IF, as of R10)

Required if any automated actions are issued for message BPXF0341I.

Before z/OS V1R10, message BPXF0341I was issued if a file system was quiesced for more than 10 minutes. As of z/OS V1R10, message BPXF0831I is issued instead. The new message includes the file system name, quiescing system, job name, and process ID (PID).

Migration action: Modify automated actions for message BPXF0341I so they now work with message BPXF0831I.

Update automation and operational procedures for file systems in a shared file system configuration (Required-IF, as of R9)

Required if you have automation or operational procedures that use MOUNT statements that specify NOAUTOMOVE, UNMOUNT, or a system list.

Before z/OS V1R9, if you mounted a file system in a sysplex-aware mount mode with the NOAUTOMOVE setting, the NOAUTOMOVE setting was changed to AUTOMOVE(YES) and message BPXF2341I was issued. Starting in z/OS V1R9, the conversion to AUTOMOVE(YES) will not occur for sysplex-aware file systems. Instead, the AUTOMOVE setting (UNMOUNT, NOAUTOMOVE, AUTOMOVE, or a system list) is now honored by z/OS V1R9 systems, independent of the sysplex-aware or sysplex-unaware capabilities for the mount mode. (The mount mode can be either read-only or read/write.)

Migrating to z/OS 1.10 - Part 2

The changes to the AUTOMOVE processing affect z/OS V1R9 systems only. If you have a mixed-release configuration, you may see the original behavior on systems prior to z/OS V1R9. For example:

- Before z/OS V1R9, if you mounted a file system in a sysplex-aware mount mode with the NOAUTOMOVE setting, the NOAUTOMOVE setting was changed to AUTOMOVE(YES) and message BPXF234I was issued. On a z/OS V1R9 system, if you mount a file system in a sysplex-aware mount mode with NOAUTOMOVE, that setting is honored. However, if the file system ownership is moved to a system earlier than z/OS V1R9, the NOAUTOMOVE setting is changed to AUTOMOVE(YES). File systems that are mounted with a system list are also changed to AUTOMOVE(YES).
- If you mount a file system in a sysplex-aware mount mode with the NOAUTOMOVE setting, all z/OS V1R9 systems will honor the setting but systems earlier than z/OS V1R9 will not. For example, if a z/OS V1R9 owner system fails and there is a z/OS V1R8 system in the shared file system configuration, the z/OS V1R8 system will attempt to take over the file system and become the new owner.

Migration action: Review MOUNT statements that specify NOAUTOMOVE, UNMOUNT, or a system list and determine whether they should be updated. Notify operators about the changed mount behavior, if needed.

Tip: To find out if the file system is sysplex-aware, use the D OMVS,F command to display each mounted file system.

- Those file systems for which this system is not the owner, and for which it says Client=N, are *sysplex-aware*. They are not clients and are locally mounted to the PFS.
- Those file systems for which this system is not the owner, and for which it says Client=Y, are *sysplex-unaware*.

Increase dump space to accommodate dumps of the zFS address space (Required-IF, as of R9)

Required if the existing dump data space is not large enough to accommodate additional dumps.

Before z/OS V1R9, system dumps did not include dumps from the zFS address space. In z/OS V1R9 and later, system dumps capture data from the zFS address space, as long as the address space is active and the cause of the dump is related to a file system problem. Consequently, a system dump can now be larger than before z/OS V1R9.

Migration action: Determine if you need to increase your dump data sets to include the additional space needed for data dumped from the zFS address space. The amount of increase depends on the amount of zFS activity and the configuration parameters. For example, the `trace_table_size` parameter specification may greatly affect the amount of storage that needs to be captured in the dump data set.

Update awk invocations to accommodate changes made to the awk utility (Required-IF, as of R9)

Required if either (or both) of the following situations is present:

- If **awk** invocations use command-line arguments that have an "=" but do not follow the rules for a variable name. For example, `./x=y`.
- An **awk** program contains a break or continue statement outside a loop.

Before z/OS V1R9, the **awk** utility did not comply with Single UNIX Specification, Version 3 (SUSv3). In z/OS V1R9, the **awk** utility has been enhanced to conform with SUSv3.

- Certain arguments that were treated as variable assignments are now interpreted as path names. Prior to z/OS V1R9, **awk** recognized command-line arguments with an "=" character as a variable assignment. Starting in z/OS V1R9, **awk** will only recognize an argument as a variable assignment if it consists of the following characters (in the order listed):

1. An underscore or alphabetic character
2. A sequence of underscores, digits, and alphabetic characters
3. The "=" character

If the argument does not follow that format, **awk** interprets it as a path name. Most **awk** commands will not have arguments that break the variable name rules.

- If a break or continue statement occurs outside a loop, **awk** either exits with an error message or, with the PTF for APAR OA22253 installed, displays a warning message, exits the current pattern or function, and continues processing. Previously, **awk** exited the current pattern or function without an error message and continued processing. Most **awk** programs will not have a break or continue statement that occurs outside a loop.

Migration action: Modify any **awk** invocations (for example, in shell scripts) that have command-line arguments with an "=" but break the rules for a variable name.

If you have an **awk** program that contains a break or continue statement outside a loop (which is reported by an error message or, with the PTF for APAR OA22253 installed, a warning message from **awk**), correct the program.

Use the new default of 64000 for the MAXFILEPROC statement (Recommended, as of R9)


Migrating to z/OS 1.10 - Part 2

Not required, but recommended if you use IBM Health Checker for z/OS to check whether the value of MAXFILEPROC is too low. (The check is USS_MAXSOCKETS_MAXFILEPROC.) If the value is 256, an exception message will be issued.

Before z/OS V1R9, the default for the MAXFILEPROC statement in the BPXPRMxx parmlib member was 256. In z/OS V1R9, the default is 64000.




Migration action: The recommended value is 64000 or higher for the MAXFILEPROC statement in BPXPRMxx. Either use the default of 64000 or specify a higher number.

z/OS 1.10 Migration Actions



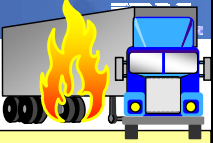
"Big Migs" for z/OS R10 from z/OS R8

Migration Actions You Should NOT Overlook:

- Add or change volumes to keep file systems in a single data set (Recommended)
- Update automation to accommodate new responses to the DISPLAY CONSOLES command (Required-IF, as of R10)
- Provide an appropriate HZSPDATA data set on fallback (Req-IF, as of R10)
- Reallocate the WLM couple data set (Required-IF, as of R10)
-  • Update callers of CNZMXURF that do not hold appropriate serialization (Required-IF, as of R10)
- Remove your *run-time* dependency on the C/C++ IBM Open Class Library (Required-IF, as of R9)
-  • IP Services: Migrate to the TN3270E Telnet server that runs in its own address space (Required-IF, as of R9)
- SNA Services: Ensure compatible levels of VTAM for HPR sessions (Required-IF, as of R10)
-  • ICSF HCR7750: Increase the size of your PKDS (Required, as of HCR7750)
- Integrated Security Server LDAP: Migration from LDAP Server to IBM TDS (Recommended)




32 © 2008 IBM Corporation

z/OS 1.10 Migration Actions



"Big Migs" for z/OS R10 from z/OS R8

Migration Actions You Should NOT Overlook:

-  • Redefine existing VSAM data sets that contain the IMBED, REPLICATE, and KEYPAGE attributes (Recommended)
-  • Migration from zFS multi-file system aggregates to compatibility mode aggregates (Recommended)
- Accommodate JES2 dynamic exits function (Required-IF, as of OA23561/OA21346)
- Use CEEPRMxx's CEEROPT instead of making copies of Language Environment initialization routines (Required-IF, as of R10)
-  • Migrate from HFS to zFS file systems (Recommended)

33 © 2008 IBM Corporation

z/OS 1.10 Migration Actions

z/OS V1 R10 Install-Related Enhancements

► Documentation Enhancements for z/OS Migration:


- Customized books contain all supported migration paths to z/OS R10!
z/OS R8 to z/OS R10, and z/OS R9 to z/OS R10,
- “Tips” have been added to appropriate migration actions for related IBM Health Checker checks (and Health Checks that are for migration).
- How to use the IBM Health Checker for migration checking is explained.

► ServerPac Deliverable:

► ServerPac in z/OS R10:

- ★ **ACCEPT RSU***: now, only RSU* and Integration Tested Service will be ACCEPTed, to make it easier to RESTORE some PTFs.
- ★ **SYSUT* and SMPWRK* values in SMP/E CSI and jobs**: updated to be more “appropriate” for large (and electronic) installs.
- ★ **SDSF for JES3 support**: SDSF will be in a separate zone, and merged when JES merge is selected. (You may merge both or neither JESes into the z/OS zone.) Removal of unused JES is unaffected.
- ★ **CEA and SYSREXX Setup**: Security customization is performed.

► (As of R9: ServerPac: **Installing Your Order** document is available in Adobe PDF format, in addition to the existing BOOK format. The obsolete LIST1403 and LIST3820 formats are removed.)...**and lots of other goodies from prior releases !**



34
© 2008 IBM Corporation

z/OS R10 Installation Enhancements in ServerPac

ACCEPT RSU*

For ServerPac orders, service is integrated with product code according to the following timeline:

- ServerPac is refreshed every month to pick up the most current RSU.
- All products incorporate high impact or pervasive (HIPER) and PTF-in-error (PE) fixes that are available up to the day your order is built. Because these PTFs are still fairly new at the time your order is built, they are not SMP/E ACCEPTed. They have been APPLYed only, therefore allowing you to back them off of your system using the SMP/E RESTORE command, if necessary. Your order also includes (in its SMP/E global zone) the latest unintegrated service that was available at the time the order was created. In addition, the PTFs that were SMP/E APPLYed but not ACCEPTed are also in the SMP/E global zone.
- All z/OS elements and features will incorporate additional service that has been through platform (integration) testing (SOURCEID ZOSV1Rn, where n is the release number).

Prior to z/OS R10, all PTFs that were APPLYed in ServerPac, were also ACCEPTed. As of z/OS R10, only the latest RSU (SOURCEID RSU*) and the PTFs with SOURCEID ZOSV1Rn are ACCEPTed.

The publication *ServerPac: Installing Your Order*, which comes with your order, specifies the integrated service level applicable to your order. Your ServerPac order will also include unintegrated service that was available when your order was built.

SYSUT* and SMPWRK* values in SMP/E CSI and jobs

The values for SMP/E’s SYSUT and SMPWRK data sets have (in some cases) been too small for installing large PTFs or large electronic orders. To accommodate this, the SYSUT and SMPWRK values have been increased in the ServerPac deliverable. These are the values that will be provided in ServerPac (in 3390 tracks):

SMPWRK1	PDS FB 80	150
SMPWRK2	PDS FB 80	150
SMPWRK3	PDS FB 80	9500
SMPWRK4	PDS FB 80	132
SMPWRK6	PDS FB 80	20000
SYSUT1	SEQ -- --	7500
SYSUT2	SEQ -- --	1500
SYSUT3	SEQ -- --	960
SYSUT4	SEQ -- --	3000

SDSF for JES3 Support

With z/OS R10 SDSF, support for JES3 has been added. To accommodate this in ServerPac, the “JES Element Selection” panel will do some enhanced processing.

On this panel, select at least one JES element for installation: JES2 (and SDSF) or JES3 (and SDSF). If your installation requires both JES elements to be installed, you can select BOTH.

For each JES element you select, specify whether the dialog is to merge the SMP/E zones of the JES element with the Base Control Program (BCP) zones. When you merge a JES element with the BCP zones, the SDSF zone is automatically merged with the BCP zones:

- If you specify the JES2 zone to be merged with the BCP zones, both JES2 and SDSF zones are merged with the BCP zones. Similarly, if you specify JES3 to be merged with the BCP zones, both JES3 and SDSF zones are merged with the BCP zones.
- If you specify both JES elements to be merged with the BCP zones, the JES2, JES3, and SDSF zones are merged with the BCP zones.

Note: If you install both JES elements, you cannot merge only one of the JES elements with the BCP zones. The error message CPPP0601011E will appear to indicate that the action is not allowed.

If you migrate JES2 or JES3 together with the rest of z/OS, IBM recommends that you specify **Y** on this panel to merge the selected JES elements into the BCP zone. However, if you plan to stage your z/OS and JES migrations separately, do not merge zones. The dialog loads the JES element's zones, but does not merge them.

CEA and SYSREXX Setup

Security customization (provided in ServerPac job RACFTGT) now includes the necessary security customization for CEA and System REXX.

z/OS 1.10 Migration Actions


z/OS R10 Install-Related Enhancements

Summary of Enhancements of System Programmer Interest

- ☺ **ISPF:** Multiple targets for MOVE/COPY, Enhanced screen swapping, and Block commands in 3.4
- ☺ **Language Environment:** Syntax checking tool for CEEPRMxx parmlib members
- ☺ **DFSORT:** Using PARMLIB members to set installation options


R9 goodies you also get in R10:

- ❖ **zFS:** IOEAGFMT removes SUPERUSER.FILESYS.PFSCNTL requirement to format a zFS in batch.
- ❖ **SDSF:** REXX interface to SDSF
- ❖ **ISPF:** Support for editing and browsing z/OS UNIX and ASCII files



35

© 2008 IBM Corporation

z/OS 1.10 Migration Actions


z/OS R10 Install-Related Enhancements

ISPF: Various Enhancements

- **Multiple targets for ISPF Editor MOVE and COPY**
 - Nice when you want to put the same line into multiple places in one fell swoop!
 - ★ **Exploitation:** Use AK (or BK) on the M or C line command, then A (or B) when done.

- **ISPF enhanced screen swapping**
 - View all the logical screens in your ISPF session and navigate more easily between them. Just point-and-shoot to switch!
 - ★ **Exploitation:** Use command SWAPBAR ON.

- **Block commands allowed in ISPF Data Set List Utility**
 - You can now enter line commands in blocks using the ISPF Data Set List Utility (option 3.4)
 - ★ **Exploitation:** Use //x and // to mark blocks and do the command (where 'x' is your line command).

36
© 2008 IBM Corporation

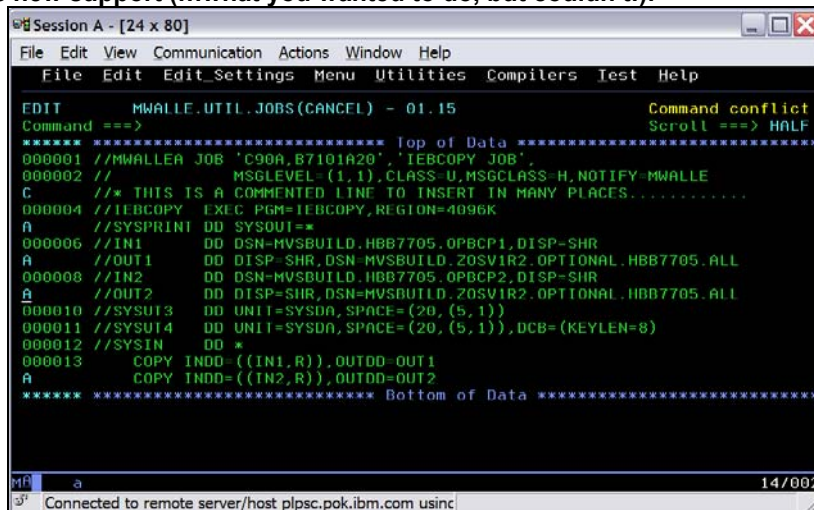
ISPF Various Enhancements

Multiple targets for ISPF Editor MOVE and COPY

The enhanced ISPF Editor MOVE and COPY primary commands lets you to move or copy data to many locations within the edit session in one operation, reducing the time you spend editing files. Previously, you would have had to move or copy data to one location at a time.

When data is to be moved or copied, the A (after) line command specifies the line after which the data is to be placed. When data is to be moved or copied to multiple destinations, the A (after) line command specifies the final destination line after which the data is to be placed. When data is to be moved or copied to multiple destinations, the AK (after, multiple targets) line command specifies each multiple destination line (apart from the final destination line) after which the data is to be placed

Example, without this new support (...what you wanted to do, but couldn't!):



```

Session A - [24 x 80]
File Edit View Communication Actions Window Help
File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT      MWALLE.UTIL.JOBS(CANCEL) - 01.15      Command conflict
Command ==>                                Scroll ==> HALF
***** Top of Data *****
000001 //MWALLEA JOB 'C90A,B7101A20','IEBCOPY JOB'.
000002 //      MSGLEVEL=(1,1),CLASS=U,MSGCLASS=H,NOTIFY=MWALLE
C      /* THIS IS A COMMENTED LINE TO INSERT IN MANY PLACES.....
000004 //IEBCOPY EXEC PGM=IEBCOPY,REGION=4096K
A      //SYSPRINT DD SYSOUT=*
000006 //IN1 DD DSN=MVSBUILD.HBB7705.OPBCP1,DISP=SHR
A      //OUT1 DD DISP=SHR,DSN=MVSBUILD.ZOSVIR2.OPTIONAL.HBB7705.ALL
000008 //IN2 DD DSN=MVSBUILD.HBB7705.OPBCP2,DISP=SHR
A      //OUT2 DD DISP=SHR,DSN=MVSBUILD.ZOSVIR2.OPTIONAL.HBB7705.ALL
000010 //SYSUT3 DD UNIT=SYSDA,SPACE=(20,(5,1))
000011 //SYSUT4 DD UNIT=SYSDA,SPACE=(20,(5,1)),DCB=(KEYLEN=8)
000012 //SYSIN DD *
000013 COPY INDD=((IN1,R)),OUTDD=OUT1
A      COPY INDD=((IN2,R)),OUTDD=OUT2
***** Bottom of Data *****

```

Example with this new support, as of R10:

```

Session A - [24 x 80]
File Edit View Communication Actions Window Help
File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT MMALLE.UTIL.JOBS(CANCEL) - 01.15 Columns 00001 00072
Command ==> Scroll ==> HALF
***** Top of Data *****
000001 //MMALLEA JOB 'C90A,B7101A20', 'IEBCOPY JOB',
000002 // MSGLEVEL=(1,1),CLASS=U,MSGCLASS=H,NOTIFY=MMALLE
000003 /* THIS IS A COMMENTED LINE TO INSERT IN MANY PLACES.....
000004 //IEBCOPY EXEC PGM=IEBCOPY,REGION=4096K
AK0005 //SYSPRINT DD SYSOUT=*
000006 //IN1 DD DSN=MVSBU1D.HBB7705.OPBCP1,DISP=SHR
AK0007 //OUT1 DD DISP=SHR,DSN=MVSBU1D.ZOSVIR2.OPTIONAL.HBB7705.ALL
000008 //IN2 DD DSN=MVSBU1D.HBB7705.OPBCP2,DISP=SHR
AK0009 //OUT2 DD DISP=SHR,DSN=MVSBU1D.ZOSVIR2.OPTIONAL.HBB7705.ALL
000010 //SYSUT3 DD UNIT=SYSDA,SPACE=(20,(5,1))
000011 //SYSUT4 DD UNIT=SYSDA,SPACE=(20,(5,1)),DCB=(KEYLEN=8)
000012 //SYSIN DD *
000013 COPY INDD=((IN1,R)),OUTDD=OUT1
000014 COPY INDD=((IN2,R)),OUTDD=OUT2
***** Bottom of Data *****
19/003
Connected to remote server/host.plpsc.pok.ibm.com using

Session A - [24 x 80]
File Edit View Communication Actions Window Help
File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT MMALLE.UTIL.JOBS(CANCEL) - 01.15 Columns 00001 00072
Command ==> Scroll ==> HALF
***** Top of Data *****
000001 //MMALLEA JOB 'C90A,B7101A20', 'IEBCOPY JOB',
000002 // MSGLEVEL=(1,1),CLASS=U,MSGCLASS=H,NOTIFY=MMALLE
000003 /* THIS IS A COMMENTED LINE TO INSERT IN MANY PLACES.....
000004 //IEBCOPY EXEC PGM=IEBCOPY,REGION=4096K
000005 //SYSPRINT DD SYSOUT=*
000006 /* THIS IS A COMMENTED LINE TO INSERT IN MANY PLACES.....
000007 //IN1 DD DSN=MVSBU1D.HBB7705.OPBCP1,DISP=SHR
000008 //OUT1 DD DISP=SHR,DSN=MVSBU1D.ZOSVIR2.OPTIONAL.HBB7705.ALL
000009 /* THIS IS A COMMENTED LINE TO INSERT IN MANY PLACES.....
000010 //IN2 DD DSN=MVSBU1D.HBB7705.OPBCP2,DISP=SHR
000011 //OUT2 DD DISP=SHR,DSN=MVSBU1D.ZOSVIR2.OPTIONAL.HBB7705.ALL
000012 /* THIS IS A COMMENTED LINE TO INSERT IN MANY PLACES.....
000013 //SYSUT3 DD UNIT=SYSDA,SPACE=(20,(5,1))
000014 //SYSUT4 DD UNIT=SYSDA,SPACE=(20,(5,1)),DCB=(KEYLEN=8)
000015 //SYSIN DD *
000016 COPY INDD=((IN1,R)),OUTDD=OUT1
000017 COPY INDD=((IN2,R)),OUTDD=OUT2
000018 /* THIS IS A COMMENTED LINE TO INSERT IN MANY PLACES.....
***** Bottom of Data *****
23/002
Connected to remote server/host.plpsc.pok.ibm.com using
    
```

Reference information: z/OS ISPF Edit and Edit Macros

ISPF enhanced screen swapping

This enhancement allows you to view all the logical screens in your ISPF session and navigate more easily between them. ISPF now provides a new system command, SWAPBAR, which displays a line at the bottom of the physical screen that contains point-and-shoot fields showing the logical screens for your ISPF session. You can put the cursor on the point-and-shoot field for a logical screen, press Enter, and the logical screen becomes the active screen.

The list of logical sessions can be activated by entering the SWAPBAR or SWAPBAR ON command on the command line. The list is displayed on the last line of the physical screen.

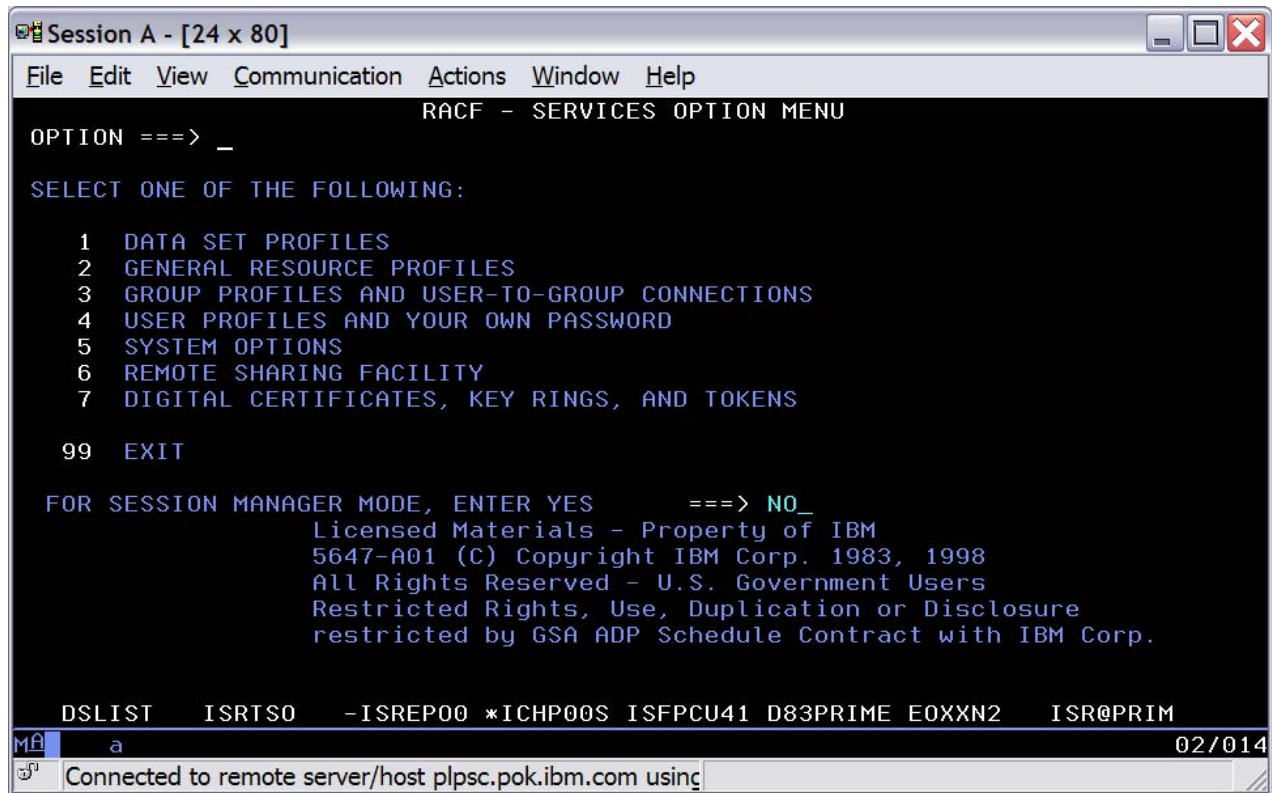
The entry for each logical session is the screen name if assigned or, if a screen name is not available, it is the panel name of the current panel displayed for the logical session. The entry for the active logical session has an asterisk (*) in the first character position and, if the name is 8 bytes long, the last character is not displayed. Also, the alternate logical session has a "-" in the first position and the 8th character is not displayed.

The list remains active until you enter the SWAPBAR or SWAPBAR OFF command. The setting for the SWAPBAR is maintained in the system profile member and applies across logons.

If the list is longer than the width of the screen, a ">" appears at the right of the list to indicate there are more entries and you can scroll right by either positioning the cursor on the ">" and pressing Enter, or by positioning the cursor on an entry and pressing PF11 which scrolls to the entry indicated. When the start of the list is not displayed, a "<" is displayed at the left of the visible part of the list to indicate that you can scroll left on the list by positioning the cursor on the "<" and Pressing Enter or by positioning the cursor on an entry and pressing PF10.

The active logical session can be changed by positioning the cursor on an entry and pressing Enter. This logical session then becomes the active logical session. The SWAPBAR entries use the same physical attribute as the action bar choices and, if in the options settings **Tab to action bar choices** is selected, then tabbing to swapbar entries also occurs.

If the cursor is positioned on the swapbar entry for the currently active session and Enter is pressed, this is treated the same as Enter being pressed within the active logical session panel. When the SWAPBAR is activated, the **Always show split line** option is deactivated and you are not able to reactivate it until the SWAPBAR is deactivated.



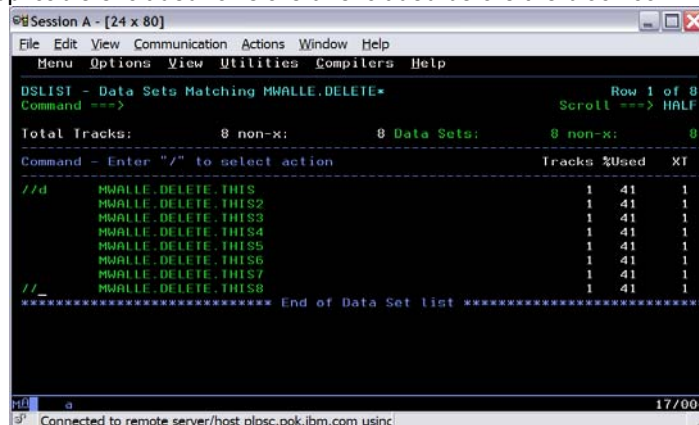
Reference information: *z/OS ISPF User's Guide Vol I, and z/OS ISPF Dialog Developer's Guide and Reference*

Block commands allowed in ISPF Data Set List Utility

You can now enter line commands in blocks using the ISPF Data Set List Utility (option 3.4). This greatly reduces the number of line commands you are required to enter on the data set list utility when you need to process a number of ISPF data sets grouped together in the list. For example, if you want to delete all the data sets displayed in the data set list, you are only required to enter the delete block line command against the first and last data sets in the list.

You mark the block by typing a "/" at the beginning of a block of rows and another "/" at the end of the block of rows. You must type the line command either immediately after the // on the first row of the block, or immediately after the // on the last row of the block.

You can enter several blocks of commands at the same time, but you cannot nest them. Single line commands are not allowed within a block command. You can execute all line commands, including TSO commands, Clists and REXX execs as block commands. If you have selected the DSLIST settings option **Execute Block Commands for excluded Data Sets**, all applicable excluded rows are unexcluded before the block commands are executed.




z/OS 1.10 Migration Actions

z/OS R10 Install-Related Enhancements

LE: Syntax checking tool for CEEPRMxx parmlib members

- **In R10, you can check the syntax of new/changed CEEPRMxx parmlib members from batch or from TSO.**
 - Parmlib member must be in a FB80 data set. Can specify concatenated members, which is nice for verification of what will be used.
 - For TSO, SCEECLST must be in SYSPROC or SYSEXEC concatenation.
 - When no errors are found, you get a nice report of the run-time options for each group and “Last Where Set”!
 - When errors are found, you get messages for what is wrong and where.



★ **Exploitation:**

★ **For batch:**

```
//CEEPRMCJ EXEC PGM=CEEPRMCC,
// PARM='CEE=(MW) '
//CEEPRMCK DD DSN=MWALLE.CEE.PARMLIB,DISP=SHR
```

★ **For TSO:**

```
====> CEEPRMCK MEMBERS(MW) DSN('MWALLE.CEE.PARMLIB')
```

37
© 2008 IBM Corporation

Syntax checking tool for CEEPRMxx parmlib members

Language Environment is enhanced to provide a tool for syntax checking of one or more CEEPRMxx members of PARMLIB before attempting to activate them. CEEPRMxx members can be syntactically checked for errors under z/OS batch or TSO/E. Before calling the syntax checker, a CEEPRMxx member must be created and placed in a PDS or PDSE.

Under z/OS batch

The CEEPRMCC program reads and parses the CEEPRMxx member for syntax errors. The CEEPRMCC program displays the run-time options report if no errors are found. The run-time options report only displays options that are specified inside the CEEPRMxx members. The CEEPRMCC expects the following inputs:

- The PARM parameter Use the PARM parameter of the EXEC job control statement to select one or more CEEPRMxx members.

```
// PARM='CEE=(xx,yy,...,nn)'
```

The two alphanumeric characters, xx,yy,...,nn, are the suffix of the CEEPRMxx members to be checked. Embedded blanks are not allowed within the PARM.
- Optional CEEPRMCK DD Statement Use DD statement to specify the data set where CEEPRMxx members are located. //CEEPRMCK DD DSN=MEENAK.SYSTEM.PARMLIB,DISP=SHR If no DD is specified, the CEEPRMCC program uses the default data set SYS1.PARMLIB.

Example:

```
//CEEPRMCJ EXEC PGM=CEEPRMCC,
// PARM='CEE=(MW) '
//CEEPRMCK DD DSN=MWALLE.CEE.PARMLIB,DISP=SHR
```

Usage notes An input data set must be a fixed record format and a record length of 80.

Under TSO/E

Under TSO/E, CEEPRMCK reads and parses the CEEPRMxx member for syntax errors. CEEPRMCK displays the run-time options report if no errors are found. The run-time options report only displays options that are specified inside the CEEPRMxx members. The syntax of the CEEPRMCK invocation is as follows:

CEEPRMCK MEMBERS(xx) DSN (data-set-name)

- **xx** The two alphanumeric characters that are the suffix of the CEEPRMxx members to be checked. The **MEMBERS** keyword parameter must always be specified.

Migrating to z/OS 1.10 - Part 2

- **data-set-name** The data set name that contains the specified CEEPRMxx member. The fully qualified data set name must be enclosed in single quotes if a TSO/E prefix is not required. The **DSN/DSNAME** keyword parameter is optional. If both DD is allocated and DSN or DSNAME is specified, the CEEPRMCK program uses the DD and the DSN/DSNAME is ignored. DD allocation overrides DSN/DSNAME specification. If no DD is allocated and no DSN or DSNAME is specified, the CEEPRMCK program uses the default data set SYS1.PARMLIB.

Example: CEEPRMCK MEMBERS (MW) DSN ('MWALLE.CEE.PARMLIB')

Usage notes:

- An input data set must be a fixed record format and a record length of 80.
- To invoke CEEPRMCK by using the documented syntax, SCEECLST must be allocated to a system file (SYSPROC or SYSEXEC). See *z/OS TSO/E REXX User's Guide* for more information about setting up and using REXX execs.

Sample Output

If there are syntax errors, no run-time options report will be displayed. Error messages will be written to the Language Environment message file. CEE3761I will be followed by other existing error messages related to syntax errors in CEEPRMxx members and end with CEE3762I.

Here is the partial output, when no syntax errors are found:

```
CEE3762I The Language Environment Parmlib checker has completed.
```

```
CEE3745I 16.38.52 Display CEEDOPT
CEE= (MW)
```

LAST WHERE SET	OPTION
PARMLIB (CEEPRMMW)	ABPERC (NONE)
PARMLIB (CEEPRMMW)	ABTERMENC (ABEND)
PARMLIB (CEEPRMMW)	NOAIXBLD
PARMLIB (CEEPRMMW)	ALL31 (ON)
PARMLIB (CEEPRMMW)	ANYHEAP (16384, 8192, ANYWHERE, FREE)
PARMLIB (CEEPRMMW)	NOAUTOTASK
PARMLIB (CEEPRMMW)	BELOWHEAP (8192, 4096, FREE)
PARMLIB (CEEPRMMW)	CBLOPTS (ON)
PARMLIB (CEEPRMMW)	CBLPSHPOP (ON)
PARMLIB (CEEPRMMW)	CBLQDA (OFF)
PARMLIB (CEEPRMMW)	CEEDUMP (60, SYSOUT=*, FREE=END,
...	...

Here is the complete output, when syntax errors are found. (Where "WEIRD" were the violations that were being detected):

```
CEE3761I The following messages pertain to the call to the Language Environment Parmlib checker.
CEE3731I The following messages pertain to the system default run-time options in the CEEDOPT in CEEPRMMX
CEE3616I The string 'WEIRD' was not a valid or supported suboption of the run-time option RPTOPTS in this
release.
CEE3731I The following messages pertain to the system default run-time options in the CEEDOPT in CEEPRMMX.
CEE3614I An invalid character occurred in the numeric string 'WEIRD' of the run-time option DEPTHCONDLMT.
CEE3731I The following messages pertain to the system default run-time options in the CELQDOPT in CEEPRMMX.
CEE3616I The string 'WEIRD' was not a valid or supported suboption of the run-time option DEPTHCONDLMT.
CEE3749I The value specified for keyword CEEROPT found in CEEPRMMX is not valid.
CEE3762I The Language Environment Parmlib checker has completed.
```

Reference information: For more information on the tool for checking the syntax of CEEPRMxx members, see: *z/OS Language Environment Customization*.

z/OS 1.10 Migration Actions

z/OS R10 Install-Related Enhancements

DFSORT: Using ICEPRMxx to set installation options

- Each ICEPRMxx member can contain options to be changed for any or all of DFSORT's eight installation environments.
- Up to ten ICEPRMxx members can be activated by a START ICEOPT started task command. The options in the activated members will be merged with the ICEMAC defaults at run-time.
- Old way, ICEMAC macro and usermods, are still supported, but this new way is much better!

★ **Exploitation:**

- ★ **Read *DFSORT Installation and Customization...basically you:***
 - ★ **code up your desired ICEPRMxx values**
 - ★ **activate them with command `start iceopt,iceprm=xx`**
 - ★ **run an ICETOOL DEFAULTS report to verify that your installation defaults are as expected**

38
© 2008 IBM Corporation

Using ICEPRMxx parmlib members to set installation options

ICEPRMxx members in concatenated PARMLIB can now be used to specify changes to DFSORT's installation options. Each ICEPRMxx member can contain options to be changed for any or all of DFSORT's eight installation environments (JCL, INV, TSO, TSOINV and TD1-TD4). Up to ten ICEPRMxx members can be activated by a START ICEOPT started task command. The options in the activated members will be merged with the ICEMAC defaults at run-time. A different ICEPRMxx member, or combination of ICEPRMxx members, for different LPARs can be activated at IPL time by including a START ICEOPT command in an appropriate COMMNDxx member in PARMLIB, or at any time by issuing a START ICEOPT command from the console.

Note: You can use MVS console security to control who can issue the START ICEOPT command. The user ID that is assigned to the address space for the ICEOPT started task must have READ authority to the PARMLIB data sets. (For instance, ADDSD 'SYS1.PARMLIB' UACC(NONE) and PERMIT 'SYS1.PARMLIB' CLASS(DATASET) ID(iceid) ACCESS(READ) .)

ICEPRMxx members are now the recommended way to change DFSORT installation defaults since they are easier to use and more flexible than the old method using the ICEMAC macro and usermods. However, the old method is still supported. The ICETOOL DEFAULTS operator can be used at any time to produce a report showing the merged PARMLIB/ICEMAC installation default values for each environment that will be used at run-time, as well as the active ICEPRMxx and ICEMAC values.

Coding Options in ICEPRMxx members

You can specify installation options for any or all of the eight environments in an ICEPRMxx member of concatenated PARMLIB. You must precede each group of options with one of the following keywords to indicate the installation **Installation Defaults**. See the *DFSORT Installation and Customization* book for syntax requirements.

Activating ICEPRMxx Members

To change the installation defaults with one ICEPRMxx member in PARMLIB, issue a START ICEOPT command to activate that member as follows: `start iceopt,iceprm=xx` where xx is the suffix of the ICEPRMxx member you want to use. For example, to activate ICEPRM01, use:

```
start iceopt,iceprm=01
```

To change the installation defaults with more than one ICEPRMxx member in PARMLIB, issue a START ICEOPT command to activate those members as follows:

```
start iceopt,iceprm=(x1,...,xn)
```

where *x1,...,xn* are the suffixes of the ICEPRMxx members you want to use.

Migrating to z/OS 1.10 - Part 2

For example, to activate ICEPRMMN and ICEPRMS3, type the following on the console:

```
start iceopt,iceprm=(mn,s3)
```

You can use a START ICEOPT command at any time to change your installation defaults by activating up to 10 ICEPRMxx members. You can issue a START ICEOPT command from the console, or include it in an appropriate COMMNDxx member in PARMLIB so it will be executed automatically at IPL time. However, we recommend that you test your ICEPRMxx members by activating them from the console before activating them automatically with a COMMNDxx member. ICEOPT issues messages to the console to indicate the status of your installation default changes as well as any errors it detects. Be sure to review these console messages and the return code from ICEOPT.

After you execute a START ICEOPT command, you can run an ICETOOL DEFAULTS report to verify that your installation defaults are as expected.

Reference information: *z/OS DFSORT Installation and Customization*



z/OS 1.10 Migration Actions Summary

- **General Migration Actions for Everyone:**
 - Default MEMLIMIT is now 2. Removed DFSORT libraries. Changed Health Checks. CADS: I for R10, I for R9.
- **BCP Migration Actions:**
 - Consoles: unique names and logon IDs, LOGON and LOGOFF reserved, unsupported subsystem console functions, ALLOWUSERKEYCSA default is NO, Remove GRS monitor PPT entries, No VOLUME on DISP=UNCATLG.
- **Communications Server Migration Actions:**
 - Removed functions, Shared FTP.DATA between FTP server and client, Make sure all HPR-capable VTAMs are R8 or higher, More region size for Telnet
- **DFSMS Migration Actions:**
 - Control data set ID for RMM, changes for EAV support, change in CA sizes assigned DEFrag command in DSS, new facility class names for (H)BACKDS NEWNAME command.
- **JES2 Migration Actions:** Automation for some \$D commands.
- **LE Migration Actions:** New runtime options (no change in behavior, though).
- **z/OS UNIX Migration Action:**
 - /etc files updated, increase dump space for zFS, submit command, ln changes.
- **z/OS R10 Installation Enhancements:** for z/OS Migration and for ServerPac
- **z/OS R10 SysProg Enhancements:** various ISPF, CEEPRMxx checker, ICEPRMxx support