

**z/OS 1.10: Migration – Part 2 of 3**

This is part two of a three-part session that will be of interest to System Programmers and their managers who are migrating to z/OS 1.10. It is strongly recommended that you review all three of these sessions for a complete migration picture.

In part two, covered will be the specific migration actions for getting to z/OS 1.10. Selected elements such as the BCP, C/C++, Communications Server, and Distributed File Service (DFS) will be included.

*The general availability date for z/OS V1 R10 is planned for September 26, 2008.*

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	DFPMS/Stream	FICON	MVS/SP	PROSM	VTAM
ADK	DFPMS/Stream	HyperStrike	NetSpool	Print Services Facility	WebSphere z
Applet	DFP/RT	IBM	NetView	PrintWiz	zArchitecture
IBM Mail Manager	Eckha	IBM Link	Open Class	RACF	zOS
CICS	ESCON	IMS	OS/390	Resource Link	zVM
DB2	Eserv	InPath	Parallel Sysplex	RSAP	zSeries
DB2 Universal Database	FICON	Language Environment		S/390	
DFPMS/SP		Multiple		S/390 Parallel Enterprise Server	
DFPMS/Stream		MVS		Sysplex Timer	
		MVSESA		SystemPac	
				System z.5	
				TSM	
				TotalStorage	
				Vantage	
				VSE/ESA	

\* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Java and all Java-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 an inherent throughput rate (TPH) value based on macroscopic and subjective using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of work being done by the user's jobstream, the I/O configuration, the change configuration, and the workload pattern. Throughput is an average; it is not a guarantee of performance. Actual performance may vary.

IBM trademarks are mentioned here to provide a point of reference for the user's information. The use of the trademarks of other companies is for identification purposes only. All other trademarks are the property of their respective owners.

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

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is intended to illustrate the advantages of those products or their published specifications. IBM does not endorse 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 those products.

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

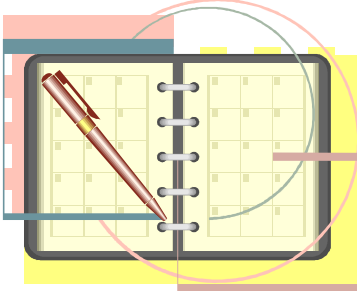
ENABLING BUSINESS.  
A THROUGH Z.

© 2008 IBM Corporation


z/OS 1.10: Migration Part 2 IBM

## z/OS 1.10: Migration Part 2 Agenda

- Migration Actions
  - General Migration Actions for Everyone
  - Element-Specific Migration Actions for z/OSR10
    - BCP
    - XL C/C++
    - Communications Server
    - From various elements:
      - ISPF
      - ICKDSF
      - Integrated Security Server LDAP
      - Security Server RACF
      - TSO/E
- Summary



3 © 2008 IBM Corporation


z/OS 1.10: Migration Part 2 

## Structure of z/OS R10: Migration Presentations

- f Part 1**
  - Planning information for R10: Content, Coexistence, System Requirements, ...
- f Part 2**
  - Migrations actions for z/OSR10 from selected elements you just saw
- f Part 3**
  - Migrations actions for z/OSR10 from selected elements:
    - DFSMS
    - Distributed File Service - zFS
    - Language Environment
    - JES2 and JES3
    - TSO/E
    - z/OSUNIX
  - Also included are highlights of some enhancements you'll see in z/OSR10 for system programmers.


**You need to review all 3 to get a complete migration picture!**  
*(along with z/OS Migration book!)*


4 © 2008 IBM Corporation

z/OS 1.10: Migration Part 2


## Definition of "Migration Action":

- f* **Migration is**
  - the installation of a new version or release of a program to replace an earlier version or release.
- f* **After a successful migration, the applications and resources on the new system function the same way they did on the old system, if possible.**
- f* **Migration does not include exploitation of new functions except for new functions that are now required to use.**
- f* **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
- f* **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

### 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.

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 Part 2


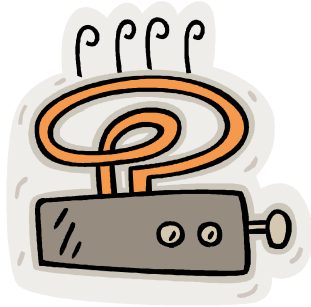
## Elements with Migration Actions for z/OS R10

*f* Documented in **z/OS V1R10 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."

*f* **From z/OS R8 to z/OS R10:**

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



➤ means that some of that element's migration actions follow within this session and the next one
 

- means that you need to look for that element's migration actions in the book

6
© 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 Part 2

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

**f Migration Actions You Can Do NOW :**

- **Install coexistence and fallback PTs (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.

**f Migration Actions Pre-First IPL:**

- **Verify that virtual storage limits are set properly (Required)**
  - Default MEMLIMIT changed in z/OSR10 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/OSR10: HASES20 (JES2), IATINTK (JES3), and IXGBLF01 (system logger initialization).
    - The NOHONORIEFUSREG keyword, new in z/OSV1R10, is set for those programs in the IBM-defined default program properties table (PPT).

7 © 2008 IBM Corporation

z/OS 1.10: Migration Part 2

## 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 SYS1.SICE\* data sets (may have logon proc impacts)..
- **Add references to new data sets and paths (Required)**
  - New in R10: SDSF data sets (ISF.SSFMOD1 needs to be in the link list and APF-authorized)! ...
  - New in R9: System Rexx data sets SYS1.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/OSCIM server, it automatically started and does not terminate.
  - New in R9: **ARCnXXXX**: One of these DFSMSdss address spaces is started automatically by DFSMShsm whenever a dump, restore, migration, backup, recover, or CDSbackup function is invoked. Terminates when hsm terminates.


8 © 2008 IBM Corporation

z/OS 1.10: Migration Part 2

## 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/OSR6).
  - **IBM Health Checker for z/OS can help you determine what to specify for the MAXCAD value!**



9

© 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* 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/atsmastr.nsf/WebIndex/PRS268>
  - Business partners: <http://partners.boulder.ibm.com/src/atsmastr.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:
      - 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)
  - 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 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 HONORIEFUSIREG | NOHONORIEFUSIREG keyword, new in z/OS

V1R10, is set to NOHONORIEFUSIREG 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
- 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 DFSMSshm 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 DFSMSshm address space, enabling more tasks to be started within the DFSMSshm address space. When DFSMSshm invokes DFSMSdss through the DFSMSdss cross-memory application interface, DFSMSshm requests that DFSMSdss use a unique address space identifier for each unique DFSMSshm function and host ID. The address space identifier for each function is in the form ARCnXXXX, where *n* is a unique DFSMSshm host ID and XXXX is an abbreviation of a DFSMSshm 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 DFSMSshm host ID 1 would result in a generated address space identifier of ARC1MIGR. The address space terminates automatically when DFSMSshm 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 you 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 Part 2

## 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 IW MCPAFN service. Change programs to not require a specific processor (that is, remove use of IW MCPAFN).
  - 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 3<sup>rd</sup> party alternative serialization products will not cause problems, because all related ENQs are issued with RNL=NO to prevent 3<sup>rd</sup> party management.
  - If you will be affected by the new enqueues, uniquely name the consoles and console logon IDs within the GRS ring.

10

© 2008 IBM Corporation

z/OS 1.10: Migration Part 2

## 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.


11

© 2008 IBM Corporation



z/OS 1.10: Migration Part 2

## 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.

12 © 2008 IBM Corporation

z/OS 1.10: Migration Part 2

## 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.

13 © 2008 IBM Corporation

z/OS 1.10: Migration Part 2

**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.**

14

© 2008 IBM Corporation

z/OS 1.10: Migration Part 2

**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.**

15

© 2008 IBM Corporation

z/OS 1.10: Migration Part 2


## 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.

16
© 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.
- 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 MGCRE 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 IEE871I 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 SCSDEMSEL 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\_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.



### **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.

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 IEA3811 are issued during system initialization. (IEA3811 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](mailto: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 DCCDDEVE entry in combination to correctly identify the device. The DCCDSDEV entries are the same as the DCCDDEV entries but they have a DCCDEYYP=DCCDDEVE 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 UCBCAN 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:

IWMSVPOL 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 Part 2



## XL C/C++ Migration Actions for z/OS R10 from z/OS R8

### Migration Actions You Can Do NOW :

- Remove your *run-time* dependency on the C/C++ IBM Open Class Library (**Required-IF, as of R9**)
  - *Development* support was removed in z/OS R5, which included the removal of the Application Support Class and Collection Class libraries. Since then, only the run-time support had been provided for the C/C++ IBM Open Class Library.
  - z/OS R8 was the last release that include that *run-time* support. If you have applications that depend on this support, they will not run after the run-time support is removed (as of z/OS R9).
  - Stop using the DLLs, and side deck libraries mentioned in your handout. Detection can be done with the IBM Open Class Library Detection tool, *iocchk*.
  - Use the C++ Standard Library (shipped with Language Environment) instead.

17
© 2008 IBM Corporation

### XL C/C++ 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*.

### XL C/C++ Migration Actions You Can Do Now

#### **Review the XL C/C++ Migration Guide for the Application Programmer (Recommended)**

*Not required, but recommended if you use XL C/C++.*

The publication *z/OS XL C/C++ Compiler and Run-Time Migration Guide for the Application Programmer* is written for application programmers, whereas *z/OS Migration* is written for system programmers. However, in some customer locations, job scope could overlap such that system programmers might find information in the XL C/C++ publication that is relevant to their responsibilities. For example, migration information related to the c89 utility in the XL C/C++ publication could be of interest. Therefore, you ought to review this XL C/C++ publication if you use XL C/C++.



#### **Remove your run-time dependency on the C/C++ IBM Open Class Library (Required-IF, as of R9)**

*Required if you are using the run-time support provided by the C/C++ IBM Open Class Library.*

In z/OS V1R5, development support for the C/C++ IBM Open Class Library, specifically, the Application Support Class and Collection Class libraries, was removed. From z/OS V1R5 to z/OS V1R8, only run-time support was provided for the C/C++ IBM Open Class Library. Now, as of z/OS V1R9, the run-time support is also removed. If you have applications that depend on the C/C++ IBM Open Class Library run-time support, they will not run on z/OS V1R9 or higher.

#### **Migration action:**

Stop using the following dynamic link libraries (DLLs), which provide the run-time support:

- Member CLB3CLAS (aliases APPSUPP and ASCCOLL) in data set CBC.SCLBDL
- Members CLBECOLL (alias COLL) and CLBEIOC (alias IOC) in data set CBC.SCLBDLL2

**Tip:** Use the IBM Open Class Detection Tool (*iocchk*) to see if an input file makes any references to IBM Open Class Library DLLs. To download the tool, go to <http://www-306.ibm.com/software/awdtools/czos/support/>.



Update any programs products that are affected by this removal. Review the target system requirements found in the *z/OS Planning for Installation* book, and in PSP buckets, for IBM products. Investigate ISV products to ensure that they do not rely upon the IBM Open Class Library run-time support.


If you are using the CBC.SCLBSID library, which contains side-decks, ensure that you are using only the IOSTREAM, IOSX64 (64-bit IOSTREAM), and COMPLEX members, as they are the only members that remain now that the C/C++ IBM Open Class Library run-time support has been removed.

For C/C++ run-time library support, use the C++ Standard Library, shipped with base element Language Environment, instead of the C/C++ IBM Open Class Library. Applications that are not changed to use the C++ Standard Library might abend with a message such as *CEE3501S The module ASCCOLL was not found*.





**Note:** An “as-is” alternative is available to allow you to continue using the C/C++ IBM Open Class Library DLLs. The same DLLs that were in z/OS V1R8 are available for download from the C/C++ support Web site. There is no charge but the DLLs are provided without warranty. To download them, go to <http://www-306.ibm.com/software/awdtools/czos/support/>.

z/OS 1.10: Migration Part 2

## 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).

18 © 2008 IBM Corporation

z/OS 1.10: Migration Part 2

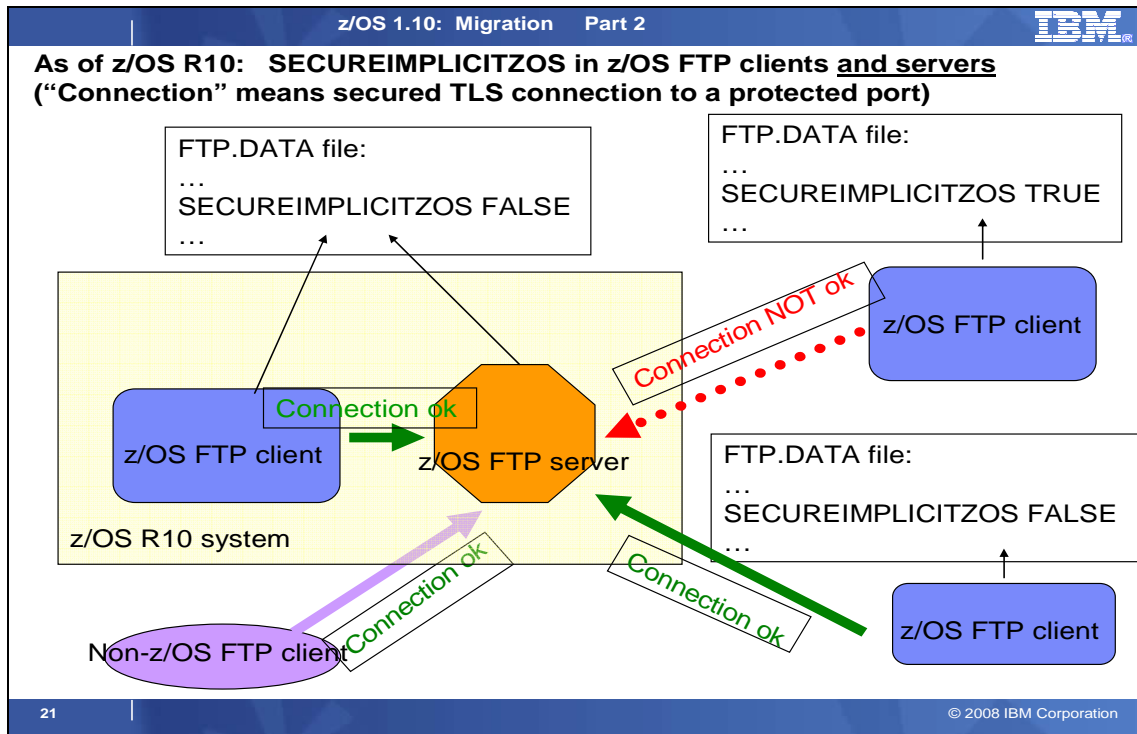
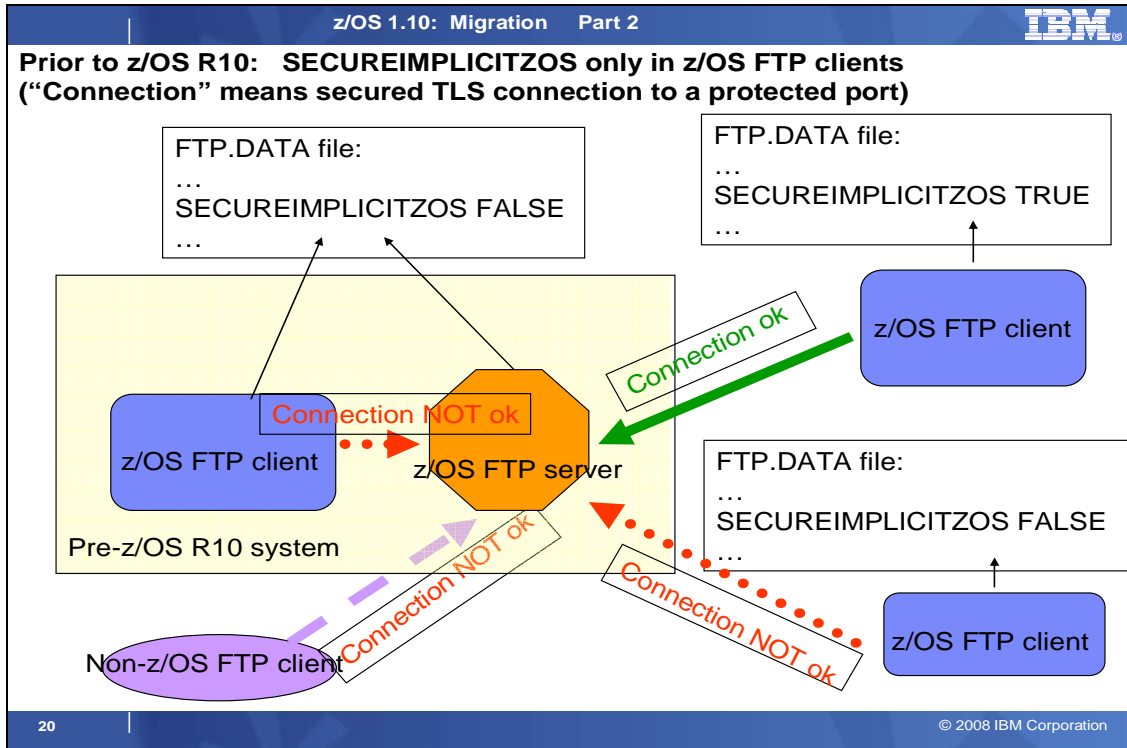
## Communications Server Migration Actions for z/OS R10 from z/OS R8



**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!**
    - oSame value for successful connection.
    - o**Non-z/OS FTP clients need SECUREIMPLICIT FALSE.**

19 © 2008 IBM Corporation





z/OS 1.10: Migration Part 2

## 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.

22

© 2008 IBM Corporation


z/OS 1.10: Migration Part 2

## 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 tha this is not the z/OS BCP APPC support!

23

© 2008 IBM Corporation

z/OS 1.10: Migration Part 2

## 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.

24
© 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.
- 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.*

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 TcplImage 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 TcplImage 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.

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 ITEXCGR. 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 ftpdply.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 Part 2

### Various Migration Actions for z/OS R10 from z/OS R8





- 
 • ICSF HCR7750: Stop using RSA private keys on a PCICC or PCIXCC/CEX2C cryptographic card (Recommended)
  - PCICC or PCIXCC/CEX2C card might possess the only copy of a retained RSA private key. As of HCR7750, cannot store new private RSA keys intended for key management usage in a cryptographic coprocessor.
- 
 • ICSF HCR7750: Increase the size of your PKDS (Required, as of HCR7750)
  - 
 • As of HCR7750, you must allocate your PKDS with a larger record size. Failure to do this before starting ICSF will result in initialization errors.
- ISPF: Modify ds allocation exit routines to accommodate z/OS UNIX files (Required-IF, as of R9)
  - As of R9, ISPF can edit, view, and browse z/OS UNIX files (as well as MVS ds). This affects existing ds allocation exit routines. They now need to be aware that they might receive a pointer to an SVC 99 parm list containing new keys used to allocate a z/OS UNIX file to a DD.

25 © 2008 IBM Corporation


z/OS 1.10: Migration Part 2

### Various Migration Actions for z/OS R10 from z/OS R8



- HCD: Set up the IBM TDS LDAP server for calling the HCD LDAP backend (Required-IF, as of R10)
  - HCD LDAP Backend must now be defined as a plug-in to the IBM TDS LDAP server. Applications programs that use the LDAP API do not need to be changed.
- ICKDSF: Use the new default for the INDEX parameter (Recommended, as of PK56092)
  - Default change on size of VTOC index when the INDEX is defaulted on the INIT command.
  - Now, the default size of the index is based on the required index records for a given volume size and the size of the VTOC .
- 
 • Integrated Security Server LDAP: Migration from LDAP Server to IBM TDS (Recommended)
  - R10 is planned to be the last release to support LDAP Server. Use IBM TDS instead (initially shipped in R8 PTF, and integrated in R9)

26 © 2008 IBM Corporation

z/OS 1.10: Migration Part 2


### Various Migration Actions for z/O S R10 from z/O S R8

- **Security Server RACF: Update database templates (Required)**
  - If you have the PTF for OA22588 installed, then no template update is necessary.
  - Message may be received to say template update isn't required.
  
- **TSO/E: Accommodate TSO/E default change for the compiler programming table (Required-IF, as of R9)**
  - IRXCMPTM default now contains values for the Alternate Library for REXX.
  - You may be able to eliminate a usermod to IRXCMPTM, or putting SEAGALT in the lpa list, if the new defaults are acceptable to you.
  - (If you use the REXX Library for z/OS product, you still must put REXX.SEAGLPA which is provided in that product in the lpa list.)

27
© 2008 IBM Corporation

### Various 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*.

### Migration Actions You Can Do Now



#### **ICSF: Stop using RSA private keys on a PCICC or PCIXX/CEX2C cryptographic card (Recommended, as of HCR7750 and integrated into R10)**

*Recommended, not required to avoid loss of keys upon a hardware failure.*

A PCICC or PCIXCC/CEX2C card might possess the only copy of a retained RSA private key. If you have applications or middleware that use the retained key functionality of these cards, loss of keys could occur if the hardware fails. The hardware failure could be caused by something as simple as an exhausted or malfunctioning card battery. Lost retained keys have the further implication of lost data for retained key management keys, and an inability to verify signatures for retained signature keys.

Starting with the *Cryptographic Support for z/OS V1R7-V1R9 and z/OS.e V1R7-V1R8* Web deliverable (ICSF FMID HCR7750), you no longer have the ability to store new private RSA keys intended for key management usage in a cryptographic coprocessor. Existing applications will continue to be able to use the retained keys and to delete them from the cryptographic coprocessor cards. However, if you have any RSA private keys on a PCICC or PCIXCC/CEX2C cryptographic card, it is recommended that you migrate them to an alternate key strategy.

**Note:** The storing of private RSA keys intended for signature usage in a cryptographic coprocessor continues to be supported when the modulus length is 2048 bits or less.

**Migration action:** If you have any RSA private keys on a PCICC or PCIXCC/CEX2C cryptographic card, migrate them to an alternate key strategy.

**Tip:** Use IBM Health Checker for z/OS to detect the existence of retained RSA private keys on a PCICC or PCIXCC/CEX2C cryptographic card. The check is named ICSFMIG7731\_ICSF\_RETAINED\_RSAKEY.

### Migration Actions Pre-First IPL





### **ICSF: Increase the size of your PKDS (Required, as of HCR7750 and integrated into**

#### **R10)**

With ICSF FMID HCR7750, the record size of the PKA key data set (PKDS) has increased. You must allocate a new PKDS with a larger record size so that internal I/O routines can handle 4096-bit tokens. Failure to migrate the PKDS before starting ICSF results in error messages CFC0286, CSFM406A, and CSFM407A (and ICSF not initializing.)

If the PKDS is shared with earlier systems, install the PTF for APAR OA21807 on those systems to allow continued sharing of the PKDS. ICSF will be able to interact with both sizes of the PKDS.

**Migration action:** Before initializing ICSF, do the following:

1. Make a backup copy of the PKDS you are currently using.
2. Create a larger PKDS. Use the JCL in SYS1.SAMPLIB(CSFPKDS). If the PKDS will be shared, place the VSAM data set where it can be shared.
3. Copy the old PKDS to the larger PKDS using the JCL in SYS1.SAMPLIB(CSFPKDCP). Protect the VSAM data sets from viewing by unauthorized personnel.
4. Initialize the larger PKDS and update the ICSF started procedures on all systems to reference the new PKDS.
5. Stop and restart ICSF on each system.

**Tip:** Use IBM Health Checker for z/OS to verify that the PKDS size in an ICSF pre-HCR7750 environment is sufficiently allocated to support 4096-bit RSA keys. The check is named ICSFMIG7731\_ICSF\_PKDS\_TO\_4096BIT.

### **HCD: Set up the IBM TDS LDAP server for calling the HCD LDAP backend (Required-IF, as of R10)**

*Required if you want to continue to use the LDAP API to search or update IODFs.*

Before z/OS V1R10, HCD used a nonstandardized LDAP Backend interface to the LDAP server to process user requests against IODF configuration data. This interface is no longer supported with z/OS V1R10. Instead, the HCD LDAP Backend must now be defined as a plug-in to the IBM TDS LDAP server.

**Migration action:**

- Update the LDAP server configuration file ds.conf. A sample of how to define the HCD LDAP Backend as a z/OS LDAP server plug-in in the server configuration file is delivered with HCD and is installed in /usr/lpp/hcd/examples/ds.conf. Also, see the topic “How to provide LDAP support for HCD” in *z/OS HCD User’s Guide* for more information about how to do this.
- Integrate the HCD schema file schema.hcd.ldif, distributed with HCD, into the LDAP server by using the following command: `ldapmodify -h ldaphost -p ldapport -D adminDN -w passwd -f /usr/lpp/hcd/etc/schema.hcd.ldif`

**Note:** To issue the ldapmodify command, a user must have administrator authority for the IBM TDS LDAP server. Application programs that use the LDAP API do not need to be changed.

### **ICKDSF: Use the new default for the INDEX parameter (Recommended, as of PK56092)**

*Recommended, not required if you install the PTF for APAR PK56092 because if you use the new default, the index size is built with a value that is based on the size of the VTOC. Therefore, you do not have to calculate the size needed for the index, as it is done automatically.*

APAR PK56092 for ICKDSF R17 (which is the ICKDSF level in z/OS V1R10 down to V1R4) provides support for the extended address volume (EAV) function in z/OS V1R10. As part of this support, there is a change to the default size of a VTOC index when the INDEX parameter of the ICKDSF INIT command is defaulted. This default change occurs after you install the PTF for the APAR, regardless of whether you exploit EAV function.

Without the APAR, when the INDEX parameter is defaulted, the index is generated starting at the track following the end of the VTOC, with a size equal to the number of tracks per cylinder. However, if you specify VTOC(END), the default index is located in the previous cylinder that precedes the VTOC and uses the whole cylinder.

With the APAR, when the INDEX parameter is defaulted, the index is generated starting at the track following the end of the VTOC. However, if you specify VTOC(END), the default index is located in the cylinder that precedes the VTOC. In both instances, the default size of the index is based on the required index records for a given volume size and the size of the VTOC.

**Migration action:** To use the new default, install the PTF for APAR PK56092 and then do not specify the INDEX parameter with the ICKDSF INIT command.



**TSO/E: Accommodate the TSO/E default change for the compiler programming table (Required-IF, as of R9)**

*Required if you modify the TSO/E-provided IRXCMPTM or provide an override to TSO/E's SYS1.LINKLIB(IRXCMPTM).*

When TSO/E REXX initializes the first language processor environment in the address space, TSO/E REXX loads module IRXCMPTM as the compiler programming table. After the compiler programming table is loaded, it is used for all compiled execs in the current and any subsequent language processor environments.

Before z/OS V1R9, the IRXCMPTM module that TSO/E provided in LINKLIB contained no entries, which indicated to TSO/E that a compiler runtime processor was not installed. It was the responsibility of the system programmer to provide a compiler programming table for TSO/E to use for compiled execs, if desired. The system programmer most likely used a TSO/E-provided sample usermod, or overrode SYS1.LINKLIB(IRXCMPTM) with an LPA list specification of IRXCMPTM from either REXX.SEAGALT (from the Alternate Library for REXX) or REXX.SEAGLPA (from the REXX Library for z/OS).

As of z/OS V1R9 and the integration of the Alternate Library for REXX into z/OS V1R9, TSO/E has changed the IRXCMPTM default to specify the Alternate Library for REXX values. Specifically, the following values are the new defaults in the TSO/E-provided IRXCMPTM:

```
FIRST_ENTRY_RTPROC DC C'EAGRTPRC'
FIRST_ENTRY_COMPINIT DC C'EAGRTXIN'
FIRST_ENTRY_COMPTERM DC C'EAGRTXTR'
FIRST_ENTRY_COMPLOAD DC C'EAGRTXLD'
FIRST_ENTRY_COMPVAR DC C'EAGRTXVH'
```

Because of the changed default values, you might have to modify your IRXCMPTM.

**Migration action:**

If the z/OS V1R9 and later TSO/E IRXCMPTM default is acceptable to you:

- You no longer need to modify IRXCMPTM. Do not install any usermods to IRXCMPTM.
- You no longer need to override SYS1.LINKLIB(IRXCMPTM) with an LPA list entry for REXX.SEAGALT (Alternate Library for REXX's load library, which provides a customized IRXCMPTM). Alternate Library for REXX's REXX.SEAGALT(IRXCMPTM) matches the TSO/E-changed default.

**Note:** If you use the REXX Library for z/OS product, you still must specify, in the LPA list, the REXX.SEAGLPA data set provided with that product. REXX.SEAGLPA(IRXCMPTM) values also match the TSO/E changed default.

If the z/OS V1R9 and later default is not acceptable to you, continue to modify IRXCMPTM with your specific values at each release upgrade.

**Migration Actions Post-First IPL****ISPF: Modify data set allocation exit routines to accommodate z/OS UNIX files (Required-IF, as of R9)**

*Required if you have specified any data set allocation exit routines with custom processing for the allocate operation.*

The data set allocation exit point is triggered when ISPF is used for several operations: create, delete, allocate, deallocate, and concatenate. ISPF passes to the exit routine a pointer to an SVC 99 parameter list. This contains a list of parameters (such as data set name and status) that the exit routine can use. Which parameters are in the list depends on which operation is being performed. The exit routine can be coded to do extra processing only for selected operations and to do nothing for others.

In z/OS V1R9, ISPF users will be able to edit, view, and browse z/OS UNIX files as well as MVS data sets. This will affect existing data set allocation exit routines. They now need to be aware that they might receive a pointer to an SVC 99 parameter list containing new keys used to allocate a z/OS UNIX file to a DD.

**Migration action:**

- If you have specified a data set allocation exit routine with custom processing for the allocate operation, you will need to change the exit routine to recognize the new keys in the SVC 99 parameter list. Otherwise, the first time a user tries to edit a z/OS UNIX file, ISPF will call the exit routine and pass a pointer to the parameter list including one of the new keys for allocating z/OS UNIX files to a DD. If the exit routine does not recognize the new key, it might fail or simply return to ISPF.

- If you have not specified a data set allocation exit routine, take no action.
- If you have specified a data set allocation exit routine, but it does not do any allocate processing (for example, if you have only customized the create and concatenate operations), take no action.



**Integrated Security Services LDAP: Migration from LDAP Server to IBM TDS (Recommended)**

*Recommended if you have not already migrated to IBM TDS because the IBM TDS LDAP server is designed to be an improvement over Integrated Security Services LDAP Server and because migration is planned to become a requirement in the release following z/OS R10.*

z/OS V1R10 is planned to be the last release of z/OS that will support the Integrated Security Services LDAP Server. A new optimized LDAP server, a component of IBM Tivoli Directory Server for z/OS (IBM TDS), was delivered by PTF for z/OS V1R8 and is integrated in z/OS V1R9 and later. If you are currently using the Integrated Security Services LDAP Server, you should investigate migrating to IBM TDS, which was designed to allow greater consolidation of LDAP directories on z/OS to help simplify enterprise management and disaster recovery.

IBM TDS is a base element of z/OS and consists of:

- A rewritten LDAP server
- An LDAP client
- LDAP client utilities

**Migration action:** For instructions about migrating from the LDAP server in Integrated Security Services to the LDAP server in IBM TDS, see *IBM Tivoli Directory Server Administration and Use for z/OS*.

**Security Server RACF: Update database templates (Required)**


To ensure that the RACF utilities function properly, use the IRRMIN00 utility to update the test and production RACF databases with the database templates for the current release level.

**Migration action:** To install the database template updates, run the IRRMIN00 utility with PARM=UPDATE.

**Note:** If you are upgrading your database templates from a level where the PTF for APAR OA22588 was installed, expect IRRMIN00 to produce a return code of 4 and message IRR8025 PARM=UPDATE specified, but template update not required. The database templates provided in z/OS V1R10 are at the same level as what is provided in OA22588; therefore, no template updates are necessary.

z/OS 1.10: Migration Part 2

## Important Actions 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)
- mig ✓
 • Update callers of CNZMXURF that do not hold appropriate serialization (Required-IF, as of R10)
- mig ✓
 • Remove your *run-time* dependency on the C/C++ IBM Open Class Library (Required-IF, as of R9)
- mig ✓
 • IP Services: Migrate to the TN3270E Telnet server that runs in its own address space (Required-IF, as of R9)
- mig ✓
 • SNA Services: Ensure compatible levels of VTAM for HPR sessions (Required-IF, as of R10)
- mig ✓
 • ICSF HCR7750: Increase the size of your PKDS (Required, as of HCR7750)
- Integrated Security Server LDAP: Migration from LDAP Server to IBM TDS (Recommended)

**...(and others from Part 3!!)**

28

z/OS 1.10: Migration Part 2

## z/OS 1.10: Migration - Part 2 Summary



- **General Migration Actions for Everyone:**
  - Default MEMLIMIT is now 2. Removed DFSORT libraries. Changed Health Checks. CADS 1 for R10, 1 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 FTPDATA between FTP server and client, Make sure all HPR-capable VTAMs are R8 or higher, More region size for Telnet
- **ICSF Migration Action:** Check for retained RSA private keys on cards.
- **ISPF Migration Action:** Data set allocation exit routines to handle z/OSUNIX enhancement.
- **HCD Migration Action:** Use IBM TDSLDP server for HCD LDAP backend.
- **ICKDSF Migration Action:** New INDEX default size.
- **T SO/E Migration Action:** New IRXCMPTM default.

**Review z/OS 1.10: Migration Part 3!**

29

© 2008 IBM Corporation