Session 7398:

Migrating to z/OS 1.12: Part 2 of 2

Marna WALLE   mwalle@us.ibm.com
z/OS System Build and Install
IBM Poughkeepsie
August 4, 2010

Permission is granted to SHARE to publish this presentation paper

in the SHARE proceedings; IBM retains the right to distribute copies

of this presentation to whomever it chooses.

© 2010 IBM Corporation

Abstract for Session 7398: Migrating to z/OS 1.12: Part 2 of 2
This is part two of a two-part session that will be of interest to System Programmers and their managers who are migrating to z/OS 1.12 from either z/OS 1.10 or z/OS 1.11. It is strongly recommended that you attend both sessions for a complete migration picture.

The general availability date for z/OS V1 R12 is planned for <u>September 24, 2010</u>.

IBM

Trademarks

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

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

*, AS/400®, e business(logo)®, DBE, ESCO, eServer, FICON, IBM®, IBM (logo)®, iSeries®, MVS, OS/390®, pSeries®, RS/6000®, S/30, VM/ESA®, VSE/ESA, WebSphere®, xSeries®, z/OS®, zSeries®, z/VM®, System i, System i5, System p, System p5, System x, System z, System z9®, BladeCenter®

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.
IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.
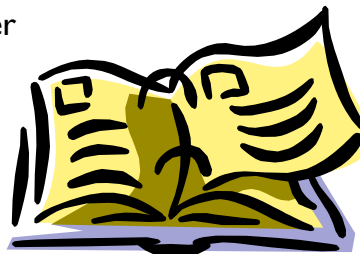
* All other products may be trademarks or registered trademarks of their respective companies.

Notes:
Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections 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 multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.
IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.
All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.
This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.
All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.
Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.
Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

| 2 | Migrating to z/OS 1.12: Part 2 of 2 | © 2010 IBM Corporation |

IBM

Migrating to z/OS 1.12: Part 2 of 2 Agenda

- **Migration Actions for z/OS R11 and R12:**
  - General Migration Actions
  - Element-Specific Migration Actions:
    - BCP
    - Communications Server
    - DFSMS
    - Distributed File Service
    - ICSF, HCM, IBM Tivoli Directory Server
    - Infoprint Server
    - JES2 and JES3
    - Language Environment
    - z/OS UNIX
- **Summary**

| 3 | Migrating to z/OS 1.12: Part 2 of 2 | © 2010 IBM Corporation |

Structure of *z/OS R12 Migration* Presentations

►Session 7399: Part 1 (Planning)

•Hopefully, you attended this presentation. If not, see proceedings.

- •Content of z/OS R12: removed elements and functions
- •Coexistence Requirements – use FIXCAT
- •Driving System Requirements – electronic, DVD, latest binder, …
- •Target System Requirements – IBM (use FIXCAT to help), non-IBM, and HW
- •Using IBM Health Checker for z/OS for migration verification.

Attend both to get a complete migration picture!

(Along with z/OS Migration book!)

►Session 7398: Part 2 (Some Migration Actions)

•Migrations actions for z/OS R12 from selected elements you just saw.

4   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

---

Definition of "Migration Action":

- **Upgrading to a new z/OS release is a two step process:**
  1. **Migration:** the installation of a new version or release of a program to replace an earlier version or release.
  2. **Exploitation:** usage of new enhancements available in the new release. Not covered in this presentation.
- **After a successful migration, the applications and resources on the new system function the same way they did on the old system, if possible.**

- **Migration actions are classified as:**
  - **Required**: required for all users
  - **Required-IF**: only required in certain cases
  - **Recommended**: good to do because it 1) may be required in the future, 2) resolves performance or usability problem 3) improves migration workload.
- **Migration actions are also classified as when they may be performed:**
  - **NOW, Pre-First IPL, or Post-First IPL**

Means "don't overlook!"   **Means some programmatic assistance is available**

5

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 V1R12 level of code to make these changes, and the changes don't require the z/OS V1R12 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 V1R12.
- *Pre-First IPL.* These are migration actions that you perform after you've installed z/OS V1R12 but before the first time you IPL. These actions require the z/OS V1R12 level of code to be installed but don't require it to be active. That is, you need the z/OS V1R12 programs, utilities, and samples in order to perform the migration actions, but the z/OS V1R12 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 V1R12 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 V1R12 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 V1R12. You need a running z/OS V1R12 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 V1R12 to be active to perform the task. You might perform the task quite a while after the first IPL.

Icons used in this presentation:

means that you shouldn't overlook this migration action.

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

Migration Actions for Elements Between z/OS R10 and z/OS R12

When migrating from z/OS R10 to z/OS R12, the specified elements in the slide above have required migration actions. Refer to *z/OS V1R12 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 R11 to R12, there are fewer migration actions that are applicable. Use the customized *z/OS V1R12.0 Migration - From z/OS V1R11.0 to z/OS V1R12.0* book to see just the migration actions appropriate for this path.

There also is a customized book for those migrating from z/OS R10 to z/OS R12 (*z/OS V1R12.0 Migration - From z/OS V1R10.0 to z/OS V1R12.0*), which is almost identical to the standard *z/OS V1R12 Migration* book. The only difference is one migration action that has been omitted from the z/OS R10-z/OS R12 customized book and is found in the standard book. This is because the behavior was in a new function introduced in z/OS R11, and therefore isn't applicable to those migrating from z/OS R10.

The customized migration books can be found at this web site:
http://www.ibm.com/systems/z/os/zos/installation/ .

General Migration Actions for z/OS R12 from z/OS R10

► <u>Migration Actions You Can Do NOW:</u>

- **Install coexistence and fallback PTFs (Required) –** You must use FIXCAT to verify the correct PTFs are installed. Use `IBM.Coexistence.z/OS.V1R12.` You cannot use the EPSPT tool to do this.

- **Add or change volumes to keep file systems in a single data set (Recommended)**
  - z/OS version root file system is large – 3100 cylinders. One 3390-3 is too small.

► <u>Migration Actions Pre-First IPL:</u>

- **Accommodate new address spaces (Recommended)**
  - New in R12: **ARC*n*RSTy** for DFSMShsm for full-volume recovery from dump. (Data set recovery from dump still uses ARC*n*REST.) Perform security customization to allow new started procedures with these names.
  - New in R11 (and on R10 with UA47493): **HWIBCPII** for BCPii. Automatically started, can be manually restarted with a procedure. Will not initialize unless customization is complete.

General Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Remove references to deleted data sets and paths (Required)**
  - Removed in R12:
    - BCP: SYS1.SCUNIMG/ACUNIMG;
      - SYS1.SCUNIMG: Pre-built image CUNIDHC2 for DB2 is removed. "Unicode on Demand" obsoletes need for this pre-built image.
      - Remove SCUNIMG from the linklist and APF authorization list.
    - LE: CEE.SCEEUMAP, and paths /usr/lib/nls/locale/ucmap/IBM and /usr/lib/nls/locale/uconvtable/IBM;
    - all the msys for Setup data sets and paths.
  - Removed in R11: LE's CEE.SCEEUTBL
- **Add references to new data sets and paths (Required)**
  - New in R12: nothing.
  - New in R11: TCPIP.SEZAEXEC (concat to current REXX libraries) , and SYS1.SCCR3BND and SCCR6BND (Metal C Runtime)

General Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Update your check customization for modified IBM Health Checks (Recommended)**
  - Changed in R12:
    - USS_PARMLIB (PARMs on MOUNT are now case sensitive in the check)
  - Changed in R11:
    - CSV_LNKLST_SPACE, HSM_CDSB_BACKUP_COPIES, HSM_CDSB_DASD_BACKUPS, HSM_CDSB_VALID_BACKUPS, USS_PARMLIB, USS_PARMLIB_MOUNTS, and XCF_FDI

- **Accept the CustomPac Installation Dialog default file system of zFS (Recommended)**
  - Default has moved from HFS to zFS. Even in your saved configuration!
  - CHANGE DSNTYPE ZFS HFS dialog command – now and later.

9   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

General Migration Actions Between z/OS V1 R10 and z/OS V1 R12

These migration actions were taken from *z/OS R12 Migration.* Some descriptions and actions have been shortened for inclusion in this presentation. For the complete descriptions and actions, refer to *z/OS R12 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 V1R12 systems during your migration, and allow backout from z/OS V1R12 if necessary. Use the SMP/E V3R5 REPORT MISSINGFIX command in conjunction with the FIXCAT type of HOLDDATA as follows:

1. Acquire and RECEIVE the latest HOLDDATA onto your pre-z/OS V1R12 systems. Use your normal service acquisition portals or download the HOLDDATA directly from http://service.software.ibm.com/holdata/390holddata.html. Ensure you select **Full** from the Download NOW column to receive the FIXCAT HOLDDATA, as the other files do not contain FIXCATs.
2. Run the SMP/E REPORT MISSINGFIX command on your pre-z/OS V1R12 systems and specify a Fix Category (FIXCAT) value of **"IBM.Coexistence.z/OS.V1R12"**. The report will identify any missing coexistence and fallback PTFs for that system. For complete information about the REPORT MISSINGFIX command, see *SMP/E Commands*.
3. Periodically, you might want to acquire the latest HOLDDATA and rerun the REPORT MISSINGFIX command to find out if there are any new coexistence and fallback PTFs.

Note that it is no longer possible to do z/OS coexistence verification with the EPSPT tool.

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 V1R12, the size of the z/OS root file system, whether HFS or zFS, was approximately 3100 3390 cylinders. This is closely 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.

Tips:

- 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 IBM Health Checker for z/OS check CHECK(IBMUSS,ZOSMIGREC_ROOT_FS_SIZE) to determine whether a volume has enough space for the z/OS root file system.
- This capability is also available in z/OS V1R9 with APAR OA28684 (PTF UA49361) and z/OS V1R10 with APAR OA28684 (PTF UA49362).

Verify SMP/E utility parameters to support assembling JES2 and SDSF (Required, as of R11)

The assembler utility parameters must be updated to support assembling of JES2 and SDSF source modules.

Migration action: Ensure you have the following parameters for your assembler utility in the SMP/E OPTION entry that will be used to install and maintain z/OS:

```
REP UTILITY(ASMA90) NAME(ASMA90)
PARM(DECK,NOOBJECT,GOFF,LIST(133),OPTABLE(UNI))
PRINT(ASMPRINT) /* see below */
RC(0).
```

- `DECK, NOOBJECT` must always be specified in the PARM list. For example, `PARM(DECK,NOOBJECT)`.
- As of z/OS V1R11, JES2 modules require `GOFF` and `LIST(133)` to assemble correctly. For example, `PARM(DECK,NOOBJECT,GOFF,LIST(133))`.
- When you specify `LIST(133)`, then you must either:
    - specify a unique output file to direct assembler output, for example, `PRINT(ASMPRINT)`. You must then provide a DDDEF in your target and DLIB zones (or a JCL DD card) for ASMPRINT specifying SYSOUT, or
    - provide a JCL DD card such as `//SYSPRINT DD SYSOUT=*,LRECL=133,RECFM=FBA`.
- SDSF modules require `OPTABLE(UNI)` to assembler correctly. `OPTABLE(ESA)` is not acceptable.

Note: ServerPacs produced at Package Version 22.10.19 and higher have the assembler utility parameters updated in the SMP/E OPTION provided.

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.

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*.

Verify that you have enough XCF groups in your CDS and enough XCF members in your XCF groups (Recommended)

Recommended to ensure you have an adequate number of XCF groups and members formatted in your sysplex couple data sets.

1. Issue the DISPLAY XCF,COUPLE command on your current system. Notice the values of MAXGROUP and PEAK for your sysplex couple data sets. These values show you the maximum number of XCF groups that the couple data sets can support, and the peak number of XCF groups ever in use in the sysplex. Also notice the values of MAXMEMBER and PEAK for your sysplex couple data sets. These values show you the maximum number of members that the couple data set can support in one group, and the greatest number of members ever in use in the largest group in the sysplex.
2. If your peak member value is close to the maximum member value, you might want to reformat your sysplex couple data sets to support a larger maximum number of members to be used by any one group.

Tip: Use IBM Health Checker for z/OS check XCF_SYSPLEX_CDS_CAPACITY, which checks the adequacy of the number of groups, members, and systems for which a sysplex CDS is formatted.

Discontinue use of IBM Virtualization Engine Enterprise Workload Manager for z/OS (Required-IF, as of R11)

Required if you use either the IBM Virtualization Engine Enterprise Workload Manager for z/OS products.
z/OS V1R11 does not support: IBM Virtualization Engine Enterprise Workload Manager for z/OS V1.1.0 (5655-M76) or IBM Virtualization Engine Enterprise Workload Manager for z/OS V2.1.0 (5655-EWM). If you are using either of these products, they will be deleted during installation of z/OS V1R11 or z/OS V1R12.
Migration action: IBM Virtualization Engine Enterprise Workload Manager (EWLM) V2.1 (5733-EWM) has no replacement. IBM is providing a transition for EWLM V2.1 clients to an IBM STG Lab Services-based offering. Contact STG Lab Services (stgls@us.ibm.com) for more information. In addition, IBM Tivoli offers workload automation and monitoring solutions with the Tivoli Workload Automation Family of products:

- Tivoli Dynamic Workload Broker and Tivoli Workload Scheduler
- Tivoli Workload Scheduler
- Tivoli Composite Application Manager for Response Time Tracking

If you want to preserve your existing copy of IBM Virtualization Engine Enterprise Workload Manager for z/OS V1R10 and EWLM V2.1, you can create a back up. However, keep in mind these products are no longer service supported.

Accept the CustomPac Installation Dialog default file system of zFS (Recommended, as of R11)

Recommended if you want to continue to use HFS data sets for your CustomPac order.
For hierarchical files that are switchable between zFS and HFS, the CustomPac Installation Dialog now sets the default to zFS. This is another step in the implementation of zFS as the strategic z/OS file system. Note that all file systems are still delivered as HFSs.
Migration action:

- For any file systems that you would like to be HFS (and not zFS) use the CHANGE DSNTYPE ZFS HFS command within the CustomPac Installation dialog to convert the file systems to HFS. Note that any HFS data sets in your saved configuration will be changed to ZFS as of z/OS V1R11, and if you want them to remain as HFS you must issue the command above.

- Subsequent saved configurations used after z/OS V1R11 will also be affected, and saved HFS file systems will be converted to ZFS unless you change the file system type to HFS.

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 V1R12 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.
  - 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 V1R12 /etc and /var subdirectories.

Copy files from your old system to the z/OS V1R12 /etc and /var subdirectories, and then modify the files as necessary to reflect z/OS V1R12 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 V1R12 elements and features use /etc:

- BCP (Predictive Failure Analysis), as of z/OS V1R12
- CIM
- Communications Server – IP
- Cryptographic Services – PKI Services and System SSL
- DCE Base Services
- DFSMSrmm
- 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 and Network Authentication Service
- Library Server
- z/OS UNIX System Services

The following z/OS V1R12 elements and features use /var:

- Cryptographic Services – OCSF

- DFSMSrmm
- 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 SMFPRM*xx* 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 SMFPRM*xx* is necessary.
- You can specify MEMLIMIT explicitly in JCL to override the system default that was set (or allowed to default) in SMFPRM*xx*.
- You can specify REGION=0M on the job statement in JCL to implicitly set MEMLIMIT to NOLIMIT, which also overrides the system default (from SMFPRM*xx*).
- 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 SMFPRM*xx*. As of z/OS V1R10, keyword HONORIEFUSIREGION | NOHONORIEFUSIREGION is available in SCHED*xx* to identify if the region and MEMLIMIT settings specified through or otherwise affected by the IEFUSI exit are to take effect for a program.

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 SMFPRM*xx*, 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 V1R12, and R11" 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
- 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 R11 and R12.

Add references to new data sets (Required)

Migration action: Add references in the following places for data sets that have been added to z/OS R11:

- Parmlib
- Proclib
- Logon procedures
- Catalogs
- Security definitions, including program control definitions
- DFSMS ACS routines
- Any backup and recovery procedures.

DDDEF	Data set name (high-level qualifiers are defaults)	DLIB or target	To element or feature	When added	Why added
SCCR3BND	SYS1.SCCR3BND	Target	Metal C Runtime Library	z/OS V1R11	For building AMODE 31 Metal C application programs using the program management binder
SCCR6BND	SYS1.SCCR6BND	Target	Metal C Runtime Library	z/OS V1R11	For building AMODE 64 Metal C application programs using the program management binder
SEZAEXEC	TCPIP.SEZAEXEC	Target	Communications Server	z/OS V1R11	Contains REXX samples that customers can concatenate to the current REXX libraries

Of special note is data set:

- **TCPIP.SEZAEXEC** (in z/OS R11, for Communications Server). This data set contains REXX samples that you can concatenate to the current REXX libraries.

No new data sets have been added in z/OS R12.

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.

There is one new address space in z/OS V1R12. ARCnRSTy is the address space identifier for full-volume recovery from dump, where n is the DFSMShsm host ID and y is the instance of the DFSMSdss started task (a number from 1 to 4). Data set recovery from dump will still use ARCnREST. See "DFSMShsm: Configure your security system to permit started procedures using new address space identifier" later in this presentation for more information.

There was one new address space in z/OS V1R11 (and in z/OS V1R10 with PTF UA47493). Its name is HWIBCPII. The Base Control Program Internal Interface (BCPii) function allows authorized applications to query, change, and perform basic operational procedures against the installed System z hardware base. As a base BCP component, the new BCPii address space allows authorized z/OS applications to access the local support element and other support elements in the HMC network to perform HMC-like functions. Connectivity to the Hardware Management Console (HMC) network is done internally, without the requirement of connecting the HMC network to an intranet or the Internet.

The HWIBCPII address space is started automatically during z/OS initialization and can be manually restarted with a procedure. When z/OS is IPLed, HWIBCPII attempts to initialize and connect to the local CPC's support element. If the proper configuration for BCPii has been completed (as documented in *z/OS MVS Programming: Callable Services for High-Level Languages*), the address space will become active and able to accept requests from BCPii applications. However, if basic configuration steps have not been done, the address space will not be able to initialize properly. A few BCPii error messages will appear on the console when these steps have not been taken. If the BCPii address space is not required by the installation, then these error messages can be ignored. These messages include:

```
HWI016I THE BCPII COMMUNICATION RECOVERY ENVIRONMENT IS NOW ESTABLISHED.
HWI007I BCPII IS ATTEMPTING COMMUNICATION WITH THE LOCAL CENTRAL PROCESSOR
COMPLEX (CPC).
HWI014I THE SNMP COMMUNITY NAME FOR BCPII IN THE SECURITY PRODUCT FOR THE
LOCAL CPC IS EITHER NOT DEFINED OR IS INCORRECT.
HWI006I BCPII ADDRESS SPACE HAS ENDED.
```

The MAXUSER value in parmlib member IEASYS*xx* 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.)

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 ICEPRM*xx* 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!
- SDSF (ISFPARMS customization). Use dynamic statements for ISFPARMS to avoid reassembly!

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 V1R11 ServerPac.

Ensure that any required service is installed prior to using the subsystem with the new z/OS V1R11 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 are no new default high-level qualifiers introduced since z/OS R9.
- 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 V1R11, 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 V1R12 are:

- USS_PARMLIB (This check will compare z/OS UNIX System Services current system settings with those specified in the BPXPRMxx parmlib members used during initialization and issue an exception message if a difference is found. For PARM values on the MOUNT statement, the check is case sensitive, detecting and raising an exception for otherwise identical values that are expressed in different cases.)

The checks that were changed by IBM in z/OS V1R11 are:

- CSV_LNKLST_SPACE
- HSM_CDSB_BACKUP_COPIES
- HSM_CDSB_DASD_BACKUPS
- HSM_CDSB_VALID_BACKUPS
- USS_PARMLIB
- USS_PARMLIB_MOUNTS
- XCF_FDI.

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 HZSPRM*xx* parmlib members, for example.
3. Make any further updates for the checks to ensure that they continue to work as intended.

General Migration Actions Post-First IPL

<none>

SHARE
Boston, MA
© 2010 IBM Corporation
Page 14 of 94
August 4, 2010
Session 7398

BCP Migration Actions for z/OS R12 from z/OS R10

Migration Actions You Can Do NOW:

- **Remove SYS1.SVCLIB SYS1.NUCLEUS, and PASSWORD from GRS RNLs (Recommended, as of R11)**
  - •Before R11, the GRS RNLs contained exclusions entries for ENQs issued for these data sets.
  - •As of R11, these entries have been removed by default in PARMLIB's GRSRNL00. If you have RNLs that contain exclusion entries for these data sets, consider removing the exclusion entries. (Which will result in global ENQs on these data sets.)
  - •If GRS ring with unique sysres per system, consider keeping the entries to reduce ENQs on the ring.

Migration Actions Pre-First IPL:

- **Use the new 16M default buffer size (BUFSIZE) for trace options with the CTIGRSxx member (Required IF, as of R12)**
  - •Before R12, IBM-supplied CTIGRS00 parmlib member would result in BUFSIZE of 128K, maximum was 16MB, and the buffer was below the line private storage.
  - •As of R12, CTIGRS00 will now result in BUFSIZE of 16MB, maximum is 2047MB, and is above the bar private storage in its own memory object.
  - •If you specify your own CTIGRSxx member on the CTRACE option in the GRSCNFxx parmlib member, change the BUFSIZE in your CTIGRSxx to 16MB. Or you could just use the IBM-provided CTIGRS00.

10 Migrating to z/OS 1.12: Part 2 of 2 © 2010 IBM Corporation

BCP Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Review the list of WTORs in parmlib member AUTOR00 (Required, as of R12)**
  - •New parmlib member AUTOR00 is shipped in the DDDEF'd PARMLIB. This should be in your parmlib concatenation. AUTOR00 in the parmlib concatenation will result in auto-reply processing being activated.
  - •Examine the AUTOR00 WTOR replies, compare the replies and delay duration to your automation and make any updates into your own AUTORxx parmlib member.
    - •IEASYSxx statement is AUTOR=(xx,00).
    - •Cannot share an IEASYSxx with AUTOR= on pre-R12 systems.
    - •Only using AUTOR00 does not require any IEASYSxx updates, therefore it remains sharable with pre-R12 systems.
  - •You may stop the activation by specifying AUTOR=OFF. Do not remove AUTOR00 from the DDDEF'd PARMLIB as service or new releases may reinstall it.
  - •There are accompanying commands for auto-reply processing, see *MVS Commands*.

- **Reassemble the stand-alone dump program (Required)**
  - •Reassemble the stand-alone dump program. The stand-alone dump JCL is changed in z/OS R12 and on lower releases with OA31077. The one-step JCL is now called "one-stage" JCL. Use the ServerPac provided job (DMPBLD) or use new sample JCL found in *MVS Diagnosis : Tools and Service Aids*.

11 Migrating to z/OS 1.12: Part 2 of 2 © 2010 IBM Corporation

BCP Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Specify valid user exits for the IFASMFDL and IFASMFDP programs (Required-IF, as of R12 and via OA29894 back to R9)**
  - **To allow user exits to be called by SMF dump programs (either IFASMFDL for logstreams or IFASMFDP for data set), they must be pre-defined on keywords in SMFPRMxx.**
    - **Use the SMFDLEXIT keyword for IFASMFDL exits.**
    - **Use the SMFDPEXIT keyword for IFASMFDP exits.**
  - **Failure do this will stop processing, with the following error message:** IFA840I USER EXIT *xxxxxxxx* NOT REGISTERED WITH SYSTEM
- **Make IFASMFDL and IFASMFDP run in an authorized environment (Required-IF, as of R12 and via OA29894 back to R9)**
  - **IFASMFDL and IFASMFDP are required run in an authorized environment. Otherwise, you'll get an S338 then a S330 abend, especially if executed under TSO/E.**
  - **If you run the SMF dump program from JCL, APF authorization is preserved and no action is required.**
  - **For TSO/E, add IFASMFDL and IFASMFDP to the AUTHPGM section of IKJTSOxx.**
  - **If calling IFASMFDL or IFASMFDP from another program that program must be authorized.**

BCP Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Ensure your ESQA specification in IEASYSxx is adequate (Req-IF, as of R11)**
  - **As of R11, expect an additional 1608 (x'648') byte increase per address space in 31- bit ESQA.**
  - **Determine the number of address spaces that you have on the system. (The additional ESQA storage is allocated at address space create time and freed at address space termination time. Therefore, you might only need to have additional ESQA for the typical peak number of active address spaces.)**
  - **Multiply the number of address spaces by 1608 bytes to get the total additional ESQA storage required.**
  - **Ensure that your SQA= statement in IEASYSxx specifies an adequate amount of additional ESQA storage, as required.**
    - **Use the VSM_SQA_THRESHOLD Health Check from the prior release to examine the ESQA in use percent and high water mark percent.**

SHARE
Boston, MA
© 2010 IBM Corporation

August 4, 2010

Page 16 of 94

Session 7398

BCP Migration Actions for z/OS R12 from z/OS R10

Migration Actions You Can Do Now:

- **Prepare for zAAP-eligible work to run on zIIPs (Required-IF, as of R11 (enabled) and back to R9 with OA27495 (disabled))**
  - As of R11, zAAP-eligible work will run on zIIP when there are no zAAPs on the server and there are zIIPs on the server and defined to the LPAR.
  - Ensure sufficient zIIP capacity to contain this work without impacting workload goals. Use PROJECTCPU. You can disable this function on R11.

Migration Actions Pre-First IPL:

- **Change default locations for LCCA or PCCA control blocks to retain 24-bit virtual storage location (Required-IF, as of R12)**
  - Before R12: if not specified in DIAGxx, default was in 24-bit virtual storage.
  - As of R12: if not specified, default is now 31-bit virtual storage. For only one standard CP online during IPL, then the LCCA and PCCA of the IPL CPU will be 24-bit storage regardless of specification.
- **Accommodate new REUSASID default (Required-IF, as of R12)**
  - In R9-R11: DIAGxx's REUSASID default was NO
  - As of R12: REUSASID is YES.
  - Verify programs can tolerate reusable ASIDs, and have no 0D3 abends.

14   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

---

BCP Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Modify SMF exit routines IEFACTRT because of lengthened fields (Required-IF, as of R11)**
  - The IBM sample IEFACTRT has lengthened some output fields to accommodate long-running job steps.
  - You may want to either rebase on the IBM sample, or verify in your own exit that output fields are long enough.
  - The field names are: TCB (9 bytes), SRB (9 bytes), SERV (13 bytes)

- **Update automation for some deleted SMF messages for step end and job end (Required-IF, as of R12)**
  - Messages IEF374I and IEF376I (single line messages) are replaced by messages IEF032I and IEF033I (multi-line messages), respectively.
    - IEF374I and IEF376I display CPU/SRB times in minutes and seconds, both allow room in the message for up to 99999 minutes. This is not sufficient for long running jobs.
    - IEF032I and IEF033I relieves that constraint by displaying the CPU/SRB times in hours, minutes and seconds, allowing for up to 99999 hours.

15   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

BCP Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Specify install script parameter when using Predictive Failure Analysis (PFA) (Required-IF, as of R12)**
  - **Run the installation script, AIRSHREP.sh, with either *migrate* or *new* as an option.**
    - ***migrate* (recommended)*: preserves PFA history data from prior releases and creates new directories for checks previously installed on your system**
    - ***new:* for first time PFA users, or to start PFA with new directories for all checks.**
  - **/etc/PFA/ini (new usage of /etc!): contains consolidated path information, for all checks. PFA can still use individual ini files from each check directory, if you desire.**
  - **Fallback considerations, due to directory structure changes. Read *z/OS Problem Management.***
  - **Failure to run the script, will result in new z/OS R12 PFA checks not running:**
    - **AIR010I PFA CHECK Check_name WAS UNABLE TO OPEN LOG FILE, ERRNO= *errno* ERRNOJR= *errnojr* FILENAME= *path_name***

16   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

BCP Migration Actions Between z/OS V1 R10 and z/OS V1 R12

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. Starting with z/OS V1R12, stand-alone dump data sets can be placed in track-managed space as well as cylinder-managed space on Extended Address Volumes (EAV).
- Use a large CISIZE and striping for IPCS dump directories, and use 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.

Tips: Control interval sizes less than 24K have been shown to be more vulnerable to fragmentation when used as IPCS dump directories, and IPCS performance can be degraded when such fragmentation occurs. In this

background, warning message BLS21110I will be issued and you might recreate the DDIR by using the CLIST BLSCDDIR. BLS21110I CISIZE(cisize) is less than 24K. It may degrade IPCS performance

Prepare for zAAP-eligible work to run on zIIPs (Req-IF, as of R11 and back to z/OS R9 with OA27495)
Required if using zIIPs with no zAAPs.
In z/OS V1R11, System z Application Assist Processor (zAAP)-eligible work running in z/OS LPARs can run on System z Integrated Information Processors (zIIP) if no zAAPs are installed on the server, and zIIPs are installed on the server and defined for the LPAR.
Migration action:
- For z/OS LPARs on servers that have zIIPs installed with no zAAPs installed, determine:
  - If you have zAAP-eligible work that will now be eligible to run on zIIPs.
  - If there is sufficient zIIP capacity to contain this work without impacting workload goals.
  - If the general-purpose (CP) capacity should be adjusted for any affected LPARs as a result of any additional projected offload.
- For z/VM LPARs where z/OS is running as a guest, determine:
  - If you have zAAP-eligible work that will now be eligible to run on real or virtual zIIPs defined to the guest.
  - If there is sufficient real zIIP capacity to contain this work without impacting workload goals.
  - If the general-purpose (CP) capacity should be adjusted for any affected LPARs as a result of any additional projected offload.

You can use PROJECTCPU in IEAOPTxx on your current release to determine how much zAAP-eligible work is running in each LPAR, and RMF (or equivalent) reports, to determine how busy your zIIPs are. If you want to disable zAAP-eligible work from running on your zIIPs, you can set the new zAAPzIIP parameter in IEASYS*xx* to NO.
Note: This support is available on z/OS V1R9 and z/OS V1R10 with APAR OA27495. The behavior on z/OS V1R9 and z/OS V1R10, however, is not enabled by default. It is enabled by default on z/OS V1R11. If you wish to enable this support on z/OS V1R9 or z/OS V1R10, use the ZZ parameter in IEASYS*xx*. The ZZ parameter is also accepted on z/OS V1R11 where ZZ is a synonym for the ZAAPZIIP parameter.

Ensure sufficient real storage to accommodate increase in hash table size (Required, as of R11)
XES locking and serialized list, along with their related XES signaling support, use internal hash tables for anchoring local, global, and message control blocks. These hash tables reside in the local and global data spaces that are obtained for serialized connections (lock and serialized list structures).
Before z/OS V1R11, the existing hash tables had 64KB (local), 4KB (global), and 1KB (message) hash table entries. This could lead to scalability problems as the number of locking and signaling resources in use in these data spaces become more and more densely populated with control blocks causing the number of "hash synonyms" to also go up. Hash class contention and processing associated with longer queues of control blocks anchored in the hash table entries lead to the scalability problems.
With z/OS V1R11, hash table sizes are increased to 256KB (local), 16KB (global), and 16KB (message) hash table entries to avoid the scalability problems experienced in earlier systems.
Migration action: Ensure sufficient real storage is defined on your z/OS V1R11 system to handle the increased real storage usage of DREF data spaces associated with connectors to serialized structures. The worst case increase in real storage usage
per system might be determined as:
(Number connectors per system to lock structure * 52M)
+ (Number connectors per system to serialized list structure * 4M)
+ (Number connectors per system to lock structure supporting user-managed rebuilds * 52M)
+ (Number connectors per system to serialized list structure supporting user-managed rebuilds * 4M)

Use DISPLAY XCF, specifying a structure name and connector name, to determine the type of serialized connector (for example, LOCK or SERIALIZED LIST), whether or not user-managed rebuilds are supported (ALLOW REBUILD = YES) by them, and which system they reside on.

Stop AXR*nn* before JES shutdown (Required-IF, as of R11 and R10 with APAR OA26802)
Required, if the AXR*nn* address spaces (TSOServer) are started.
Before z/OS V1R11 (without the PTF for APAR OA26802 installed), the AXR*nn* address spaces (TSO Server) were started under the MASTER subsystem. Beginning with z/OS V1R11 (and in z/OS V1R10 with the PTF for APAR OA26802 installed), the AXR*nn* address spaces are started under the primary subsystem allowing System REXX to

use JES services. Therefore, you must terminate the AXR*nn* address spaces before you shut down JES2. You do not have to stop the AXR address space; this action only affects the secondary AXR*nn* address spaces.
Migration action: When you need to stop JES:

- Shut down all AXR*nn* address spaces using one of the following methods.
    - o Issue D JOBS,AXR* to display the AXR*nn* address spaces that are active and selectively cancel each AXR*nn* address space. This method is the least disruptive.
    - o Use the FORCE AXR,ARM command. This command terminates both AXR and all the AXR*nn* address spaces. Note that using this command will terminate AXR as well, which runs under the MASTER subsystem, and might not need to be terminated.

Remove SYS1.SVCLIB, SYS1.NUCLEUS, and PASSWORD from global resource serialization RNLs (Recommended, as of R11)

Not required, but but recommended because by using the z/OS V1R11 default your system will have global ENQs on data sets SYS1.SVCLIB, SYS1.NUCLEUS, or PASSWORD. The change can be made on your z/OS V1R10 system before installing z/OS V1R11, or the change can be made on your z/OS V1R11 system after installing that release.

Before z/OS V1R11, the global resource serialization default resource name lists (RNLs) contained exclusion entries for ENQs issued for data sets SYS1.SVCLIB, SYS1.NUCLEUS, and PASSWORD. With z/OS V1R11, these exclusion entries have been removed from PARMLIB's GRSRNL00. If you have RNLs that contain exclusion entries for any of the three data sets, you should consider removing the exclusion entries.

The requirement for SYS1.SVCLIB and SYS1.NUCLEUS to be in the exclusion RNL became obsolete long before z/OS V1R9. The change can be made on any supported z/OS release at any time.

The PASSWORD data set contains the password for each data set that is password protected. It is strongly recommended to use SAF to invoke a security product such as RACF to protect data sets, rather than using password protection. If you are not password protecting data sets, PASSWORD can also be removed from the exclusion RNL on any supported z/OS release at any time.

Migration action:

- If you have a modified version of the RNLs that still contains SYSDSN PASSWORD, SYSDSN SYS1.NUCLEUS, or SYSDSN SYS1.SVCLIB, consider removing these entries from the exclusion list to ensure that the data sets are properly serialized. But note that if your system is running in ring mode and it has a unique SYSRES volume per system, consider keeping the entries to reduce ENQs going around the ring.
- If you are using password protection (PASSWORD), use system authorization facility (SAF) instead.

Update procedures for vary systems out of the sysplex (Required-IF, as of R11)

Required if your local procedures depend on explicit reaction to message IXC102A and the installation is not running z/OS as a VM guest.

You can use the VARY XCF,sysname,OFFLINE command to vary a system out of the sysplex. Before a system can be removed from the sysplex, it must be reset or isolated from the channel subsystem to ensure that it is no longer performing I/O against shared sysplex resources such as data sets or coupling facility structures.

Before z/OS V1R11, the VARY XCF process would only initiate automatic isolation if a Sysplex Failure Management (SFM) policy were in effect. If no SFM policy were active, a system reset would be required. The system would prompt the operator with message IXC102A to determine when the reset had been performed.

Beginning with z/OS V1R11, VARY XCF always initiates automatic isolation when physically feasible and when z/OS is not running as a VM guest, regardless of whether there is an active (SFM) policy. The system issues message IXC102A only if automatic isolation is not feasible or if no surviving system is able to complete isolation successfully. This change does not depend on the availability of the BCPii or system status detection functions.

Migration action: Update local procedures for removing systems from the sysplex to reflect the fact that message IXC102A may no longer be issued.

Use the new default for the system response to a status update missing condition (Recommended, as of R11)

Not required, but recommended because specifying or defaulting to ISOLATETIME(0) allows SFM to fence and partition a failed system without operator intervention and without undue delay.

You can use the sysplex failure management (SFM) policy to specify how a system is to respond to a status update missing condition. The system default is used when no action is specified for the system or when no SFM policy is active.

Before z/OS V1R11, the system default action was PROMPT, which prompted the operator when a system entered a status update missing condition. As of z/OS V1R11, the system default action has changed to ISOLATETIME(0),

which allows other systems to take immediate action to isolate the failed system. If a system enters a status update missing condition and there is no active SFM policy, the monitoring system will take the system default action:

- If both systems are at or above z/OS V1R11, and the failing system was able to identify itself to the sysplex for automatic isolation (either to a CF for fencing or through XCF System Status Detection), ISOLATETIME(0) is used as the default action against the failing system.
- If either system is prior to z/OS V1R11, and the failing system was able to identify itself to the sysplex for automatic isolation (either to a CF for fencing or through XCF System Status Detection), PROMPT is used as the default action against the failing system.

The DISPLAY XCF,C command shows the SFM action the system expects, though the monitoring system might use a different default action if no SFM policy is defined.

Migration action:
- To obtain the new behavior, which is recommended, specify or default to ISOLATETIME(0) to allow SFM to fence and partition a failed system without operator intervention and without undue delay.
- To obtain the pre-z/OS V1R11 behavior, set up an SFM policy that specifies PROMPT for systems that should not be isolated automatically upon status update missing.

Tip: Use IBM Health Checker for z/OS check XCF_SFM_ACTIVE. This check reports whether SFM is active (and if it is active, whether the recommended settings are in use for the SFM policy). If the check indicates that SFM is not active, this migration action is applicable to the system on which the check ran, meaning that when this system enters a status update missing condition, the changed default from PROMPT to ISOLATETIME(0) might be in effect.

Enable ENF 51 listeners to listen to conditional ENQ contention (Req-IF, as of R11)
Required if listeners default to no qualifier at registration.
Before z/OS V1R11, ENF 51 only allowed listeners to be notified of unconditional contention. With z/OS V1R11, previously unused byte 3 of the ENF 51 qualifier is used. Now, ENF 51 listeners that default to no qualifier at registration can be driven for conditional contention events. Listeners that specify the QUAL parameter with the QMASK parameter including the BYTE3 value on an ENFREQ ACTION=LISTEN macro should not be affected by the change. However, if listeners do not specify QUAL, or specify a QMASK without BYTE3, they will receive conditional contention events and might misinterpret them as traditional unconditional contention events.

Migration action: Do one of the following:
- If the QUAL parameter is specified on an ENFREQ ACTION=LISTEN macro with a QMASK parameter that includes the BYTE3 value, there is no incompatibility and no action is required.
- If QUAL is not specified or QMASK is specified without BYTE3, include BYTE3. (The QUAL specification should not need to be changed, assuming its value for the third byte is 0.)
- Listeners who are interested in getting control for the new "waitless" conditional contention events could consider excluding BYTE3 on the QMASK or using one of the newer QUAL constants defined in data area ISGE51CN.

Discontinue use of IPCS problem management subcommands (Required-IF, as of R11)
Required if you currently use the IPCS problem management subcommands to report and track problems IBM has removed.
Interactive problem control system (IPCS) problem management subcommands were functionally stabilized in 1981 but left in IPCS. Since then, customers have been advised to migrate to other problem management tools. z/OS V1R10 was the last release that included the subcommands. As of z/OS V1R11, the subcommands have been removed from z/OS. The subcommands are:
- ADDDSN — add a data set name to a problem
- ADDPROB — add a problem to IPCS
- DELDSN — delete a data set from a problem
- DELPROB — delete a problem from IPCS
- LISTDSN — list data set attributes
- LISTPROB — list problems
- MODDSN — modify data set attributes
- MODPROB — modify problem attributes

Note that IBM plans to continue to enhance the dump and trace analysis and display facilities of IPCS.
Migration action: Do not use the IPCS problem management subcommands anymore. Use other problem management tools to report and track problems. For example, consider migrating to IBM Tivoli Information Management for z/OS V7 (5698-A08).

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*.

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*.
Note: When the IPLTXTEXIST parameter (which was introduced by ICKDSF R17 APAR PK16403) is specified with the REFORMAT command using the IPLDD parameter, WTOR message ICK21836D is suppressed if IPL text already exists

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 (DMPBLD). 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.
The stand-alone dump one-step (now called one-stage) JCL is changed in z/OS V1R12, and in prior levels with APAR OA31077. If using one-stage JCL to create your stand-alone dump program, ensure that appropriate changes are made to your one-stage JCL. Sample JCL can be found in *z/OS MVS Diagnosis: Tools and Service Aids*.

Use the new 16M default buffer size for trace options with the CTIGRSxx member (Required-IF, as of R12)
Required if you modified any IBM-supplied procedures.
Before z/OS V1R12, the default buffer value (BUFSIZE) for the trace option with the GRS component in the IBM-supplied CTIGRS00 parmlib member was 128K, the maximum value was 16MB, and the buffer was in GRS below the bar private storage. Starting with z/OS V1R12, the default size used by CTIGRS00 is increased to 16MB, the maximum is 2047MB, and the buffer is in GRS above the bar private storage in its own memory object. If you specify your own CTIGRS*xx* member on the CTRACE option in GRSCNF*xx* parmlib member, change the BUFSIZE in CTIGRS*xx* to 16MB.
Migration action: Change the trace option BUFSIZE in your CTIGRS*xx* parmlib member to 16M or make the new default value (16M) active.

Specify valid user exits for the IFASMFDL and IFASMFDP Programs (Required-IF, as of z/OS V1R12, and z/OS V1R11, z/OS V1R10, and z/OS V1R9 with APAR OA29894)
Required if you use user exit routines.
When you run the SMF log stream dump program (IFASMFDL) or the SMF data set dump program (IFASMFDP), the name of an installation-written exit routine that is given control at the indicated times can be specified by the USER*x* (*x*=1, 2, or 3) parameter.
Beginning with z/OS V1R12 (and z/OS V1R11 and z/OS V1R10 with the PTFs for APAR OA29894 applied), to allow user exits to be called by the SMF dump programs, the exits must now be pre-defined to the system using the new keywords shipped in the SMFPRM*xx* parmlib member. The SMFDLEXIT keyword allows exits to be specified for the IFASMFDL program, and the SMFDPEXIT keyword allows exits to be specified for the IFASMFDP program.
Both keywords have the same suboptions, USER1, USER2 and USER3. The suboptions allow multiple exits to be specified for each user exit point in the respective dump program. The syntax follows.
```
SMFDLEXIT( USER1( exit1, exit2, ... ) | NOUSER1 ,
      USER2( exit1, exit2, ... ) | NOUSER2 ,
      USER3( exit1, exit2, ... ) | NOUSER3 )
SMFDPEXIT( USER1( exit1, exit2, ... ) | NOUSER1 ,
      USER2( exit1, exit2, ... ) | NOUSER2 ,
      USER3( exit1, exit2, ... ) | NOUSER3 )
```

Migration action: You must apply the PTF for APAR OA29894 if you share the SMFPRM*xx* parmlib member, with the keyword SMFDLEXIT or SMFDPEXIT specified, on multiple systems.
- Register an exit specified for IFASMFDL or IFASMFDP using the USER1, USER2 or USER3 parameters through the SMFPRM*xx* parmlib member. If you fail to do this, you will receive the following error message, and IFASMFDL or IFASMFDP will stop the processing: `IFA840I USER EXIT xxxxxxxx NOT REGISTERED WITH SYSTEM`
- View the D SMF,O command output to determine if the exits are successfully registered.

Note: You do not need to explicitly define the exits used by the operation of RACF SMF unload utility; they are registered by default:
```
SMFDLEXIT(USER2(IRRADU00),USER3(IRRADU86))
SMFDPEXIT(USER2(IRRADU00),USER3(IRRADU86))
```

Make IFASMFDL and IFASMFDP run in an authorized environment (Required-IF, as of z/OS V1R12, and z/OS V1R11, z/OS V1R10, and z/OS V1R9 with APAR OA29894)

Required if you do not invoke the SMF dump program as a jobstep task.

When you run the SMF log stream dump program (IFASMFDL) or SMF data set dump program (IFASMFDP), the DUMP function is permitted even in an unauthorized environment. (Note: The CLEAR function of the SMF data set dump program requires APF authorization. No such function exists for the SMF log stream dump program. Beginning with z/OS V1R12 (and z/OS V1R11 and z/OS V1R10 with the PTFs for APAR OA29894 applied), the IFASMFDL and IFASMFDP programs are required to run in an authorized environment. Otherwise, the programs will lose authorization and you will get an S338 followed by a S330 abend, especially if the programs are being executed under TSO/E.

Migration action:
- If you run the SMF dump program using JCL, the APF-authorization assigned to it is preserved, and no action is required.
- For the TSO/E environment, you need to add IFASMFDL and IFASMFDP to the AUTHPGM section of the IKJTSO*xx* parmlib member when the programs are being invoked using a TSO/E CALL command.
- If LINKMVS or ATTCHMVS is used in a REXX program to invoke IFASMFDL or IFASMFDP, change the invocations of IFASMFDL and IFASMFDP using LINKMVS or ATTCHMVS to use the TSO/E CALL command. In addition, add the IFASMFDL and IFASMFDP program to the AUTHPGM section of the IKJTSO*xx* parmlib member.
- If calling IFASMFDL or IFASMFDP from another program, that program must be authorized.

Note: A ++HOLD(ACT) is being shipped with the PTFs for APAR OA32258 to notify that IFASMFDL and IFASMFDP need to be executed in an authorized environment.

Specify installation script parameter when installing Predictive Failure Analysis (Required-IF, as of R12)

Required if you use PFA.

Before z/OS V1R12, when installing Predictive Failure Analysis (PFA) it was not necessary for you to specify how to handle the existing data in the check directories. Beginning with z/OS V1R12, you must append a parameter, *migrate* or *new*, on the installation script to specify if the PFA check directories retain data from the prior release. If you do not append the *migrate* or *new* parameter, the AIRSHREP.sh script fails.

Migration action: Provide the *migrate* or *new* parameter when running the PFA install script, AIRSHREP.sh, or when using the sample JCL for batch provided in SYS1.SAMPLIB.
- *migrate*: Use the *migrate* parameter to preserve PFA history data from the prior release. The *migrate* option is recommended for all installations that previously used PFA. When you specify the *migrate* parameter, the install script preserves data from the prior release and creates new directory structures for checks not previously installed on your system.
- *new*: Use the *new* parameter if you are installing PFA for the first time or if you want to delete everything from prior releases and start PFA with new directories. When you specify the *new* parameter, the install script deletes the existing check directories and creates a new directory structure for all the checks.

For example, to run the install script from your home directory:
1. On the OMVS command line, go to the home directory for the PFA user: cd /pfa
2. Run the install script appending the migrate parameter: /usr/lpp/bcp/ AIRSHREP.sh migrate

Refer to *z/OS Problem Management* for considerations and actions required for Predictive Failure Analysis (PFA) when falling back to z/OS V1R11 or z/OS V1R10 from z/OS V1R12.

Note that PFA is a new user of /etc. /etc/PFA/ini is a new file that contains consolidated path information. If found, this file will be used for all checks. However, if you want to specify different paths for different checks, you can continue to use the ini file found in each check directory.

☑ Change default locations for LCCA or PCCA control blocks to retain 24-bit virtual storage location (Required-IF, as of R12)

Required if you want to keep the 24-bit virtual storage location for IHALCCA or IHAPCCA.

In parmlib member DIAG*xx*, you can specify the virtual location of the LCCA control block (mapped by the IHALCCA macro) or the location of the PCCA control block (mapped by the IHAPCCA macro) to be used when a CPU is brought online. Before z/OS V1R12, if the IHALCCA or IHAPCCA structures specified on the CBLOC VIRTUAL24 or VIRTUAL31 keyword of DIAG*xx* did not specify either the 24-bit (VIRTUAL24) or 31-bit (VIRTUAL31) location, the default location for the LCCA or PCCA was in 24-bit virtual storage.

Beginning with z/OS V1R12, the default location for the LCCA and PCCA is now in 31-bit virtual storage if you do not specify VIRTUAL24 or VIRTUAL31 for the structures IHALCCA and IHAPCCA. If only one standard processor is online during the IPL, the LCCA and PCCA of the IPL CPU will be in 24-bit storage regardless of the specification.

Migration action: If you want to retain the 24-bit virtual storage location for IHALCCA or IHAPCCA, you specify VIRTUAL24 on the CBLOC keyword in DIAG*xx* parmlib member:

```
VIRTUAL24(IHALCCA)
VIRTUAL24(IHAPCCA)
```

Tip: The following z/OS V1R12 migration checks, (available with APAR OA32015 on z/OS V1R11 and z/OS V1R10), determine if your current settings match RMODE 31 for the LCCA or PCCA control blocks: CHECK(IBMSUP,ZOSMIGV1R12_SUP_LCCA_ABOVE_16M) and CHECK(IBMSUP,ZOSMIGV1R12_RCF_PCCA_ABOVE_16M). As is the convention, these checks are shipped inactive and are to be activated when exploring migration actions.

Remove reference to Unicode Services pre-built image CUNIDHC2 (Required-IF, as of R12)

Required if the SYS1.SCUNIMG is in the LNKLST specification.

Starting in z/OS V1R7, Unicode Services provided support to dynamically load tables into storage when a Unicode service request is made and the table is not already in storage. This enhancement is colloquially known as "Unicode On Demand." If the appropriate table needed to satisfy the service request is not in storage, Unicode Services will load the table dynamically without disrupting the caller's request. All tables needed for character conversion, case conversion, normalization, and collation services are now loaded automatically into storage when they are required and not already present. These tables are added to
other tables already in storage.

Starting in z/OS V1R12, the pre-built image CUNIDHC2 has been eliminated. This pre-built image contained all the conversion tables supported by DB2 and would be loaded into storage when you had an empty Unicode environment (no UNI=*xx* in the IEASYS*xx* member) and the first requestor of a Unicode conversion service would be DB2. Given that most customers would use only a handful of these tables and given that Unicode Service has the capability to dynamically load tables into storage, the need for pre-built image has become obsolete. Unicode Services will no longer ship the pre-built image SYS1.SCUNIMG(CUNIDHC2) and will no longer automatically load the pre-built image. We recommend that you use the Unicode On Demand capability to load all tables.

Migration action:

- Remove SYS1.SCUNIMG from the LNKLST specification.
- Remove SYS1.SCUNIMG from the APF authorization list. Note that if your installation is running with the entire LNKLST as APF authorized (LNKAUTH=LNKLST), you do not need to take any additional action. If your installation is not running with the LNKLST as APF authorized (LNKAUTH=APFTAB), you need to explicitly remove the APF authorization for this data set.
- Remove the catalog entry for SYS1.SCUNIMG.
- Remove the following DDDEF entries:

```
Name    Data Set Name
ACUNIMG SYS1.ACUNIMG
SCUNIMG SYS1.SCUNIMG
```

Note: If you want to create and load an image with the same conversion tables as the eliminated pre-built image, do the following:

1. Invoke the image generator program using the JCL provided in SYS1.SCUNJCL(CUNJIUTL). Use the conversion statements specified in SYS1.SAMPLIB(CUNSISM6). CUNSISM6 contains all the conversion statements needed to build the pre-built image.
2. After building the image, specify the image name in the CUNUNI*xx* parmlib member.
3. Specify the corresponding UNI=*xx* parameter in your IEASYS*xx* parmlib member and re-IPL.

Remove classification rules with the ETC work qualifier (Recommended, as of R12)

Not required, but recommended, otherwise you will have to delete the classification rules the next time you use the WLM ISPF application to modify the EWLM subsystem type.

Beginning with z/OS V1R12, the workload management (WLM) service definition no longer supports the work qualifier EWLM transaction class name (ETC) for classification rules of the subsystem type EWLM. If the activated service definition contains classification rules with qualifier type ETC, they are simply ignored by WLM. However, the next time you modify the classification rules for subsystem EWLM with the WLM ISPF Application, you have to delete the ETC rules before you save your modifications. Although z/OS V1R12 disregards classification rules with the ETC work qualifier, we recommend that you remove them. If you do not remove the rules, you will have to delete them the next time you use the WLM ISPF application to modify the EWLM subsystem type.

Migration action: If your WLM service definitions contain classification rules for subsystem type EWLM with the ETC work qualifier, start the WLM ISPF application and choose the Classification Rules option from the Definition Menu. Use the Modifiy option (3) for the IBM-supplied subsystem type EWLM. Delete all rows with the ETC qualifier type by using the Delete row Option (D).

Update the SFM policy to control automatic termination of impaired critical members (Recommended, as of R12)

Not required, but recommended when you want to designate how long XCF will wait before initiating termination of the impaired critical members.

Starting with z/OS V1R12, a member of an XCF group can identify itself as being critical to the operation of the group or the system. If a critical member appears to be inoperative (impaired) and the condition persists long enough, XCF automatically terminates the member in an attempt to resolve the problem. For a member that is critical to the operation of the system, this termination causes the system to be removed from the sysplex. For more information about XCF groups, see *z/OS MVS Setting Up a Sysplex*.

Members of the SYSGRS group, for instance, are critical to the operation of the system. If any GRS member is impaired, ENQ processing is likely impacted throughout the sysplex. Failure to perform ENQ processing in a timely fashion has significant negative impact. Thus if a GRS member appears to be impaired, XCF will automatically remove from the sysplex, the system on which that member resides.

You can set the MEMSTALLTIME parameter in your sysplex failure management (SFM) policy to control how long XCF allows a critical member to persist in an impaired state before it initiates termination of the member (or the member's system). If the MEMSTALLTIME specification resolves to NO (either implicitly or explicitly), XCF will terminate an impaired critical member if the condition persists as long as the failure detection interval (INTERVAL) of the system on which the member resides, or if the condition persists as long as two minutes, whichever is greater. To determine which groups are using the critical support, issue the appropriate XCF display command.

The MEMSTALLTIME parameter also determines how long XCF allows a signalling sympathy sickness condition to persist before terminating a stalled group member that is contributing to the problem.

The MEMSTALLTIME parameter indicates the number of seconds that XCF should wait before it terminates a member that is impacting the sysplex. A MEMSTALLTIME value of 120 (two minutes) seems to suit many installations as it provides some additional time for the system to resume normal operation, yet allows automatic action to resolve the problem before the sympathy sickness condition critically impacts the sysplex. Installations that resolve such conditions through manual intervention sometimes use a higher value to allow time for such intervention to be accomplished. Installations that are less able to tolerate sympathy sickness conditions sometimes set lower values.

Accommodate new REUSASID default (Required-IF, as of R12)

Required, as 0D3 abends might occur with downlevel products that provide no toleration support for reusable ASIDs. Because reusable ASIDs have been available since z/OS V1R9, it is reasonable to expect that the current levels of products are tolerant of reusable ASIDs.

In z/OS V1R9, the REUSASID(YES|NO) parameter in parmlib member DIAG*xx* was introduced with a default of NO. Starting with z/OS V1R12, the default is changed to YES.

When a reusable ASID is requested by the START command or the ASCRE macro, this reusable ASID is assigned if REUSASID(YES) is specified in DIAG*xx*. If REUSASID(NO) is specified in DIAG*xx*, an ordinary ASID is assigned. The default is REUSASID(YES). The use of reusable ASIDs might result in system 0D3 abends, if products or programs have not been upgraded to tolerate reusable ASIDs. For more information about reusable ASIDs, see *z/OS MVS Programming: Extended Addressability Guide*.

Migration action:
1. On z/OS V1R11 or z/OS V1R10 systems, specify REUSASID(YES) in the parmlib member DIAG*xx*. On z/OS V1R12 systems, keep REUSASID(YES), or allow it to default to YES. You can use the SET DIAG=*xx* command to change the REUSASID (YES|NO) option.
2. Verify that no 0D3 abends occur as a result.
3. If 0D3 abends do occur, apply appropriate maintenance to the affected code.

4. If this is not possible, specify REUSASID(NO)

Review the list of WTORs in parmlib member AUTOR00 (Required, as of R12)
In z/OS V1R12, the DDDEF"d PARMLIB provides an AUTOR00 member. This member should be found in your parmlib concatenation during IPL and will result in auto-reply processing being activated. If the WTORs listed in AUTOR00 are automated by your existing automation product, ensure that the replies in AUTOR00 are appropriate. You can see the contents of the default in AUTOR00 in *MVS Planning: Operations* and verify that the proper actions are taken.
Migration action: Examine the WTOR replies in the AUTOR00 parmlib member. If the replies or delay duration are not desirable, you can create a new AUTOR*xx* parmlib member and make corresponding changes. Also compare the replies to what your automation product would reply to these WTORs. Make sure that the AUTOR00 replies are in accordance with the replies from your automation product. IBM does not recommend making updates to AUTOR00, because updates to AUTOR00 might be made by the service stream or in new z/OS releases.
Notes:
1. If you have created an AUTOR*xx* parmlib member, update the IEASYS*yy* parmlib member that you use for IPL. Add the following statement to the IEASYS*yy* member: AUTOR=(xx,00) . Here *xx* corresponds to the AUTORxx parmlib member that you created. The IEASYS*yy* members specifying AUTOR cannot be shared with prior z/OS releases. If you only need the default AUTOR00 settings, you can omit specifying AUTOR= in IEASYS*yy*, and other z/OS levels can continue to use IEASYS*yy*. Even if AUTOR= is not specified in IEASYS*yy*, AUTOR00 is used if it exists.
   Note: If you have automation already in place for a WTOR and that WTOR now appears in AUTOR, the AUTOR version still can be used. Review the WTORs in AUTOR and your automation; either remove the WTOR from AUTOR or from your automation.
2. If you don't want to activate auto-reply processing, specify AUTOR=OFF in the parmlib member IEASYS*xx* or in response to message `IEA101A SPECIFY SYSTEM PARAMETERS`. It is not recommended that you remove AUTOR00 from parmlib, because service or new releases might reinstall AUTOR00. If there is no AUTOR00 member in parmlib, auto-reply is not activated and the following messages are produced:
```
CNZ2600I AUTO-REPLY POLICY ATTEMPTING TO USE AUTOR=00.
IEA301I AUTOR00 NOT FOUND IN PARMLIB
CNZ2601I AUTO-REPLY POLICY NOT ACTIVATED.
NO ENTRIES SPECIFIED
```
3. The IEASYSyy members specifying AUTOR=OFF cannot be shared with prior z/OS releases.

Update automation that handles some deleted SMF messages: IEF374I and IEF376I (Required-IF, as of R12)
Required if you use automation programs or other procedures to handle messages IEF374I and IEF376I.
Starting in z/OS V1R12, message IEF374I is replaced by message IEF032I and message IEF376I is replaced by message IEF033I. If you use automation programs to handle messages, or you have operator or other procedures that deal with messages, you should update the programs or procedures appropriately.
Migration action: Modify automated actions for IEF374I so they now work with message IEF032I. Modify automated actions for IEF376I so they now work with message IEF033I.

Ensure correct use of CIRB (Required, as of R11)
When the CIRB function (Create Interruption Request Block) is used with SVC-entry and the IRB is to be run on a task different than the task that issued CIRB (identified within field IQETCB), and the STAB=DYN option was requested, the system is told to free storage. Before z/OS V1R11, the freemain was conditional and would always fail (with a bad return code) in the case mentioned above because the storage was not owned by the freeing task. Since you would not see this, as it was only a return code, you were not aware of the failure. Starting with z/OS V1R11, the freemain is changed to unconditional so that you will be aware of your error and can fix or change your code.
Migration action: The best practice is to change to use SCHEDIRB instead of CIRB. SCHEDIRB has been recommended for many years. Otherwise, take one of the following actions according to your need:
- If you do not need the system to free the storage, do not specify STAB=DYN.
- If you do need the system to free the storage, make sure that you are in PSW key 0 and place the TCB address under which the IRB will run in register 4 when you invoke CIRB and specify the BRANCH=YES option of CIRB.

Ensure your ESQA specification in IEASYSxx is adequate (Required-IF, as of R11)

Required if your ESQA specification in IEASYSxx is inadequate.
In z/OS V1R11, each address space requires 1608 bytes of additional 31-bit Extended System Queue Area (ESQA) storage. The additional ESQA storage is allocated at address space create time and freed at address space termination time. Therefore, additional ESQA is only required for the typical peak number of active address spaces. If ESQA storage is unavailable at any time to the system, the system will try to obtain the storage needed from Extended Common Storage Area (ECSA). When ESQA expands into ECSA, IRA104E will be issued. If ESQA and ECSA are unavailable, the system allocates space from SQA followed by CSA below 16 megabytes. As SQA storage becomes scarce, the system rejects LOGON, MOUNT, and START commands, and stops new jobs from being created and address spaces from being swapped in. ESQA/SQA storage shortages can eventually result in a system failure.

Migration action:
1. Determine the number of address spaces that you have on the system. One way to do this is to issue the DISPLAY A,L command and total the address spaces in the IEE114I and IEE115I messages. Note that the additional ESQA storage is allocated at address space create time and freed at address space termination time. Therefore, you might only need to have additional ESQA for the typical peak number of active address spaces.
2. Multiply the number of address spaces by 1608 (x'648') bytes to get the total additional ESQA storage required.

Tip: Ensure that your SQA= statement in IEASYSxx specifies an adequate amount of ESQA. Use VSM_SQA_THRESHOLD, provided by the IBM Health Checker for z/OS, from the prior release to examine the ESQA in use percent and high water mark (HWM) percent. Then add the additional ESQA storage needed from the prior step to calculate the projected ESQA in use and HWM percent. Adjust the amount of ESQA needed by your installation.

Use the new default of COUPLExx parmlib member INTERVAL (failure detection interval) (Recommended, as of R11)

Not required, but recommended because the new default value can ensure consistency with the existing excessive spin parameters that are specified through EXSPATxx
The default handling of the COUPLExx parmlib member INTERVAL (failure detection interval) value is changed to ensure consistency with the existing excessive spin parameters that are specified via EXSPATxx.
Before z/OS V1R11, XCF used twice the default spin loop timeout value, plus 5, for the default failure detection interval. Note that XCF used the default spin loop timeout value and not what you specified in the EXSPATxx parmlib member.
With z/OS V1R11, XCF derives the default INTERVAL value from the excessive spin processing parameters in EXSPATxx. The value is computed as follows: (N+1)* SpinTime + 5 where N is the number of excessive spin recovery actions, +1 indicates the implicit spin action, and SpinTime is the excessive spin loop timeout interval. The result might be higher effective failure detection intervals being used than the value that was explicitly coded in the COUPLExx parmlib member INTERVAL keyword.
If the excessive spin parameters are not explicitly set via an EXSPATxx parmlib member, the IBM-supplied default spin parameters are used. The value is computed as follows: (1+4) * SpinTime + 5
Migration action: If the COUPLExx parmlib member codes INTERVAL, perform the following actions as needed:
- If the specified INTERVAL is greater than the default INTERVAL that will be derived from the EXSPATxx specifications, no action is needed.
- If the specified INTERVAL is less than the default INTERVAL and you want to use the specified INTERVAL, enable the USERINTERVAL switch.

If the COUPLExx parmlib member does not specify INTERVAL and the default INTERVAL value is taken, perform the following actions as needed:
- If the new default INTERVAL is acceptable, no action is needed.
- If you want to use the old default, you must code INTERVAL to set the desired value and proceed as above.
- If COUPLE does not specify INTERVAL but specifies OPNOTIFY, you should consider specifying a relative OPNOTIFY value so that OPNOTIFY can be adjusted to maintain the same relative delta between the old default INTERVAL and the new default INTERVAL.

Note: If a COUPLExx parmlib member is used by more than one system, the migration actions for the new system might conflict or cause different behaviors on pre-z/OS V1R11 systems.

Accommodate quoted argument changes in System REXX MODIFY AXR command (Required-IF as of R11, and back to R9 with OA26802)

Required if quotation marks are used in the MODIFY AXR,rexxexecname args command and the REXX exec expects and processes the quotation marks.

Before z/OS V1R11 (and APAR OA26802 on prior releases), arguments bounded by quotation marks were passed through to the REXX exec unaltered (including the quotation marks). With changes introduced in z/OS V1R11 (and by APAR OA26802 on prior releases), the final argument string that is passed through to the REXX exec will have bounding quotation marks removed and single quotation marks will be substituted and passed as part of the argument, when two single consecutive quotation marks are encountered within a quoted string. Arguments not within quotation marks continue to be upper-cased by system command processing.

Migration action: Determine if you have the new behavior (step 1), then proceed to steps 2 and 3 if quotation marks are currently being used within the argument string of the MODIFY AXR command.

1. The presence or absence of this enhancement can be detected within a program as follows:
   a. REXX – Issue the MODIFY AXR,SYSREXX REXXLIB command from the exec by including the statement AxrCmdRc = AXRCMD('F AXR,SR R',Msg.,4). When this support is not present ASA100I SYNTAX ERROR: STATUS is returned in the Msg.1 stem variable.
   b. ASSEMBLER – include an AXREXX macro call specifying REQUEST=GETREXXLIB. Return code 8 reason code AxrBadRequest is returned when the AXREXX GETREXXLIB support is not present. AxrBadRequest is defined as x'00000846' in SYS1.MACLIB(AXRZARG).
2. Preserve the lower case of the argument. To preserve the lower case of arguments, it is necessary to bound them by quotation marks. As bounding quotation marks are removed with z/OS V1R11, this new behavior might affect argument processing within the REXX exec. Within the REXX exec, it is unnecessary to check for bounding quotation marks when parsing the argument string.

Input argument string	Resolved argument string
'www.lakeminnewaska.org'	www.lakeminnewaska.org
'Fire declared ' out	Fire declared OUT

3. Pass quotation marks as part of the argument. If quotation marks are currently used, it will be necessary to use the new syntax when entering the argument string portion of the MODIFY AXR command when running a REXX exec from the console. Bounding quotation marks are removed with z/OS V1R11. To preserve a single quotation mark in an argument string, two consecutive quotation marks must be imbedded within the argument string. For example, 'The cat''s tail. would be resolved to The cat's tail.
4. Additonal examples follow:

Input argument string	Resolved argument string
'Minnewaska''s Scenic Beauty'	Minnewaska's Scenic Beauty
''' Shawangunk' Mountain 'Trails'	' Shawangunk MOUNTAIN Trails
'''Lake Awosting'''	'Lake Awosting'
''Ranger Guide''	RANGER GUIDE
'''''Mossy Glenn Footpath'''''	''Mossy Glenn Footpath''

Remove message flood automation exits (Required-IF as of OA25602 on R9 and R10)

Required if message flood automation is used.

Before APAR OA25602 on z/OS V1R10 and prior releases, if you wanted to implement message flood automation, you had to do so with a command exit specified in parmlib member MPFLST*xx* and with message processing installation exit IEAVMXIT. Beginning with APAR OA25602, message flood automation is integrated in z/OS, eliminating the use for the exit routines. If you want to continue using message flood automation, you must remove what you have implemented with the exit routines.

Migration action:

- Update your MPFLST*xx* parmlib member to remove all .CMD USEREXIT(CNZZCMXT) statements.

- If you use exit IEAVMXIT only for message flood automation, update your CONSOL*xx* parmlib member to change the INIT statement option of UEXIT(Y) to UEXIT(N). Do not do this if you want to continue to use IEAVMXIT for other purposes.

- If you modified your IEAVMXIT exit routine to incorporate message flood automation processing, you must remove the message flood automation code, reassemble, and use the linkage editor or program management binder to install. In order to successfully reapply IEAVMXIT, the USERMOD must be removed using the SMP/E RESTORE command prior to the APPLY of the PTF for APAR OA25602, and should be discarded since it is no longer required. **Note:** Before APAR OA25602, messages could be made exempt from Message Flood Automation processing by coding a simple BR 14 MPF exit and adding the message to the MPFLST*xx* parmlib member. This way of bypassing Message Flood Automation processing no longer works with APAR OA25602 because Message Flood Automation now processes all messages before MPF receives control. You must review ALL MPF exits to see if you have used this technique to bypass Message Flood Automation and make changes required. To exempt messages from processing by Message Flood Automation with APAR OA25602 installed, the preferred technique is to add the message to the SPECIFIC message list in the MSGFLD*xx* PARMLIB member, specifying the IGNORE keyword.

If the message flood automation exits are not removed, message flood automation might not be operable. If it is operable, it will be the old message flood automation, not the new, integrated message flood automation.
Tip: Use IBM Health Checker for z/OS check CNZ_OBSOLETE_MSGFLD_AUTOMATION introduced in APAR OA25602, available in z/OS V1R11, to validate that the obsolete version of message flood automation is not in use.

Accept the new lower storage limit for stopping SVC dump data capture (Recommended, as of R11)
Not required, but recommended because it limits DUMPSRV's use of virtual (and therefore also auxiliary) storage while reducing the tuning efforts required to support a given MAXSPACE value.
Before z/OS V1R11, SVC dump data capture would not be prevented or stopped until auxiliary storage usage reached 85%. At this level of utilization the system might be unresponsive. For sure, no new jobs could be started. As of z/OS V1R11, the new AUXMGMT=ON default lowers the thresholds at which SVC dump capture activity is affected. New SVC dumps are not allowed when auxiliary storage usage reaches 50%, and an SVC dump that is in the process of capturing data is truncated, resulting in a partial dump, when auxiliary storage usage reaches 68%. Once either limit is reached, data capture for new dump requests is not allowed again until auxiliary storage usage drops below 35%. This new behavior makes system availability a higher priority than capturing failure data (a complete dump) when resources are restricted. It also makes tuning the MAXSPACE parameter less daunting. A very large MAXSPACE value can be specified because the amount of available auxiliary storage is a key factor in automatically limiting what SVC dump data capture uses. The lower 68% point leaves the system in a more responsive state.
Migration action: Do one of the following:
- To use the new limits, there is nothing to do. The system always IPLs with AUXMGMT=ON by default. Note that you can display the current status of AUXMGMT by using the DISPLAY DUMP,OPTIONS (D D,O) command.   **Note:** Once SVC dump processing has detected a shortage, the auxiliary storage utilization must drop below 35% before new SVC dump requests will be honored. The condition cannot be removed by simply changing the setting of AUXMGMT from ON to OFF.
- If you have spent considerable effort tuning the real, auxiliary, and MAXSPACE resources for a particular behavior, you might want to keep the pre-z/OS V1R11 behavior. To suppress the new behavior and use the prior limits, ensure that the CHNGDUMP SET,SDUMP,AUXMGMT=OFF command is issued out of a SYS1.PARMLIB(COMMND*xx*) member that receives control during IPL. Remember that with AUXMGMT=OFF, the MAXSPACE value must be accompanied by the appropriate increase in auxiliary storage (paging) resources (at least three times the MAXSPACE value), above the system's worst case usage of auxiliary storage

Note: In the case where AUXMGMT=ON, and new SVC dumps are prevented because auxiliary storage usage has exceeded 50%, then message `IEA412I SLIP TRAP ID=GK01, 1 SDUMPS NOT SCHEDULED. RETURN CODE=08, REASON CODE=46` is issued. When SVC dump processing has detected a shortage, the auxiliary storage utilization must drop below 35% before new SVC dump requests will be honored. The condition cannot be removed by simply changing the setting of AUXMGMT from ON to OFF.

Adjust interval time of recovery allocation message reminders (Recommended, as of R11)
Not required, but recommended if you do not want to receive frequent reminder messages indicating the pending of recovery allocation.
Currently, when recovery allocation messages IEF238D, IEF433D or IEF434D are issued, system resources, such as the SYSIEFSD.Q4 ENQ resource, are held while the WTOR message is outstanding.

```
IEF238D jobname - REPLY [DEVICE NAME] [,] ['WAIT'] OR 'CANCEL'
IEF433D jobname - WAIT REQUESTED -- REPLY 'HOLD' OR 'NOHOLD'
IEF434D jobname - INVALID REPLY. REPLY 'HOLD' OR 'NOHOLD'
```
If the operator does not respond to the message prompt in a timely manner, it might affect other system processes, such as VARY OFFLINE command. Beginning in z/OS V1R10, message IEF882E is issued every 90 seconds to remind the operator to respond to the outstanding recovery allocation messages.
```
IEF882E jobname [procstep] stepname IS WAITING FOR A REPLY TO pmsgid
```
When the operator responds with WAIT or HOLD to messages IEF238D and IEF433D, system resources are held while waiting for another job to free a device for use by this job. Beginning with z/OS V1R10, message IEF883E is issued every 90 seconds to remind the operator to respond to the situation.
```
IEF883E jobname [procstep] stepname IS WAITING FOR DEVICE(S) WHILE HOLDING SYSTEM
RESOURCES
```
Beginning with z/OS V1R11, a new option is available that permits you to adjust the 90-second default interval time for reminder messages IEF882E and IEF883E. You can now specify, in seconds (10-999), the amount of time between display of the reminder messages, or you can specify 0 to disable them.
Note: Multiple IEF882E or IEF883E messages may be issued if several jobs have entered IEF238D processing. The message is deleted when the reminder is no longer needed, such as when the WTOR is answered, the job is canceled, or a new message with an updated time value is issued. The descriptor code of IEF882E and IEF883E is 11, therefore, the default attribute is highlighted in bold, red text.
Migration action:
- To specify the intervals for messages IEF882E or IEF883E, use the SYSTEM REMIND_INTV(*xxx*) parameter in the ALLOC*xx* parmlib member or the SETALLOC command. The number of seconds (10-999) you set indicates how many seconds between reminder messages.
- To disable display of the messages, you can set the interval to 0. SYSTEM REMIND_INTV(0)

Tip: Consider using the POLICY statements in the ALLOC*xx* parmlib member to automate responses to IEF238D and IEF433D and eliminate delays caused by these messages or eliminate slowdowns caused by the WAIT/HOLD function.

BCP Migration Actions Post-First IPL
Carry over your existing CPPC policy (Required-IF, as of R12)
Required if you want to continue using your existing CPCC workspace.
Beginning with z/OS V1R12, the installation process for Capacity Provisioning Control Center (CPCC) on a Windows® workstation is new. The code is delivered as an MSI package and is installed using the Windows® Installer. The new CPCC can coexist with older versions of the CPCC on a workstation. However, z/OS V1R12 CPCC uses a different default workspace. Therefore, if you want to continue using your existing CPCC workspace, point to the respective directory during the installation of z/OS V1R12 CPCC. It is also possible to change the workspace when the CPCC starts.
Migration action:
- Install z/OS V1R12 CPCC using information in *z/OS MVS Capacity Provisioning User's Guide*.
  Note: With z/OS V1R12, the location of the default workspace changes from, for example,
  C:\Documents and Settings*user*\My Documents\IBM\IBM Capacity Provisioning Control Center
  to
  C:\Documents and Settings*user*\Application Data\IBM\IBM Capacity Provisioning Control Center
  The actual directories depend on the Windows version that you are using.
- If you want to continue using your existing (pre-z/OS V1R12) CPCC workspace, point to the desired workspace during the installation process. You can also switch the workspace at a later time by navigating to the desired workspace when the CPCC starts.

Evaluate applications that parse AMBLIST command LISTLOAD or LISTIDR output (Required-IF, as of R12)
Required if your application depends on the output generated by the AMBLIST command LISTLOAD or LISTIDR control statement processing.
1. Prior to z/OS V1R12, the AMBLIST command LISTLOAD output for load modules, in the MODLIST area of the output, AMBLIST printed only the hexadecimal representation of TEXT records. Beginning with z/OS V1R12, AMBLIST command LISTLOAD output for load modules is made to work similarly to the same output for program objects. In particular, the output now includes the TEXT records output in EBCDIC format to the right of the hexadecimal representation. This also means that some of the columns for the hexadecimal output have been shifted.
2. Prior to z/OS V1R12, the AMBLIST commands LISTLOAD and LISTIDR output header for UNIX program objects printed the constant ****UNIX**** for both MEMBER NAME and LIBRARY, in the output header for each

processed UNIX program object. Beginning in z/OS V1R12, AMBLIST commands LISTLOAD and LISTIDR output header for UNIX program objects display the actual directory path name and file name of the UNIX file. It is represented this way regardless of whether or not the MEMBER is specified as a relative path name. Note that MEMBER NAME and LIBRARY still only utilize a single output line each, therefore, if either exceeds the number of allotted characters they will be truncated on the left and preceded by two periods and a space (..).

Migration action: Output from the AMBLIST command is not an intended programming interface.

1. Evaluate applications that parse AMBLIST command LISTLOAD output to ensure that either there is no dependency on the TEXT output alignment or the additional data to the right of the hexadecimal output.
2. Evaluate applications that parse AMBLIST command LISTLOAD or LISTIDR output to ensure that there is no dependency on either the MEMBER NAME or LIBRARY in the output header being the constant ****UNIX**** or any other particular format.

To reduce future impact and maintenance, IBM suggests migrating the parsing routines or applications to use an IBM provided programming interface such as the Program Management APIs. See Using the binder application programming interfaces (APIs) in*z/OS MVS Program Management: Advanced Facilities*.

Evaluate applications for sensitivity to changes to binder API GETE return and reason codes
(Required-IF, as of R12)

Required if your application depends upon both the return code and reason code to be 0 from the binder regular API GETE function call or binder C API __iew_getE() function call.

Prior to z/OS V1R12, the binder regular API GETE (GETESD) function call and the binder C API __iew_getE() function call would, in some circumstances, end with a return code of 0 and reason code of 0 even when there was no more ESD information of the specified type to be returned. This would only occur with certain combinations of the optional parameters that allow selection criteria.

Beginning with z/OS V1R12, these functions always end with a return code of 4 and a reason code of 83000800 in this circumstance. This return code and reason code combination identifies that an end-of-data condition was detected while some data of the specified type might still have be returned. It is necessary to examine the COUNT parameter to determine the number of data entries that have been returned, which might be 0.

Migration action:

*   Check for application source that uses binder regular API GETE function call. This is most commonly done using the IEWBIND High Level Assembler macro, though the parameter list can be coded directly without the assistance of the macro.
*   Check for applications that use the binder C API __iew_getE() function call. Source code can be checked for this function call, and object modules and program modules (if they are not bound with EDIT=YES) can also be checked for the external symbol reference using AMBLIST.

If such usages are found, make sure the "successful" condition includes a check for RETCODE=4, RSNCODE=83000800 in addition to RETCODE=0, RSNCODE=0, and that the code logic takes into consideration that the RETCODE=4, RSNCODE=83000800 might be accompanied with COUNT=0.

Ensure analysis tools interacting with HIS output accommodate HIS state change events
(Required-IF, as of R12)

Required if you start HIS and have programs that analyze the HIS output.

Hardware instrumentation services (HIS) is a function that collects hardware event data for processors in SMF records type 113, subtype 2, as well as UNIX System Services output files. You can only use HIS for IBM System z10 or later machines. In z/OS V1R12 (and in z/OS V1R11 and z/OS V1R10 with the PTF for APAR OA30486 installed), functionality is added to the HIS component which causes changes in the output filename formats produced by HIS (.CNT, .SMP, .MAP), as well as introducing additional lines to the .CNT file possibly causing incompatibilities. In addition, an increase in SMF Type 113 records might be noticed.

Any tools that programmatically open the HIS output files (.CNT, .MAP, or .SMP), and any tools that programmatically analyze the HIS output .CNT file, should be analyzed and updated to accommodate the new formats.

Migration action: What to look for:

*   The new filename output format is an indication that the support is installed.
*   The first line of the .CNT file indicates output version 2.
*   New output message HIS032I "STATE CHANGE DETECTED ACTION=action."
*   An additional line in the output of DISPLAY HIS command, which describes the action that should be taken should a state change event occur (specified by the operator at the start of a collection run).

What to do:

*   If programmatically opening files, ensure the new output file format is handled.

SHARE
Boston, MA
© 2010 IBM Corporation

August 4, 2010

Page 31 of 94

Session 7398

- If programmatically parsing the .CNT file, ensure to check the VERSION identifier in the header. If the identifier is VERSION 2, be prepared for the new STATECHANGE line.

Detect program objects that have multiple INITIAL LOAD segments (Required-IF, as of R12)

Required if the following conditions Only users or ISVs that produce program object programs, which might have multiple INITIAL LOAD segments, need to take action if all of the items listed below are true:

- *Program is required to reside in a program object.*
- *Program has multiple segments.*
- *Program has multiple initial load classes.*
- *Program has mixed RMODEs.*
- *Program is link-edited with the RMODE option to override the binder default.*

Note: In most cases, even if a program has multiple segments containing INITIAL LOAD classes, no action is required.

Prior to z/OS V1R12, the binder RMODE option only applied to the first module segment, which contains some but possibly not all the initial load classes, though always the class wherein lies the entry point. Subsequent segments containing other initial load classes were not affected by the RMODE option, and thus the binder determined the RMODE based on the attributes of the classes contained therein.

Beginning with z/OS V1R12, the binder RMODE option applies to all initial load classes by default. Thus all segments containing initial load classes are affected. This new behavior takes affect only when the RMODE binder option is specified.

Also the RMODE option has been expanded so that either the new or previous behavior can be explicitly requested.

Migration action: To determine whether you might be affected by this change, you can run the AMBLIST service aid against your program objects and examine the output. However, if you do not explicitly specify the RMODE option when you bind the program, it is not affected even if the following two conditions are true. Run AMBLIST using the LISTLOAD control statement. Using the OUTPUT=MAP option results in a smaller amount of output that still contains the pertinent information.

The following two conditions must both be true for the program to be affected by
the new RMODE option behavior:

1. In the Module Summary, only programs that are program objects and are PO FORMAT: 3 or higher are affected.
2. In the SEGMENT MAP, only programs for which there is class entry SEGMENT 2, and TYPE INITIAL, are affected. **Note:** The RMODE differs for the SEGMENT 1, INITIAL load classes, and the SEGMENT 2, INITIAL load classes.

Given that these two conditions are true, there are then generally three possible situations:

1. If the RMODE option is not being specified, there is no need to do anything, as the behavior is identical in z/OS V1R12.
2. If you specified RMODE=ANY, expected all segments to be RMODE=31, but find that one segment is RMODE=24, part of the program is using below-the-line storage. In most cases, this is not desirable and the new binder behavior will remedy the situation. It is possible that you rely on having part of the program use below-the-line storage. You can use RMODE(ANY,COMPAT) to revert to the pre-z/OS V1R12 behavior.
3. If you specified RMODE=24, expected all segments to be RMODE=24, but find that one segment is RMODE=31, part of the program is using above-the-line storage. In most cases, this is desirable and the new binder behavior could cause a problem by causing the program to use more below-the-line storage. It is also possible that you rely on having part of the program use above-the-line storage. You can use RMODE(24,COMPAT) to revert to the pre-z/OS V1R12 behavior.

Modify SMF exit routine IEFACTRT because of lengthened fields (Required-IF, as of R11)

Required if you use SMF exit routine IEFACTRT.

A sample IEFACTRT exit routine is provided in SYS1.SAMPLIB in member IEEACTRT. This sample exit routine puts a job summary in the log of a job using WTO with routing code 14. The summary includes both step and job information. To accurately reflect data for long-running job steps, the following fields in the output generated by the sample IEFACTRT exit routine have increased in size:

Field name	Pre-z/OS V1R11 field length	z/OS V1R11 field length
TCB	6 bytes	9 bytes
SRB	6 bytes	9 bytes
SERV	6 bytes	13 bytes

If you use IEFACTRT, whether the IBM sample or a routine that you wrote, make sure that it can handle the increased field lengths.
Migration action: Review your IEFACTRT exit routine to make sure that the changed fields are long enough, and modify it if necessary. If you are using the IBM sample, rebase on the sample because it accommodates the longer field lengths. If you are using your own exit routine, verify that the longer field lengths are accommodated. Also review and change, if necessary, any automation programs that are affected.
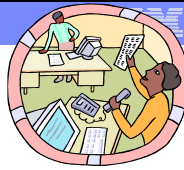

Keep the real frames included in GRS real storage consumption for use (Recommended, as of R11)

Not required, but recommended to avoid confusion caused by the expected change in the report GRS real storage usage.

Before z/OS V1R11, Resource Management Facility (RMF) reports and other monitoring tools might indicate a significant growth in Global Resource Serialization (GRS) real frame usage compared with the previous release. Starting with z/OS V1R11, GRS's QSCAN buffer processing is changed to no longer require that frames backing discarded virtual pages be immediately returned to the RSM available frame queue, but are now taken by the system if needed. This approach improves the storage management of the buffers as they usually do not need to be re-backed as QSCANs come and go. Thus, GRS is not really using any more frames, and the frames are still available but can be taken if needed by RSM due to the storage being needed elsewhere. The real frame usage count includes both frames that are in use by the address space and ones that are readily available for the system to reclaim when they are needed.
Migration action: No action is necessary to accept the new behavior. To avoid any confusion, inform the system administrator.

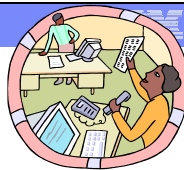## Communications Server Migration Actions for z/OS R12 from z/OS R10

Migration Actions You Can Do NOW:

- **IP Services: Migrate from Network Database function (Req-IF, as of R11)**
  - •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 (Req-IF, as of R11)**
  - •Investigate using a DHCP server on Linux for System z.
- **IP Services: Migrate from the Boot Information Negotiation Layer function (Req-IF, as of R11)**
  - •Consider using the product IBM Tivoli Provisioning Manager for OS Deployment V5.
- **IP Services: Migrate from BIND DNS 4.9.3 (Req-IF, as of R11)**
  - •For a caching-only name server, use z/OS Resolver DSN Caching function in R11 instead.
  - •Some /etc configuration files have changed for this removal.
- **IP Services: Migrate from BIND 9.2.0 (Recommended)**
  - •For a caching-only name server, use z/OS Resolver DSN Caching function in R11 instead.

Communications Server Migration Actions for z/OS R12 from z/OS R10

Migration Actions Before Install:

- **Accommodate DEFAULT route usage changed for CINET environments (Required-IF, as of R11)**
  - •Before R11, the DEFAULT route was tracked only as a flag for each stack, indicating that the stack had a DEFAULT route defined. These routes were masked by selection logic.
  - •As of R11, the DEFAULT route is treated like all other subnet and network routes, including the potential for multiple duplicate routes on a stack, having an associated cost, and selection of a stack with the least cost for that route (which is generally desirable). However, on systems where multiple stacks have a DEFAULT route defined, they can now be selected causing undesirable results (including connection timeouts).
  - •Follow instructions in the FlashAlert (and in handout).

Migration Actions Pre-First IPL:

- **IP Services: Ensure the TN3270 Telnet server can end automatically when OMVS shutdown is issued (Required-IF, as of R12)**
  - •Before R12, Telnet was active but unusable when you issued F OMVS,SHUTDOWN
  - •As of R12, Telnet stops automatically when you issue F OMVS,SHUTDOWN. Ensure you are aware the Telnet server will stop when this command is issued.

Communications Server Migration Actions for z/OS R12 from z/OS R10
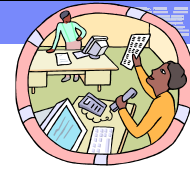
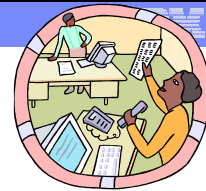**Migration Actions Pre-First IPL:**

- **IP Services: Update syslogd config for archiving rules with shared z/OS UNIX file destinations (Required-IF, ONLY on R12)**
  - On R11, syslogd could perform automatic archiving.
  - As of R12, if you use the same z/OS UNIX file destination on multiple syslogd rules, automatic archiving is not supported on any of the rules.
  - Change the destination to be unique, to preserve syslogd automatic archiving.

- **Update procedures that use the syslogd job name (Required-IF, as of R11)**
  - Before R11, the job name of the syslog daemon was the cataloged proc name plus an additional character (usually "1"), when started from a proc and that proc was less than 8 chars in length. An additional char was also appended to the program name when started from the z/OS UNIX shell.
  - As of R11, syslog daemon job name is the same as the proc name or the program name.
  - Update any automation. If you run two copies, ensure they will be uniquely named. Update the syslogd UDP port reservation in the TCP/IP profile to use the syslogd proc name.

Communications Server Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **IP Services: Run the syslog daemon in the background when starting from the UNIX shell (Required-IF, as of R11)**
  - As of R11, syslogd does not automatically run in the background when started from the UNIX shell (or from a shell script like /etc/rc) or in BPXBATCH.
    - If it is started from /etc/rc or another shell script, or from BPXBATCH, add a trailing ampersand character.
    - If you start syslogd manually from the UNIX shell, include a trailing ampersand on the command line.
    - If you start syslogd from a cataloged procedure that uses BPXBATCH, add a trailing ampersand character, and after that include a sleep command in your script to give syslogd time to initialize before the shell script ends. Use a file for STDOUT and STDERR DD's (not a data set).
    - If you start syslogd from a cataloged procedure that does not use BPXBATCH, no changes are needed.

Communications Server Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-IPL:

- **SNA Services: Ensure that dump data sets are large enough to include VTAM internal trace data space (Required, as of R11)**
  - Prior to R11, VTAM INOP dump processing did not capture VIT data space in the dump.
  - As of R11, this processing automatically captures the VIT data space, when VIT is in use.
  - Ensure that the dump data set allocation will accommodate this add'l space.
  - Health Check determines whether the VIT data space table size is 50 MB (5).
- **SNA Services: Increase region size for VTAM (Required-IF, as of R11)**
  - As of R11, ECSA usage for HPR is reduced. However, VTAM private storage usage has increased. The increase depends upon the number of RTP endpoints that are supported on the LPAR. If VTAM is using nearly all its allocate region size, you'll need to increase it.
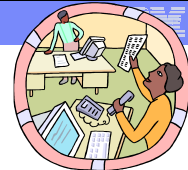 - Increase region size for VTAM by 1.5 MB for each 1,000 RTP pipes.

21   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

Communications Server Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **IP Services: Accept new resolver behavior of saving the results of DNS queries (Recommended, as of R11)**
  - Prior to R11, the z/OS system resolver did not cache results of DNS queries.
  - As of R11, the resolver automatically saves these results.
  - New behavior requires no action! This give you performance and usability enhancements.
  - If you want prior behavior (disable resolver DNS caching function) or you want to selectively limit the caching function, you must take action.
- **IP Services: Disable resolver monitoring of name server responsiveness (Required-IF, as of R12)**
  - Prior to R12, the z/OS system resolver did not provide notification where a DNS name server was not responding to resolver queries sent to the name server.
  - As of R12, the resolver will notify the operator when an unresponsive name server is detected by default.
  - If you do not want notification of potential problems, then use UNRESPONSIVETHRESHOLD(0) in the resolver setup file. 0 means not to monitor the name server responsiveness. Default is 25%: percentage of resolver queries to monitor.

22   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

Communications Server Migration Actions for z/OS R12 from z/OS R10
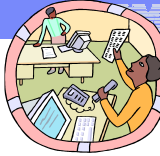
Migration Actions Pre-First IPL:

- **IP Services: Restore resolver UDP request timeout interval duration (Required-IF, as of R12)**
  - Prior to R12, default setting for considering UDP requests to a name server to have time out was 30 seconds (as specified by RESOLVERTIMEOUT).
  - As of R12, the default is 5 seconds.
  - If you want prior behavior, specify RESOLVERTIMEOUT 30 in TCPIP.DATA.
- **IP Services: Update modified Netstat message catalogs to include timestamp (Required-IF, as of R12)**
  - As of R12, Netstat verifies its message catalogs are at the same maintenance level as the Netstat program. If the timestamps don't match, default messages are used.
  - For each customized message catalog (netmsg.cat or netmsg6.cat), take action.
- **IP Services: Update automation that keys on TN3270E Telnet server messages (Required-IF, as of R12)**
  - Prior to R12, all Telnet messages began with "TELNET" and DEBUG, LUNS, and LUNR message began with "TELNET jobname" after the message id.
  - As of R12, the first word "TELNET" is replaced with the jobname, and DEBUG, LUNS and LUNR are shifted left.

23 Migrating to z/OS 1.12: Part 2 of 2 © 2010 IBM Corporation

Communications Server Migration Actions Between z/OS V1 R10 and z/OS V1 R12

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 (Required-IF, as of R11)

Required if you use the NDB function.
z/OS V1R10 was the last release in which z/OS Communications Server supported the Network Database (NDB) function. Starting with z/OS V1R11 you have to use the distributed data facility (DDF) provided by DB2 for z/OS, and the DB2 Runtime Client.
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 (Required-IF, as of R11)

Required if you use the DHCP server function.
z/OS V1R10 was the last release in which z/OS Communications Server supported the Dynamic Host Configuration Protocol (DHCP) server function. Starting with z/OS V1R11, you have to use a DHCP server on a system other than your z/OS system..
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 (Required-IF, as of R11)

Required if you use the BINL function.
z/OS V1R10 was the last release in which z/OS Communications Server supported the Boot Information Negotiation Layer (BINL). Starting with z/OS V1R11, you have to use another product, such as IBM Tivoli Provisioning Manager for OS Deployment V5 (5724-Q99), for network-based operating system installation services.
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 (Required-IF, as of R11)
Required you use BIND DNS 4.9.3.
z/OS V1R10 was the last release in which z/OS Communications Server supported BIND DNS 4.9.3.
Migration action:
- If you have been using BIND DNS 4.9.3 as a caching-only name server, use the z/OS Resolver DNS Caching function in z/OS V1R11 instead.
- If you have been using BIND DNS 4.9.3 as a primary or secondary authoritative name server, migrate to BIND DNS on Linux for System z.
- If you want to implement a temporary replacement, implement BIND DNS 9.2.0, which has been included in z/OS beginning with V1R4. Use the dnsmigrate tool to migrate a z/OS BIND 4.9.3 DNS server configuration file to a z/OS BIND 9.2.0 DNS server configuration file. Note that support for BIND DNS 9.2.0 is planned to be removed in a future release.
- If you exploit the Connection Optimization (DNS/WLM) feature of BIND 4.9.3, you have to use an alternative solution. 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: On z/OS V1R10 and later systems, use check ZOSMIGV1R10_CS_BIND4 for BIND DNS 4.9.3 and check ZOSMIGV1R11_CS_BIND9 for DNS 9.2.0 to determine if a BIND DNS name server function is in use.

IP Services: Migrate from DNS BIND 9.2.0 (Recommended)
Not required, but recommended because this action is planned to become a requirement in a future release.
In a future release of z/OS, support for BIND DNS 9.2.0 will be discontinued. If you are using this name server, you are encouraged to find a replacement as soon as you can.
Migration action:
- If you have been using BIND DNS 9.2.0 as a caching-only name server, use the z/OS resolver DNS caching function in z/OS V1R11 instead.
- If you have been using BIND DNS 9.2.0 as a primary or secondary authoritative name server, migrate to BIND DNS on Linux for System z.

Tip: On z/OS V1R11 and later, use IBM Health Checker for z/OS check ZOSMIGV1R11_CS_BIND9 for DNS 9.2.0 to determine if a BIND DNS name server function is in use.

IP Services: Accommodate DEFAULT route usage changed for CINET environments (Required-IF, as of R11)
Required if your system has been identified as affected (see below). Systems running a single TCP/IP stack will not be affected.
(This Communications Server Flash Alert can be found at: http://www-01.ibm.com/support/docview.wss?uid=swg21410999&myns=z000&mynp=OCSG004CE&mynp=OCSWG90&mync=E)

When running a system with multiple TCP/IP stacks active (Common INET environment), UNIX System Services keeps a table of routing definitions to determine which stack should be used for a new outbound connection (the prerouter function). After upgrading to z/OS 1.11, the potential exists for the stack selection to be different for applications that use the DEFAULT route (also called 'route of last resort', and indicated by a subnet mask of all 0s) than had been selected on earlier releases. This can cause some applications to behave differently, including having connection failures and timeouts.

Details: In releases prior to z/OS 1.11, the DEFAULT route was tracked only as a flag for each stack, indicating that the stack had a DEFAULT route defined. With z/OS 1.11, this has been changed to treat the DEFAULT route like all other subnet and network routes, including the potential for multiple duplicate routes on a stack, having an associated cost, and selection of a stack with the least cost for that route.

Selecting the stack with the least cost path to the requested destination address is generally desirable. However, on systems where multiple stacks have a DEFAULT route defined, these might exhibit different behavior with an upgrade to z/OS 1.11. Where before the existence of these routes was masked by the selection logic used in earlier releases, they can now be selected causing undesirable results (including connection timeouts).

The problem occurs when one or more stacks have a DEFAULT route defined, but they do not actually have network connectivity to get to all addresses. This typically occurs when the devices used by a stack have connectivity only to a subset of the total network (such as an isolated laboratory LAN), and a DEFAULT route is coded to cover associated addresses that are not directly connected to that LAN (require going through a router to other LAN segments). If this is done using a GATEWAY or BEGINROUTES section of the TCPIP PROFILE, this route will be listed with the lowest cost. If the stack that actually does have connectivity to the entire network learns its DEFAULT route using dynamic routing (OMPROUTE), its route will have a higher cost. When this occurs, the lower cost stack will always be selected in z/OS 1.11. If the desired route is on the default stack, it would have always been selected in earlier releases.

Determining whether your system is affected:

1. Examine the BPXPRMxx PARMLIB member, and find the NETWORK statement with DOMAINNAME(AF_INET) specified. If this statement has TYPE(INET) specified, this system will be unaffected by the upgrade. If it has TYPE(CINET), then continue to the next step.

   Alternatively, you can do a DISPLAY OMVS,PFS operator command (see the BPXO046I message for a description of the output). If the PFS TYPE for AF_INET SOCKETS (typically the first or second entry) shows **INET**, your system is configured for a single stack and will not be affected. If it shows **CINET**, continue to the next step.

2. Do a DISPLAY TCPIP operator command with no options specified. It will list the names of the TCPIP stacks that have ever been active since the last IPL. If there is only one name listed, then your system will be unaffected as long as no other stack is started.

3. In the BPXPRMxx member, examine the list of SUBFILESYSTYPE statements with TYPE(CINET). Note which one has the DEFAULT keyword specified. If none of them do, then the new stack selection logic (while potentially different) will not have a significant effect.

   If using the alternative DISPLAY OMVS,PFS output, note the FLAGS in the second section. If one has **CD**, then either no default stack has been specified or the one that is specified is not active (and will have an **SD** flag). If one has **SC**, then continue with the next step.

4. Do a DISPLAY OMVS,CINET operator command (see the BPXO047I message for a description of the output). Note which stacks in the HOME INTERFACE list have the **DRS** flag. If one of them is the default stack (determined above), then continue with the next step.

   If the system is already running z/OS 1.11, then the **DRS** flag will not appear. Instead go to the end of the HOST ROUTE INFORMATION section, where there will be one or more entries with NET DESTINATION 000.000.000.000 and NET MASK 000.000.000.000. Skip step 5 and use the information from these entries for step 6.

5. Display the routing table for each active stack on the system (listed in the output from step 2). This can be done either from TSO (NETSTAT ROUTE DETAIL TCP stackname) or the operator console (DISPLAY TCPIP,stackname,NETSTAT,ROUTE,DETAIL). Note the entries listed with Destination Default, which will be at the beginning of the report.

6. If the default routes listed for the default stack all have a lower METRIC value than those for the other stacks, then your system will be unaffected by the upgrade as long as that does not change.

7. Test the accuracy of the default routes by using the FTP command to connect to a server whose address will not cause selection of any other route listed by step 4. From TSO enter the command FTP -p stackname xx.xx.xx.xx (TIMEOUT 15, first using the default stack (verify the connection works) and then using all other stacks that have a default route. If the command has a timeout making the connection, it is likely that the default route through that stack is invalid.

Migration action:
For systems configured for CINET, review the routing definitions before upgrading to later releases to ensure that nonexistent routes are not listed. To avoid this problem, ensure that the DEFAULT route is not configured on stacks that have limited network connectivity. For those stacks, the correct configuration is to explicitly list the subnets that are directly or indirectly connected to the devices used without coding a DEFAULT route.

When an undesired DEFAULT route is being learned through dynamic routing, identify the source of that route and update the configuration of the router advertising it. For OMPROUTE, the DEFAULT route will be advertised in one of the following conditions:

*   Originate_Default_Route=YES is specified on the AS_Boundary_Routing or IPv6_AS_Boundary_Routing statement when using OSPF
*   The Originate_RIP_Default or IPv6_Originate_RIP_Default statement is coded when using RIP

Remove this specification where this is not appropriate.

Background information:
The stack selection logic performed by OMVS is as follows:

*   The prerouter table affects only connections outbound from the z/OS system when stack affinity has not otherwise been established. Inbound connections are always associated with the stack on which the packets arrived. That is likely why only a subset of applications are affected; the others are servers that service inbound connections.
*   If the application binds to a specific local address, then the stack owning that address is used for that socket from that point forward. Most client (outbound) applications do not do an explicit bind.
*   On the connect() call (or sendto() call for UDP traffic), the requested remote address is used to search the prerouter table. It first finds the best match; a host route for the address (if one exists), then a subnet route with the most specific mask (most bits ON).
*   If multiple stacks list that matching route, the one with the smallest cost (the METRIC value) is used.
*   If multiple stacks have that lowest cost and one of them is the default stack, then the default stack is used. If not, the stack selected will be one of the set (it is not defined which one, probably the first encountered in the prerouter table).
*   If no other route matches, then the default route is used. Here is where the difference between the release levels occurs:
    *   For z/OS 1.10 and earlier, the prerouter table only tracks the existence of a default route on each stack (the DRS flag). If one of those is the DEFAULT stack, then that will be used; otherwise, one of the others is selected.
    *   For z/OS 1.11 (and future releases), the prerouter table lists the default route the same way as it does all other routes (destination 0.0.0.0, mask 0.0.0.0, with a metric and associated stack). So at this point, the rules for selection become the same as described for selection of a subnet route.

IP Services: Modify applications to no longer add IPv6 type 0 routing headers to outgoing packets (Required-IF, as of R11)
Required if you run applications on the TCP/IP stack that build IPv6 type 0 routing headers in outgoing packets.
The IPv6 type 0 routing header is deprecated, as described in RFC 5095 (http://www.ietf.org/rfc/rfc5095.txt). Beginning with z/OS V1R11, an application will receive an error if it attempts to add an IPv6 type 0 routing header to outgoing packets. If your installation is affected by this change, you must update applications that run on the TCP/IP stack to remove the use of the IPV6_RTHDR option to build IPv6 type 0 routing headers. **Note:** The IPv6 type 0 routing header is rarely used.

Migration action:
1. Review RFC 5095 (http://www.ietf.org/rfc/rfc5095.txt) for a description of the need for deprecating the IPv6 type 0 routing header.
2. Identify any applications you run on your TCP/IP stack that use the IPV6_RTHDR option to build IPv6 type 0 routing headers in outgoing packets.
3. For each application identified in Step 2, determine if changes are required in order to continue using the application on a stack that does not support IPv6 type 0 routing headers.
4. For each application that requires changes, complete those changes or discontinue use of the application.

IP Services: Update automation to accommodate FTP output that is changed for extended address volumes (Required-IF, as of R11)
Required if you use FTP to display statistics about the space on any volume or to display information about MVS data sets with a common high-level qualifier, and you process the space statistics or information programmatically.
The output of several FTP subcommands and commands has changed in support of extended address volumes:
- The FTP client LOCSITE and SITE subcommands with the QDISK parameter, and the FTP server SITE command with the QDISK parameter, display statistics about the amount of space available on a volume. In z/OS V1R11, the output format has been changed to accommodate the larger amounts of free space available on extended address volumes.
- The FTP client DIR subcommand and the FTP server LIST command display a list of directory entries or a list of files in a file group on a remote host, or a list of the members of the partitioned data set, as well as auxiliary information about the files. In z/OS V1R11, the output format has been changed to accommodate the larger data sets supported by MVS.

You must change FTP client API programs and automation tools to recognize the new formats. Note that the new formats are used for all volumes.
Migration action:
- Be aware of the changes in messages and replies that FTP uses to report space statistics for volumes:
  o The following messages have changed: EZA2192I, EZA2193I, and EZA2194I.
  o The following replies have changed:
```
200-          Percent Free Free Largest    Free
200- Volume Free    Cyls Trks Cyls--Trks Exts   Use Attr
```
- Be aware of changes in the information the z/OS FTP client reports in message EZA2284I when you use the DIR subcommand to list information about MVS data sets.
- Update any automated processing tools or FTP client API applications to recognize the format of messages and replies issued by FTP to report space statistics for volumes.

IP Services: Update procedures that use the syslogd job name (Required-IF, as of R11)
Required if you have automation or procedures that use the syslogd job name, or if you reserve the syslogd UDP port using the job name.
Before z/OS V1R11, the job name of the syslog daemon (syslogd) was the name of the cataloged procedure plus an additional character (usually the character "1") when syslogd was started from a procedure and when the procedure name was less than 8 characters in length. An additional character was also appended to the program name when syslogd was started from the z/OS UNIX shell. Beginning with z/OS V1R11, the syslogd job name is the same as the procedure name or program name.
Migration action:
- Update any automation or other procedures that use the syslogd job name. If you run two copies of syslogd, be sure to use a unique job name for each copy.
- Update the syslogd UDP port reservation in the TCP/IP profile to use the syslogd procedure name.

IP Services: Accept the new behavior of TCP receive buffer size (Required-IF, as of R11)
Required if you have inbound streaming TCP connections over networks with large bandwidth and high latency.
Before z/OS V1R11, there was no automatic buffer size tuning of inbound streaming TCP connections. Starting in z/OS V1R11, the TCP stack automatically tunes the ideal window size for inbound streaming TCP connections over networks with large bandwidth and high latency. The stack enables the TCP receive buffer size to grow as large as 2 MB for such TCP connections, irrespective of the TCPMAXRCVBUFRSIZE value. The pre-V1R11 behavior is no longer possible; this is a performance enhancement.
Migration action: Update applications that process output from the Netstat ALL/-A command so they can handle the larger values for the reported RcvWnd, ReceiveBufferSize, and ReceiveDataQueued fields.

IP Services: Update IP filter policy to filter IP fragments correctly for RFC 4301 compliance (Required-IF, as of R12)

Required if your IP filter policy selectively matches routed traffic based on TCP port, UDP port, ICMP type and code, ICMPv6 type and code, OSPF type, or MIPv6 type. If you configured the RFC4301Compliance parameter on the IpFilterPolicy statement, it is recommended that you remove the parameter because it is deprecated beginning with z/OS V1R12.

Beginning with z/OS V1R12, all IP security filters must be compliant with RFC 4301. You can no longer use the RFC4301Compliance parameter on the IpFilterPolicy statement to specify whether Policy Agent enforces compliance. The RFC4301Compliance parameter is ignored and Policy Agent enforces the rule that ensures all IP filters are compliant.

IP filter policy support for filtering fragments was 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 with 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. RFC 4301, "Security Architecture for the Internet Protocol", specifies the base architecture for IPSec-compliant systems, including restrictions on the routing of fragmented packets.

Communications Server does not implement stateful fragment checking, therefore, restrictions were added as required by RFC 4301 to require that IP filter rules for routed traffic do not allow specific ports, types, or codes. The configuration parameter RFC4301Compliance could be used in z/OS V1R10 and z/OS V1R11 to optionally configure whether the RFC 4301 rules should be applied. Beginning with z/OS V1R12, this parameter (RFC4301Compliance on the IpFilterPolicy statement) is deprecated. All IP filter rules must support the RFC 4301 rules and restrictions.

Migration action: Follow the steps documented in z/OS R12 Migration.

Tip: Use IBM Health Checker for z/OS check ZOSMIGV1R11_CS_RFC4301 to determine whether IPSec filter rules are compliant with RFC4301. This check is available in z/OS V1R10 with the installation of PTF UA47191 (APAR OA28605).

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.

Communications Server Migration Actions Pre-First IPL

IP Services: Update automation that keys on TN3270E Telnet server messages (Req-IF, as of R12)

Required if you have automation checking Telnet server messages.

Prior to z/OS V1R12, all Telnet messages began with the word "TELNET" after the message identifier. The DEBUG, LUNS, and LUNR messages began with "TELNET *jobname*" after the message identifier. Starting with z/OS V1R12, the first word "TELNET" is replaced with the jobname. For the DEBUG, LUNS, and LUNR messages, all other words in the message are shifted to the left. If you have automation which assumes certain positions for these messages, you must make accommodations to accept the new format of these messages.

Migration action: Ensure that no automation expects the first word of a Telnet message to be "TELNET" and ensure that any automation that keys on the DEBUG message EZZ6035I, or the LUNS and LUNR messages EZZ6091I - EZZ6099I, is updated to handle the word shift to the left.

IP Services: Ensure the TN3270E Telnet server can end automatically when an OMVS shutdown command is issued (Required-IF, as of R12)

Required if you issue an F OMVS,SHUTDOWN command when the Telnet application is active.

Prior to z/OS V1R12, if the Telnet application was active when you issued the F OMVS,SHUTDOWN command, Telnet continued to be active but in an unusable state. Starting with z/OS V1R12, Telnet stops automatically when you issue the F OMVS,SHUTDOWN command. Messages are issued in the order they appear below:

```
BPXI055I OMVS SHUTDOWN REQUEST ACCEPTED
EZZ6008I tnproc STOPPING
EZZ6010I tnproc SERVER ENDED FOR PORT 23
```
EZZ6009I *tnproc* SERVER STOPPED

Migration action: Review your procedures for issuing the F OMVS,SHUTDOWN command and be aware the Telnet server will stop when the command is issued.

IP Services: Disable resolver monitoring of name server responsiveness (Required-IF, as of R12)

Required if you do not want notification of potential problems with name servers in your network.
Prior to z/OS V1R12, the z/OS Communications Server system resolver did not provide notification of situations where a Domain Name System (DNS) name server was not responding to resolver queries sent to the name server. Starting with z/OS V1R12, the resolver will notify the operator when an unresponsive name server is detected. This function provides critical diagnostic information to the network operator and allows the operator to take steps to avoid performance and availability problems. If you do not want the resolver to provide these notifications, you must take action.
Migration action:
- If you do not already have a resolver setup file, create one.
- Code the UNRESPONSIVETHRESHOLD(0) statement in a resolver setup file. A *percentage_of_queries* value of 0 indicates that resolver should not monitor name server responsiveness. The default percentage is 25% of resolver queries.
- If resolver is active, issue the MODIFY *resolver*,REFRESH,SETUP=*resolver_setup_file* command to cause the resolver to stop monitoring name server responsiveness.
- If the resolver is not active, start the resolver.

IP Services: Restore resolver UDP request timeout interval duration (Required-IF, as of R12)

Required if your network requires longer than 5 seconds for resolver queries to be processed by a name server under normal network conditions.
Prior to z/OS V1R12, the default setting for considering UDP requests to a name server to have timed out (as specified by the TCPIP.DATA RESOLVERTIMEOUT statement) was 30 seconds. Starting with z/OS V1R12, the default is 5 seconds. If you want the resolver to continue to use the old value, you must take action.
Migration action:
- Code the RESOLVERTIMEOUT 30 statement in each TCPIP.DATA dataset where the previous UDP timeout duration interval is needed.
- If the resolver is active when all the TCPIP.DATA datasets are updated, issue the MODIFY *resolver*,REFRESH command to cause the resolver to refresh the TCPIP.DATA settings.
- If the resolver is not active, start the resolver.

IP Services: Ensure applications tolerate a larger addrinfo structure (Required-IF, as of R12)

Required if your application has a dependency on the size of the struct addrinfo structures returned by the getaddrinfo() subroutine points.
The addrinfo structure is used by the getaddrinfo() function to hold host address information. In z/OS V1R12, the addrinfo structure (struct addrinfo) is enhanced to comply with RFC 5014. As a result, the struct addrinfo is larger than it was prior to z/OS V1R12. When the res parameter returned by getaddrinfo() is a chain of pointers to struct addrinfo structures, the pointers point to the larger struct addrinfo.
Migration action: Be aware that the res parameter returned by the getaddrinfo() subroutine will point to larger struct addrinfo structures. Change your application as necessary to accommodate larger struct addrinfo structures.

IP Services: Disable IP validation checks when defining key exchange policy rules for a dynamic VPN (Required-IF, as of R12)

Required if you want bypass the IP security key exchange validation check that verifies that the identity of the remote peer matches the IP address of the remote peer.
You can configure policy rules for IP security when defining key exchange policy for dynamic virtual private networks (DVPNs). Prior to z/OS V1R12, there was no automatic way to verify that the identity of the remote peer matched the IP address of the remote peer. Beginning with z/OS V1R12, such IP validation checks are enforced by default. If you do not want the IP validation checking enabled on your policy rules for security key exchanges, you must disable it.
Notes:
1. The new default to enable IP validation checking is a benefit; however, one example of why you might want to disable the checking is if your remote security endpoint is expected to be behind a network address translation (NAT) device.

2. A value coded on the parameter to disable the automatic IP validation checking is ignored if the identity of the peer is not an IPv4 or IPv6 address.

Migration action: Specify the value YES on the BypassIpValidation parameter of the KeyExchangePolicy or KeyExchangeAction statements. (The value of NO is the default).

IP Services: Update modified Netstat message catalogs to include timestamp (Req-IF, as of R12)

Required if you are customizing either the netmsg.cat or netmsg6.cat message catalog.

z/OS V1R12 enhances Netstat to verify that its message catalogs are at the same maintenance level as the Netstat program. If the message catalog time stamp does not match the Netstat program time stamp, Netstat uses the default messages. If you are customizing the netmsg.cat or netmsg6.cat message catalogs, you must take action to preserve the time stamp.

Migration action:

1. Issue the **dspcat** command with the -t -g options against the netmsg.cat (for example, issue **dspcat netmsg.cat -g -t**) to retrieve the necessary output to supply to the **gencat** command to generate your translated message catalog. The z/OS Communications Server netmsg.cat is installed in the z/OS UNIX file system directory /usr/lpp/tcpip/lib/nls/msg/C. The output file displays as follows:

```
The time stamp of catalog netmsg.cat is: 2009 112 00:11 UTC
$delset 1
$set 1
$quote "
```

2. Remove the first line of the output file generated by the **dspcat** command and replace it with a $timestamp directive and the time stamp value from the same output file. The new output file displays as follows:

```
$timestamp 2009 112 00:11 UTC
$delset 1
$set 1
$quote "
```

3. Make any other appropriate changes to customize your message catalog and issue the **gencat** command to generate the new message catalog.

4. Repeat steps 1-3 for the netmsg6.cat message catalog.

IP Services: Release addrinfo storage after resolver thread task terminates (Recommended, as of R12)

Not required, but recommended to prevent excessive consumption of application virtual storage by resolver functions.

The resolver allocates storage in the application address so contain the addrinfo data returned by the getaddrinfo() function. Prior to z/OS V1R12, the resolver released this storage when the task that issued the getaddrinfo() function terminated. With z/OS V1R12, the resolver releases the addrinfo storage when the process task that owns the task that issued the getaddrinfo() function terminates. Applications with long-running tasks that do not explicitly release addrinfo storage using the freeaddrinfo() function might experience an increase in storage usage.

Migration action: Update application source code to ensure that the freeaddrinfo() function is issued to release all addrinfo storage obtained using the getaddrinfo() function.

IP Services: Update syslogd configuration for archiving rules with shared z/OS UNIX file destinations (Required-IF, as of R12 and only if you are coming from z/OS R11)

Required if you configure multiple syslogd rules that use the same z/OS UNIX file destination, and if you configure automatic archiving for that destination file.

Starting with z/OS V1R11, you can configure the syslogd daemon (syslogd) to perform automatic archiving for z/OS UNIX files. You do this by specifying the -N parameter on the appropriate syslogd rules in the configuration file, along with specific configuration statements for the archive function. Beginning with z/OS V1R12, if you use the same z/OS UNIX file destination on multiple syslogd rules, the automatic archiving function is not supported on any of these rules.

Migration action: If you configure multiple syslogd rules that use the same z/OS UNIX file destination, and configure automatic archiving for that destination file, perform the following steps:

- If you do not require automatic archiving, remove the -N parameter from all rules that share the same z/OS UNIX file destination.
- If you do require automatic archiving, change the destination to a unique z/OS UNIX file for each rule that shares the same z/OS UNIX file destination.

Note: If you do not perform either of these steps, message FSUM1273 *jobname* AUTOMATIC ARCHIVE NOT USED FOR RULES WITH SHARED DESTINATION is issued to the operator console, and one or more instances of message FSUM1272 WARNING: ARCHIVE FUNCTION DISABLED FOR RULES WITH SHARED DESTINATION *filename* is written to the syslogd error destination. The automatic archiving function is turned off for all syslogd rules

that share z/OS UNIX file destinations. Although automatic archiving is turned off, syslogd continues to log messages to the destination file.

IP Services: Allow usage of the full range of ephemeral ports (Required-IF, as of R11, and back to R9 with PK66387)
Required if you have security definitions in your network that restrict ephemeral ports to various ranges.
Before z/OS V1R11, TCP/IP used the lowest available ephemeral ports when establishing new connections. Starting in z/OS V1R11, or in V1R10 or V1R9 with APAR PK66387, TCP/IP uses the full range of available ephemeral port values (1025 -65535) beginning with the lowest port before an available port is reused. You must take action if you have security definitions in your network that restrict ephemeral ports to various ranges.
Migration action: Do one of the following:
- Change the definitions in your network to allow the full range of ephemeral ports. The range of allowed values is 1025 -65535.
- Modify the TCPIP profile to restrict the ephemeral ports used so that the values of the ephemeral ports match the values that are allowed in the network. For example, to exclude the use of ports 6666-6669 for both TCP and UDP applications, you can add the following PORTRANGE statements to the TCPIP profile:
```
PORTRANGE 6666 4 TCP RESERVED
PORTRANGE 6666 4 UDP RESERVED
```

IP Services: Run the syslogd daemon in the background when starting from the UNIX shell (Required-IF, as of R11)
Required if syslogd is started from the UNIX shell.
Starting in z/OS V1R11, the syslog daemon (syslogd) does not automatically run in the background when it is started from the UNIX shell. As a result, you need to include a trailing ampersand character (&) on the syslogd command line to run it in the background. If you do not include a trailing ampersand, control will not return to your shell session until syslogd ends. If syslogd is started from a shell script, in particular /etc/rc, this can cause the shell to hang, waiting for syslogd to end. For example, if you run syslogd from/etc/rc and you do not change syslogd to run in the background, OMVS will not complete initialization until OMVS times out.
Migration action: Determine how syslogd is started:
- If it is started from /etc/rc or another shell script, or from BPXBATCH, add a trailing ampersand character to the command line.
- If you start syslogd manually from the UNIX shell, include a trailing ampersand on the command line.
- If you start syslogd from a cataloged procedure that uses BPXBATCH, add a trailing ampersand character to the command line, and include a sleep command in your script after starting syslogd to give syslogd time to initialize before the shell script ends. Also, if you start syslogd from a cataloged procedure that uses BPXBATCH, you must use an HFS file for the STDOUT DD and STDERR DD or the job will not end.
- If you start syslogd from a cataloged procedure that does not use BPXBATCH, no changes are needed.

IP Services: Accept the new resolver behavior of saving the results of DNS queries (Recommended, as of R11)
Not required, but recommended if you are using the resolver to query DNS servers because the new behavior can eliminate some setup, monitoring, and administration steps.
Before z/OS V1R11, the z/OS system resolver did not cache results of domain name server (DNS) queries. Starting in z/OS V1R11, the resolver automatically saves the results of such queries. This is a performance and usability enhancement and can eliminate some setup, monitoring, and administration steps. However, if you want to keep the behavior of past releases and disable the resolver DNS caching function, you must take action.
Migration action: Do one of the following:
- If you want to accept the new behavior and cache the results of previous DNS queries, no action is necessary. Optionally, you can code the CACHE resolver setup statement.
- If you have a caching-only name server defined locally, you can choose one of the following steps:
  - If you want to accept the new behavior and allow the resolver to cache results of previous DNS queries, you can remove the caching-only name server from your system, saving operation and configuration costs. Alternatively, you may choose to use resolver system-wide caching to maintain the DNS A, AAAA, and PTR record information to obtain performance improvements for those searches, but continue to use the caching-only name server for the other DNS record types.
    Guideline: If the local caching-only name server is the only name server in the NSINTERADDR list of name servers to be contacted, replace the caching-only name server entry with one or more

name server IP addresses to be contacted. If there is already more than one name server in the NSINTERADDR list of name servers, simply delete the IP address of the local caching-only name server.
- o If you want to keep the existing behavior and want only the caching-only name server to cache results of previous DNS queries, you must code the NOCACHE resolver setup statement. This option might be attractive to you if a significantly large percentage of the information cached by the caching-only name server involves DNS records other than the DNS A, AAAA, and PTR records that the resolver will cache.
- If you want to selectively limit the caching function, you must code the TCPIP.DATA NOCACHE statement in the TCPIP.DATA files used by applications that are not permitted to utilize the caching function.

IP Services: Make changes for Netstat enhancements (Required-IF, as of R11)
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
BIND	/usr/lpp/tcpip/samples/IBM/EZANSBOO	/etc/named.boot	In V1R11, BIND DNS 4.9.3 function was removed from z/OS.
BIND	/usr/lpp/tcpip/samples/IBM/EZANSCH4	/etc/caching.boot	In V1R11, BIND DNS 4.9.3 function was removed from z/OS.
BIND	/usr/lpp/tcpip/samples/IBM/EZANSSL4	/etc/slave.boot	In V1R11, BIND DNS 4.9.3 function was removed from z/OS.
BIND	/usr/lpp/tcpip/samples/IBM/EZATDDSD	/etc/dhcpsd.cfg	In V1R11, BIND DNS 4.9.3 function was removed from z/OS.
BIND	/usr/lpp/tcpip/samples/IBM/EZATDLSS	/etc/binlsd.cfg	In V1R11, BIND DNS 4.9.3 function was removed from z/OS.

Internet Key Exchange Daemon (IKED)	/usr/lpp/tcpip/samples/iked.conf	/etc/security/iked.conf	In z/OS V1R12, a new configuration statement is provided to enable FIPS 140 mode for the IKE daemon.
Network Security Services Server Daemon (NSSD)	/usr/lpp/tcpip/samples/nssd.conf	/etc/security/nssd.conf	In z/OS V1R12, new configuration statements are provided to configure new functions of the IPSec discipline, including FIPS 140 mode and HTTP retrieval and caching of certificates, certificate bundles, and certificate revocation lists.
Policy Agent	/usr/lpp/tcpip/samples/pagent_CommonIPSec.conf	N/A	In z/OS V1R12, new IPSec configuration options enable the use of IKEv2, and of new cryptographic algorithms for IPSec traffic.
Policy Agent	/usr/lpp/tcpip/samples/pagent.conf	/etc/pagent.conf	In z/OS V1R11, new configuration statements are provided to support automatic starting, stopping, and monitoring for a selected set of applications. In z/OS V1R12, the RFC4301Compliance parameter is deprecated on the IpFilterPolicy statement.
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.
syslog daemon	/usr/lpp/tcpip/samples/syslog.conf	/etc/syslog.conf	In z/OS V1R11, new configuration statements are provided to support automatic archiving of z/OS UNIX files.
z/OS UNIX Telnet server (otelnetd)	None	/etc/otelnetd.banner	In z/OS V1R11, this file was added to display a banner page, prior to the login prompt, when a user connects to the z/OS UNIX Telnet server (otelnetd).

SNA Services: Ensure VTAMSG2 in not used in your VTAMLST definitions (Required-IF, as R12)

Required if you have used the name VTAMSG2 in any of your VTAM definition statements.

Starting in z/OS V1R12, VTAMSG2 is a new reserved name of an internal major node built by VTAM. As such, there should be no other use of the name VTAMSG2 in the VTAM definitions that are used by your system.

Migration action: Investigate all your VTAM definitions. If you find the name VTAMSG2 in any of them, change the name to a non-reserved name. Note that if you continue to configure and try to activate VTAMSG2, message `IST607I VARY ACT FOR VTAMSG2 FAILED – INVALID NODE TYPE OR STATE` will be issued.

SNA Services: Disable Enterprise Extender connection health verification (Required-IF, as R12)

Required if you used fewer than five ports to start an EE connection in a previous release, you must code the start option EEVERIFY=NEVER before you start VTAM.
Starting in z/OS V1R12, VTAM verifies the health of the Enterprise Extender (EE) connection while the connection is being activated. VTAM sends a Logical Data Link Control (LDLC) probe to the remote partner using all five ports during the activation. If the partner is not reachable by any port for any reason, VTAM does not start the EE connection. VTAM issues an error message. If you do not want VTAM to verify the health of the EE connection, code the start option EEVERIFY=NEVER before starting VTAM.
Migration action: If you are using less than five ports for all of your EE connections, then you need to specify the EEVERIFY=NEVER start option. If you have specific EE connections, you need to specify EEVERIFY=NEVER on the PU or GROUP statement for that EE connection.

SNA Services: Code MULTPATH start option when using multipath (Required-IF, as R12)

Required if you want to continue using multipath for EE connections.
Prior to z/OS V1R12, multipath support was enabled by a single switch in the TCP/IP profile for both Enterprise Extender (EE) and TCP/IP traffic. Beginning with z/OS V1R12, multipath support is disabled (by default) for EE connections. If you want to continue to use multipath for your EE connections, you must code the start option MULTPATH=TCPVALUE before starting VTAM.
Migration action: Add MULTPATH=TCPVALUE to the VTAM start option file ATCSTR*xx*.

SNA Services: Ensure that dump data sets are large enough to include VTAM internal trace data space (Required, as of R11)

Before z/OS V1R11, VTAM INOP dump processing did not capture the VTAM internal trace (VIT) data space in the dump. Starting in z/OS V1R11, VTAM INOP dump processing automatically captures the VIT data space in the dump when the VIT data space is in use. This provides more information for problem determination in INOP dumps. In order to continue to capture all INOP dump information, you must ensure that your dump data set allocation will accommodate the additional space required for the VIT data space. The maximum possible size of the VIT data space, 50 MB, should be taken into consideration in your dump data set allocation procedure.
Migration action:
1. Be aware that if the VIT data space is active, VTAM INOP dumps will automatically capture the VIT data space (ISTITDS1) to provide more information in the dump for problem determination. This can result in larger INOP dumps.
2. Review the size of your current dump data sets and ensure that they are large enough to accommodate the maximum size of the VIT data space.
    - Because VIT data space tracing is not new in z/OS V1R11, your dump data sets might already be large enough for INOP dumps. VTAM SVC dumps have always dumped the VIT data space. These include dumps taken by ABEND recovery routines (FRRs and ESTAE routines) and any dumps taken with the F *vtamproc*,CSDUMP command. If your system can currently capture a complete dump of VTAM, no changes are required.
    - If current VTAM dumps are incomplete, modify your dump data set allocation to accommodate the maximum size of the VIT data space.
3. VIT data space tracing is important to capture sufficient trace data for VTAM problem determination. With VIT data space tracing enabled at all times, you might be able to resolve VTAM problems on the first failure, preventing the need for you to recreate the problem, or wait for the problem to occur again, to get additional documentation. To ensure that you always have the maximum amount of VIT trace data available for problem diagnosis, do one of the following:
    - Activate the VIT data space at VTAM start by coding DSPSIZE=5 on any TRACE,TYPE=VTAM,MODE=INT start option included in the VTAM start list or on the VTAM start command.
    - If the VIT data space is not activated at VTAM start, activate it with the MODIFY *vtamproc*,TRACE,TYPE=VTAM,DSPSIZE=5 command.

Tip: Use IBM Health Checker for z/OS check CSVTAM_VIT_DSPSIZE to determine whether the VIT dataspace table size is set to 5 (50 MB).

SNA Services: Increase region size for VTAM (Required-IF, as of R11)

Required if you have large numbers of HPR connections and VTAM is using a high percentage of its allocated region size.
ECSA usage for HPR has been reduced. To accomplish this reduction, some growth in VTAM private storage usage was required. The amount of VTAM private storage increase will depend upon the number of RTP endpoints that are supported on the LPAR. If VTAM is using nearly all of its allotted region size, you will need to increase it.
Migration action:
1. Determine the maximum number of RTP pipes that have endpoints in this VTAM.
2. Increase the region size for VTAM by 1.5 MB for each 1,000 RTP pipes.

Communications Server Migration Actions Post-First IPL

IP Services: Ensure that preference values associated with IPv6 router advertisement routes are as expected (Required, as of R12)

Required if your stack is IPv6 enabled and your adjacent routers are originating IPv6 router advertisement messages that contain preference values other than the default value (medium) for either the default route or an indirect prefix route.
Starting in z/OS V1R12, TCP/IP processes preference values that are provided in IPv6 router advertisement messages. Prior to z/OS V1R12, all router advertisement routes were generated with a medium preference value.
Migration action:
1. Review the reference information for a description of how z/OS Communications Server uses the preference values that are received in IPv6 router advertisement messages.
2. Ensure that the preference values associated with the router advertisement routes generated from information received in router advertisement messages are as expected.
3. If any preference values differ from what you expected, make the necessary configuration changes on the advertising routers.

IP Services: Ensure that IKE daemon retransmission settings are compatible (Recommended, as of R11)

Not require, but recommended if you have specified a KeyWait, KeyRetries, DataWait, or DataRetries parameter in your IKED configuration file to control the IKE daemon's retransmission behavior.
Before z/OS V1R11, the following four Internet Key Exchange daemon (IKED) parameters were used to control the IKE daemon's retransmission behavior: KeyWait, KeyRetries, DataWait, DataRetries.
If you used the IBM Configuration Assistant, these parameters' values were set by selecting retry limit and retry interval values for the Phase 1 IKE key negotiation retry tuning or Phase 2 IKE data negotiation retry tuning settings on the Advanced IKE Daemon Settings dialog under the IPSec perspective. The parameters established a fixed retransmission interval and maximum number of retransmissions for a given IKE message.
Starting in z/OS V1R11, the old parameters are ignored and the following two new parameters are used to determine all IKE message retransmission: IkeInitWait and IkeRetries.
Like the old parameters, these new parameters are also set through the Advanced IKE Daemon Settings dialog of the IBM Configuration Assistant. The new parameters enforce a geometrically increasing retransmission interval (each successive interval is twice as long as the one that preceded it). The length of the first interval is specified by IkeInitWait and the number of transmissions is specified by IkeRetries.
You cannot keep the previously-existing behavior of controlling the IKED retransmissions. In most cases, the new default behavior will work well for all of your IKE peers. However, there may be some circumstances where closer control over the retransmission behavior is required (often because of low bandwidth, or slow or unreliable network connections). In these cases, you might need to specify one or both of the new parameters to tailor the behavior. In most cases, the change will be to reduce the allowed number of retransmissions to avoid very long retransmission timeouts.
Migration action:
1. If you generated the IKED configuration file using the IBM Configuration Assistant, check the Advanced IKE Daemon Settings dialog under the IKE Daemon Settings tab under the IPSec perspective for the z/OS image. If any IKE negotiation retry tuning settings are specified, you might have some site - or network-specific retransmission considerations to accommodate. If you configured your IKED configuration file manually, check it to see if any of KeyWait, KeyRetries, DataWait, or DataRetries were specified to tailor the IKE daemon's retransmission behavior. If they were, you might have some site- or network-specific retransmission considerations that must be accommodated.
2. Set the IKE retry tuning values in the IBM Configuration Assistant Advanced IKE Daemon Settings dialog or, if you manually edit your IKED configuration file, specify the IkeInitWait and/or IkeRetries parameters to accommodate the site- or network-specific considerations.

SNA Services: Create a new APPN topology checkpoint data set (Required-IF, as of R11)
Required if you checkpoint APPN topology and restore it with the INITDB=TOPO or INITDB=ALL start option.
In z/OS V1R11 the format of the records in the APPN topology checkpoint data set has changed. A topology checkpoint data set created on a prior release cannot be read by a z/OS V1R11 network node, and a topology checkpoint data set created in z/OS V1R11 cannot be read by a prior release. Therefore, after migrating an APPN network node to z/OS V1R11, you must create a new APPN topology checkpoint data set. When CP-CP sessions are started for the first time with other APPN network nodes, performance might be impacted temporarily because a full topology exchange is required.
Migration action: To create a new APPN topology checkpoint data set after migrating an APPN network node to z/OS V1R11, do the following:
1. Start all desired CP-CP sessions between network nodes in the APPN network and activate desired transmission groups (TGs). Because you cannot retrieve existing APPN topology from a topology checkpoint data set created with a prior release of VTAM when a z/OS V1R11 VTAM is started, a full topology exchange will occur when the z/OS V1R11 network node joins an APPN network for the first time. This happens when the z/OS V1R11 network node activates CP-CP sessions with any adjacent network node in the APPN network. This initial exchange, made necessary by the difference in old and new data set formats, can degrade performance until all of the network topology is transmitted to the z/OS V1R11 network node. After the z/OS V1R11 network node has received all of the network topology from the adjacent network node, there is no further negative performance impact caused by the difference in format of the new topology checkpoint data set, as long as you create a topology checkpoint data set with the new format on the z/OS V1R11 network node after the initial topology exchange. Otherwise, a full topology exchange will be required every time the z/OS V1R11 network node is recycled.
2. When your APPN network is stable, enter the following command to create a topology checkpoint data set in the new format: MODIFY procname,CHKPT,TYPE=TOPO. A checkpoint of APPN topology is also done automatically when you end VTAM with a HALT or HALT QUICK command but not when you end VTAM with a HALT CANCEL command.
3. After the topology checkpoint data set is initially created, you can restore it with INITDB=TOPO or INITDB=ALL when you restart z/OS V1R11 VTAM.

If you start VTAM as a network node with the INITDB=TOPO or INITDB=ALL start option with a version of the APPN topology checkpoint data set that is not recognized, you will receive message `IST1288I TOPOLOGY DATASET RETRIEVAL WAS NOT SUCCESSFUL, CODE = 12`.

DFSMS Migration Actions for z/OS R12 from z/OS R10

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

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

•IMBED, REPLICATE, and KEYRANGE support is **no longer** planned to be removed in the foreseeable future. However, it is still strongly discouraged to use these attributes.

•Determine which VSAM data sets are affected, then schedule a time to redefine them.

•Use tools: IMBDSHIP, and the new z/OS R11 Health Check for catalogs!

•**Planned for z/OS R12!** These attributes will be removed automatically from VSAM data sets logically dumped using DFSMSdss and migrated using DFSMShsm when DFSMSdss is the data mover during restore and recall processing.

• **Migrate from VRSEL(OLD) to VRSEL(NEW) in parmlib member EDGRMMxx (Required-IF, as of R11)**

•In R11, VRSEL(OLD) has been removed, default has been changed to NEW.

•On R10, you'll get a warning message EDG2317E and RC=4 in EDGHSKP processing.

24   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

---

DFSMS Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

• **Modify DFSMSdss applications to handle larger I/O buffers (Required-IF, as of R12)**

•Prior to R12, the DFSMSdss maximum I/O buffer size for writing to and reading from a tape during DUMP, COPYDUMP, and RESTORE was 65,520 bytes.

•As of R12, the maximum is 256K blocks.

•If you have an application that handles I/O buffers with the DFSMSdss API or XMAPI, ensure that is handles up to the new maximum of 262,144 bytes.

• **Remove DFSMShsm user-defined patch to disable or enable use of DFSMSdss cross-memory API (Required-IF, as of R12)**

•Prior to R12, DFSMShsm use of the DFSMSdss cross-memory API could be disabled or enabled with a patch specified in ARCCMDxx.

•As of R12, the patch support has been removed. Use the SETSYS DSSXMMODE command in ARCCMDxx parmlib member instead.

25   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

DFSMS Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Use ALL, DATE, or VERSIONS with (H)BDELETE command and ARCHDEL macro (Required-IF, as of R11)**
  - Prior to R11: you could use (H)BDELETE *dsn* to delete all backup versions of *dsn*.
  - As of R11: you must specify the ALL keyword on this command to delete all backup versions. You must use ALL (a new keyword), DATE, or VERSIONS on the (H)BDELETE command.
  - On z/OS R10: OA29178 allows the ALL keyword for compatibility.

- **Use DFSMSrmm parmlib options instead of usermod to specify how VRSEL processing handles duplicate GDGs (Required-IF, as of R11)**
  - Prior to R11: you could use a usermod to EDGVREC to influence how VRSEL handled duplicate GDGs.
  - As of R11: No more usermod! It will be ignored if you install it on z/OS R11. Use GDG operand on the OPTION command in parmlib member EDGRMMxx on z/OS R11 and higher.

26   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

---

DFSMS Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Update the storage size for jobs that invoke ADRDSSU (Required-IF, as of R11)**
  - Beginning with z/OS V1R11, jobs invoking DFSMSdss ADRDSSU might require a storage size of greater than 2M.
  - Check the JCL for jobs that invoke ADRDSSU. If the REGION parameter specifies a region of 2M or smaller, increase the region size. If the SIZE parameter on the EXEC statement specifies a size of 2M or smaller, increase the size.
  - Check your JES initialization parameters. If the REGION parameter specifies a default region size of 2M or smaller for each job step, you may want to increase this.

- **Permit DEVMAN to call ICKDSF (Required-IF, as of R11)**
  - Starting with z/OS V1R11, support is added to the device manager (DEVMAN) to invoke ICKDSF for the new DEVMAN REFVTOC function. During startup of the DEVMAN address space, ICKDSF is loaded by DEVMAN regardless of whether or not you have enabled the REFVTOC function.
  - If you use program control in RACF to restrict access to ICKDSF, you must ensure the profile which protects the ICKDSF resource in the PROGRAM class allows the user ID associated with the DEVMAN address space to have READ access.
  - Failure to do this means you will not be able to exploit the new REFVTOC function. Note that DEVMAN initialization continues even if DEVMAN does not have sufficient authority.

27   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

DFSMS Migration Actions for z/OS R12 from z/OS R10

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

- **Update operator procedures, system automation, and exit installation or activation for dynamic installation exits in DFSMSrmm (Required-IF, as of R11)**
  - Prior to R11, DFSMSrmm loaded the installation exits during initialization.
  - As of R11, z/OS dynamic exit services are used. Exits will be loaded from LINKLIB during initialization.
  - Replace F DFRMM,REFRESH EXITS command with the dynamic exit commands.
  - If your exits are in LINKLIB, then there is nothing to do. If your exits are not in LINKLIB, then either move them to LINKLIB or use PROGxx or SETPROG to get them loaded.
  - No change to the exit load module names or the exit routines are required.

Migration Actions Post-First IPL:

- **Use indirect zFS file system data set catalog support (Recommended, as of R12)**
  - This may help you with your migration from HFS to zFS, if you want to use indirect catalog entries for your zFS data set.
  - On z/OS R12: zFS data sets can be cataloged (ONLY zFS data sets, no other VSAM LDS).
  - Systems prior to R12 cannot process the indirect catalog entries for zFS data sets, and will fail with a volume not found error.
  - To use this support, do a physical copy of the zFS data set. See *z/OS Planning for Installation* for examples.

DFSMS Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Define facility class names to protect (H)BACKDS RETAINDAYS commands (Required-IF, as of R11)**
  - As of R11: new hsm facility class names protect the BACKDS RETAINDAYS and HBACKDS RETAINDAYS commands. Only available on R11, prior levels will be able to recover backup versions created with the RETAINDAYS keyword. New facility class names must be defined to protect from unintended use of this support by unauthorized users.
  - Define new facility class profiles to authorize to the (H)BACKDS commands:
    - STGADMIN.ARC.BACKDS.RETAINDAYS
    - STGADMIN.ARC.ENDUSER.HBACKDS.RETAINDAYS

DFSMS Migration Actions Between z/OS V1 R10 and z/OS V1 R12

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

DFSMS Migration Actions You Can Do Now

DFSMSdfp: Back up SMS control data sets (Recommended)

Recommended to ensure data integrity.
Migration action: Do the following on your pre-z/OS V1R11 systems:
1. Back up SMS control data sets according to established procedures in the event that fallback is required. The control data set format is VSAM linear.
2. Install all coexistence PTFs defined for z/OS V1R11.

In addition, if you modified and activated a higher-level policy on a pre-z/OS V1R11 system, do the following to ensure that the ACDS can be accessed on z/OS V1R11:
1. On the pre-z/OS V1R11 system, save the active ACDS as an SCDS with the SETSMS SAVESCDS command.
2. On z/OS V1R11, update, translate, validate, and activate the saved SMS policy.

Note: With z/OS V1R11 (and prior releases by APAR OA22373), CDS uplevel checking only checks the formatted release against the updating release level. The construct size change no longer causes warning messages.

DFSMSdfp: Modify exit routines to support 31-bit UCB addresses (Required-IF, as of R12)

Required, if an exit routine does not support 31-bit UCB, it will need to be upgraded to support 31-bit UCB addresses.
Before z/OS V1R12, DADSM captured the unit control block (UCB) address passed to it in its interfaces. Starting in z/OS V1R12, DADSM will no longer capture the UCB address passed to it and will use the actual 31-bit UCB address. The UCB address will either be above or below the 16 MB line. This address in turn will be passed into the DADSM pre- and post- installation exit routines (IGGPRE00, IGGPOST0) in the current 4-byte IEXUCB field. Therefore, the address in IEXUCB may now contain a 31-bit UCB address. Exit routines that do not support 31-bit UCB addresses will need to be upgraded to support a 31-bit UCB address.
Migration action: Update the exit to support a 31-bit UCB address.

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

Recommended to avoid degraded performance and wasted DASD space.
No supported release of z/OS honors the IMBED, REPLICATE, and KEYRANGE attributes for new VSAM data sets. In fact, using these attributes can waste DASD space and often degrades performance. Servicing these VSAM data sets has become increasingly difficult. In some cases, unplanned outages have occurred. For these reasons, IBM recommends that you stop using IMBED and REPLICATE, and that you minimize or eliminate your use of KEYRANGE.
IMBED and REPLICATE were intended as performance improvements and have been rendered obsolete by newer, cached DASD devices. Striped data sets provide much better performance than KEYRANGE and should be viewed as a candidate for any existing KEYRANGE data sets.
Starting in z/OS V1R12, DFSMSdss provides some assistance in identifying and converting data sets with KEYRANGES, IMBED or REPLICATE attributes:
- Data set dump processing and restore processing issue a message `ADR508I (ttt)-mmmmm(yy), THE FOLLOWING DATA SETS REQUIRE SOME ACTION TO BE TAKEN` when data sets with those attributes are encountered.
- Logical restore processing automatically converts indexed VSAM data sets with the IMBED or REPLICATE attributes, but not the KEYRANGES attribute, to key-sequenced data sets (KSDS) that do not have those attributes. It also issues message `ADR507I (ttt)-mmmmm(yy), DATA SET dsn WAS RESTORED WITHOUT THE IMBED OR REPLICATE ATTRIBUTES` indicating that the data set has been converted.

Migration action:
1. Determine which VSAM data sets and ICF catalogs were defined with the IMBED, REPLICATE, or KEYRANGE attribute. Data set dump processing and restore processing issue message `ADR508I (ttt)-mmmmm(yy), THE FOLLOWING DATA SETS REQUIRE SOME ACTION TO BE TAKEN` to indicate data sets with the attributes. To further help you identify the data sets, you can get a tool

that reads existing VSAM data sets and ICF catalogs, and reports which ones have these attributes. The tool is available from the software server (ftp.software.ibm.com) in the s390/mvs/tools directory as IMBDSHIP.JCL.TRSD. Download the file in binary format and unterse it on your z/OS system using AMATERSE or TRSMAIN. Instructions for using the tool are included in the downloaded JCL. **Notes:**

- The tool only checks data sets that are on DASD.
- "AMATERSE" and "TRSMAIN" are names for a service aid that compresses and decompresses data exchanged with IBM. "AMATERSE" is the preferred program name since its integration into z/OS V1R9. "TRSMAIN" is the original program name and is now shipped as an alias entry point to AMATERSE. For more information about AMATERSE, including several differences with TRSMAIN, see *z/OS MVS Diagnosis: Tools and Service Aids*.

2. Schedule a time for the affected VSAM data sets and ICF catalogs to be unavailable, and redefine them. For VSAM data sets you can use JCL similar to the following:

```
//* EXPORT A KSDS
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//INDD DD DSN=EXAMPLE.KSDS,DISP=OLD
//OUTDD DD DSN=EXAMPLE.KSDS.EXPORTED,DISP=(NEW,CATLG),
// SPACE=(CYL,(1,1)),UNIT=SYSDA
//SYSIN DD *
     EXPORT EXAMPLE.KSDS -
     INFILE(INDD) -
     OUTFILE(OUTDD) -
     TEMPORARY
//* NOW IMPORT THE EXPORTED COPY
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//INDD DD DSN=EXAMPLE.KSDS.EXPORTED,DISP=SHR
//SYSIN DD *
     IMPORT -
     INFILE(INDD) -
     OUTDATASET(EXAMPLE.KSDS)
```

|
For ICF catalogs, see informational APAR II13354 for step-by-step instructions on using IDCAMS EXPORT/IMPORT with ICF catalogs..

Tip: Use check CATALOG_IMBED_REPLICATE on z/OS V1R11 to detect IMBED and REPLICATE attributes in your master catalog and any connected user catalogs.

The recommendation to migrate from IMBED, REPLICATE, and KEYRANGE was originally made in the z/OS V1R6 timeframe. In Software Announcement 204-180 (RFA39951), dated August 10, 2004, IBM announced its intent to withdraw support for VSAM IMBED, REPLICATE, and KEYRANGE attributes in a future release. Based on customer feedback, IBM no longer plans to remove this support from z/OS in the foreseeable future. IBM still recommends that you stop using these attributes and plans to remove IMBED and REPLICATE attributes during logical DFSMSdss restore operations and DFSMShsm recall operations as announced in Software Announcement 207-175 4 (RFA45594), dated August 7, 2007. The DFSMSdss function to aid in conversion was added in z/OS V1R12.

DFSMSrmm: Replace CIM providers and CIM classes (Required-IF, as of R12)

Required if you have not yet updated your code to handle the current key formats and are still using the backward-compatible CIM provider (provided as rmmcim19.tar.Z compressed tar archive).
In z/OS V1R10, the keys used for the DFSMSrmm CIM classes were changed. Prior to z/OS V1R12, you had the option of using a backward-compatible CIM provider, rather than updating your code to handle the new key formats. Beginning with z/OS V1R12, the backward-compatible CIM provider is no longer supported. If you have not done so already, you must unregister the previous CIM providers and CIM classes, and register the currently supported CIM providers and CIM classes.
Migration action:
1. Update your code to handle the key formats used in z/OS V1R10 and later. For the current formats, see the table in *z/OS R12 Migration.*

2. Using the rmmutil.sh tool, unregister all the z/OS V1R9 CIM providers and unload all the z/OS V1R9 CIM classes.
3. Using the same rmmutil.sh tool, register the complete set of z/OS V1R12 CIM providers and load the z/OS V1R12 CIM classes.

DFSMSdfp: Define additional volumes to ML1 OVERFLOW and NOOVERFLOW pools for backup and migration (Required-IF, as of R11)

Required if you are migrating from a system that defined ML1 OVERFLOW volumes using the ADDVOL MIGRATION(OVERFLOW) command.

The way that DFSMShsm uses ML1 OVERFLOW volumes has changed. Before z/OS V1R11, volumes that you defined as OVERFLOW volumes using the ADDVOL MIGRATION(OVERFLOW) command were used for backup only for data sets greater than 500 tracks that encountered out of space errors. Beginning with z/OS V1R11, volumes defined with ADDVOL MIGRATION(OVERFLOW) are used by DFSMShsm for both backup and migration of large data sets.

There is also a new command, which determines the size of data sets for which an ML1 OVERFLOW volume is preferred for migration or backup: SETSYS ML1OVERFLOW(DATASETSIZE(*dssize*) THRESHOLD(*threshold*))

The default for DATASETSIZE is 2000000 KB, so by default, DFSMShsm backs up and migrates data sets that are 2000000 KB and larger to ML1 OVERFLOW volumes.

Note that many installations do not use ML1 OVERFLOW volumes. The default setting for ADDVOL MIGRATION is NOOVERFLOW, so if you do not specify the OVERFLOW parameter specifically now, your installation is probably not using ML1 OVERFLOW volumes and this migration action does not apply to you.

Migration action: Define additional volumes to ML1 OVERFLOW and NOOVERFLOW pools as follows:
1. Check to see if you are defining ML1 OVERFLOW volumes using the ADDVOL MIGRATION(OVERFLOW) command.
2. If you are, evaluate what the changed behavior will look like on your installation. See ML1OVERFLOW: Manage ML1 OVERFLOW volumes in *z/OS DFSMShsm Storage Administration* for information on how different size data sets will be backed up and migrated to ML1 OVERFLOW or NOOVERFLOW volumes.
3. Define additional volumes to the ML1 OVERFLOW and NOOVERFLOW pools if needed.
4. If needed, adjust the SETSYS ML1OVERFLOW command settings to change the size of data sets that DFSMShsm will prefer to migrate and back up to ML1 OVERFLOW volumes

DFSMSrmm: Migrate from VRSEL(OLD) to VRSEL(NEW) in parmlib member EDGRMMxx (Required-IF, as of R11)

Required if use the VRSEL(OLD) operand of the OPTION command in DFSMSrmm parmlib member EDGRMMxx.

The VRSEL(OLD) or VRSEL(NEW) operand of the OPTION command in DFSMSrmm parmlib member EDGRMM*xx* specifies the type of vital record processing you want performed. In z/OS V1R11, VRSEL(OLD) has been removed. From z/OS V1R8 to V1R10, if you did not migrate from VRSEL(OLD) to VRSEL(NEW), you received warning message EDG2317E when you ran VRSEL(OLD) processing, and EDGHSKP processing ended with job step return code 4. Beginning with z/OS V1R11, you can no longer specify VRSEL(OLD) in parmlib and the default value is changed to VRSEL(NEW).

Specifying OPTION VRSEL(OLD) will result in error messages from parse:

```
EDG0208I RECORD INPUT:- OPTION VRSEL(OLD)
EDG0209E PARSE MESSAGE:- IKJ56712I INVALID KEYWORD, OLD
EDG0215D ERRORS DETECTED IN INITIALIZATION PARAMETERS
- ENTER ?Y? TO CONTINUE OR ?N? TO CANCEL
```

Reply with "Y" and the default VRSEL(NEW) is used. Reply with "N" and the response is:

```
EDG0104E DFSMSRMM SUBSYSTEM INITIALIZATION FAILED
EDG0107A ENTER SUFFIX OF INITIALIZATION MEMBER OR "CANCEL"
```

Migration action: To migrate from VRSEL(OLD) to VRSEL(NEW), follow the steps in *z/OS V1R10.0 DFSMSrmm Guide and Reference* and information with APAR II14475.

Tip: Use IBM Health Checker for z/OS to determine whether you need to perform this migration action. The relevant check is named ZOSMIGV1R11_RMM_VRSEL_OLD and is available through PTF UA45968 on z/OS V1R9 (APAR OA26947) and PTF UA45967 on z/OS V1R10 (APAR OA26947). The check determines whether you are still using OPTION VRSEL(OLD).

DFSMS Migration Actions Pre-First IPL

DFSMSdfp: Evaluate applications and modify for EAV enhancements (Required-IF, as of R12)

Required if application programs allocated the non-VSAM data sets listed below with the extended attributes option from z/OS V1R11.

Starting with z/OS V1R12, additional non-VSAM data set types in the extended addressing space (EAS) are supported. This includes support for sequential (BASIC or LARGE), partitioned (PDS or PDSE), Catalogs, and BDAM data sets in EAS. Before z/OS V1R12, any program trying to open one of these types of non-VSAM data sets that have been allocated with extended attribute DSCBs (format 8 DSCB from a z/OS V1R12 system) failed with abend IEC144I 313-0C. Now on a z/OS V1R12 system, applications can open these non-VSAM data sets allocated with extended attribute DSCBs with standard BSAM, BPAM and QSAM access.

In many cases, application programs will function as usual. However, application programs that reference the data set extents found in the DEB or access the DSCB or its extents must change to either support format 8 DSCBs and 28-bit cylinder addressing or to abend when they encounter such data sets.

Migration action:
- Look for applications that allocated the following types of non-VSAM data sets in EAS with the extended attributes option from z/OS V1R11:
    - Sequential (BASIC or LARGE)
    - Partitioned (PDS or PDSE)
    - Catalogs
    - BDAM

    Look for applications that allocate these non-VSAM data sets in EAS with the EATTR=OPT data set keyword on JCL or in SMS data class to specify that the data set can have extended attribute DSCBs (format 8/9 DSCBs) and can optionally have extents in the EAS. In z/OS V1R11, the EATTR keyword would have no affect for these non-VSAM data sets until the data set type is enabled in the system for EAS. However, in z/OS V1R12 these data set types are enabled for EAS.

- Affected applications must either support format 8 DSCBs and 28–bit addressing or to issue an abend code to fail processing when they encounter such data sets, because the system will no longer automatically abend with IEC144I 313-0C.

DFSMSdfp: Accommodate new DCBE macro option (Required-IF, as of R12)

Required if you use the XTIOT, UCB nocapture and DSAB-above-the-line options of dynamic allocation, or if any of the callers of your program might have used those options to allocate files which your program will open .

Before z/OS V1R12, the DCB access methods did not support the nocapture UCB option of a dynamic allocation. Starting in z/OS V1R12, the DCB access methods will support a new DCBE option, LOC=ANY, to signify that the application program supports the XTIOT, UCB nocapture and DSAB-above-the-line options of dynamic allocation. The new DCBE option also signifies that the application program has no dependancy on any of XTIOT, UCB address or DSAB-above-the-line options of dynamic allocation.

Migration action:
1. Review, and modify if needed, the installation exit routines that receive a TIOT entry or UCB address directly or indirectly. These include:
    a. DADSM pre- and post-processing exits (IGGPRE00 and IGGPOST0).
    b. Tape management exits (IFG019LA (label anomaly), IFG019VM (volume mount), IFG019FV (file validation), IFG019FS (file start on volume) and IFG055FE (file end on volume)). They receive UCB addresses from TEPMUCB in IFGTEP, from DEBUCBA and from TIOEFSRT. All three sources will allow a 31-bit UCB address. TEPMUCB sometimes will be an uncaptured 31-bit address. If the new DEB31UCB bit is on, the UCB address and modeset byte will be different as described in IEZDEB.
    c. NSL tape exits (NSLOHDRI, NSLEHDRI, NSLOHDRO, NSLEHDRO, NSLETRLI, NSLETRLO, NSLCTRLO, IEFXVNSL and NSLREPOS). OPEN, EOV and CLOSE will always turn DEB31UCB on in the work area to signify that DXDEBUCB is a 31-bit UCB address and the modeset byte is moved as described above. Even though the UCB address field will be four bytes, it typically will still contain a three-byte address.
    d. Volume label editor routines (IFG0193C and IFG0553C). Same work area changes as described for the NSL routines.
    e. DCB OPEN installation exit (IFG0EX0B).
    f. The IGXMSGEX installation exit for the MSGDISP macro already supports its caller passing a 31-bit UCB address but there might be more cases in which this occurs.
    g. The data management ABEND installation exit (IFG0199I) passes a UCB address in field OAIXUCBA. It might now be 31-bit.
    h. IECIEUCB as mapped by the IECIEPRM macro for the ISO/ANSI Version 3 and Version 4 installation exits (IFG0193G) contains the tape UCB address. In the past this has always been a 24-bit addresss. Now it might be a 31-bit address.

The above exits are documented in *z/OS DFSMS Installation Exits*. IEFDB401, a dynamic allocation input validation exit, is documented in *z/OS MVS Installation Exits*.
2. Verify that the installation exit routines will not be affected adversely.
3. Set the NON_VSAM_XTIOT option to YES or NO in the DEVSUP*xx* parmlib member.
4. Change the program to set LOC=ANY in the DCBE macro. The default is LOC=BELOW.

DFSMSdss: Recompile and link-edit exit routines or applications that change options in the ADRUFO block (Required-IF, as of R12)
Required if you change options in the ADRUFO block with either the exit routine or application.
Starting with z/OS V1R12, if you change options in the ADRUFO block with either:
- the options installation exit routine, ADRUIXIT, or
- an application that invokes DFSMSdss with the API or XMAPI and then uses the Presenting ADRUFO Record exit (Eioption exit 13),

you must recompile and link-edit the exit routine or application using macro libraries provided with z/OS V1R12.
Migration action: Recompile and link-edit the exit routine or application using macro libraries provided with release z/OS V1R12.

DFSMSdss: Modify applications to handle larger I/O buffers (Required-IF, as of R12)
Required if the application invokes DFSMSdss with the API or XMAPI.
Starting with z/OS V1R12, DFSMSdss uses BSAM instead of EXCP to read from and write to DFSMSdss dump data sets during DUMP, COPYDUMP, and RESTORE operations. This allows DFSMSdss to support 256K blocks when writing to and reading from a tape. Before z/OS V1R12, the maximum was 65,520 bytes. If you have an application that invokes DFSMSdss with the API or XMAPI, you should ensure that the application handles I/O buffers up to the new maximum of 262,144 bytes. The affected exit options are EIOP03 and EIOP06. For these exits, the storage (buffer) pointed to by EIRECPTR may now be greater than 65,520 bytes, up to a maximum of 262,144 bytes.
Migration action: Ensure that the application can handle I/O buffers that are up to 262,144 bytes. If the application cannot handle I/O buffers that are up to 262,144 bytes, you can:
- Specify PARM='USEEXCP=YES' in OPTPTR of the API or the EXEC PARM with PGM=ADRDSSU (for example, EXEC PGM=ADRDSSU,PARM='USEEXCP=YES'). You can also specify PARM='USEEXCP=YES' in OPTPTR of the API that invokes ADRXMAIA, or specify it on the EXEC PARM with PGM=ADRXMAIA (for example, EXEC PGM=ADRXMAIA,PARM='USEEXCP=YES').
- Set a block size to an acceptable value either with the JCL DD statement or when dynamically allocating the DD for your application during processing of the Presenting ADRUFO Record exit (Eioption exit 13).
- Code BLKSZLIM and set the BLKSZLIM value to 65520. Note that DSS only supports BLKSZLIM of 65520 and above; for a backup to be compatible with releases earlier than z/OS V1R10, the backup must have no larger than 65,520 byte blocks. BLKSZLIM can be set in the following places:
  - o BLKSZLIM keyword on the DD statement or dynamic allocation. The BLKSZLIM keyword on a DD statement keyword is described in *z/OS MVS JCL Reference*.
  - o Block size limit in data class. Set by storage administrator. Available even if the data set is not SMS-managed. The block size in the data class is described in *z/OS DFSMS Implementing System-Managed Storage*.
  - o System default set in TAPEBLKSZLIM keyword in DEVSUP*xx* parmlib member in SYS1.PARMLIB. Also available in DFA. The TAPEBLKSZLIM parameter is described in *z/OS MVS Initialization and Tuning Reference*.
- If the application inspects the DTVBLKSZ field in the volume record of physical dumps or the DTHBLKSZ field in the data set header of logical dumps, you may need to make further changes. When a physical backup is created on tape with a block size that is greater than 65,520 bytes, the DTVBLKSZ field is X'FFFE'. The block size is stored in an extended volume record following the volume record, DTSDEVOL. When a logical backup is created on tape with a block size greater than 65,520 bytes, the DTHBLKSZ field is X'FFFE'. The block size is stored in a new extended tape header record, S, following the data set header record.

Note: Tapes with greater than 65,520 bytes BLKSIZE can be read by z/OS V1R11 and z/OS V1R10 systems, with the PTFs for APAR OA30822 installed.

DFSMShsm: Update applications that depend on LIST command output (Required-IF, as of R12)
Required if your applications depend on the RACF IND field value in the output with the OUTDATASET or SYSOUT destination, or if your applications depend on LIST COPYPOOL(cpname) output.
The DFSMShsm LIST command output changed in z/OS V1R11 and in z/OS V1R12.

- Starting in z/OS V1R11, the LIST DSNAME(*dsname*) BCDS and LIST LEVEL(*hlq*) BCDS output no longer display the RACF IND field when OUTDATASET or SYSOUT (default) is specified as the destination for the output. The RACF IND field is displayed when TERMINAL is specified as the destination for the output.
- Starting in z/OS V1R12, the LIST COPYPOOL(*cpname*) output includes: a new FASTREPLICATION state (FCFRRINCOMPLETE), fast reverse restore status field (FCFRR=), and recovery complete status field (RECOVERYINCOMPLETE=). This new output is displayed when OUTDATASET, SYSOUT, or TERMINAL is specified as the destination for the output.

These changes can affect applications that depend on LIST output.

Migration action:
1. Remove any dependency on the RACF IND field on the LIST DSNAME(*dsname*) BCDS or LIST LEVEL(*hlq*) BCDS output when using OUTDATASET or SYSOUT.
2. Update applications that depend on LIST COPYPOOL(*cpname*) output to handle the new FASTREPLICATION state and new fields.

DFSMShsm: Remove user-defined patch that disables or enables use of the DFSMSdss cross memory API (Required-IF, as of R12)
Required if the ARCCMDxx parmlib member of SYS1.PARMLIB contains: PATCH .MCVT.+433.
Before z/OS V1R12, DFSMShsm use of the DFSMSdss cross memory API could be disabled or enabled by patching the ARCCMD*xx* parmlib member of SYS1.PARMLIB. Starting in z/OS V1R12, support for disabling or enabling DFSMSdss cross memory support through patching ARCCMD*xx* has been removed. Use of DFSMSdss cross memory support can be controlled for backup, CDS backup, dump, migration, full-volume recovery, and data set recovery by including the SETSYS DSSXMMODE command in the ARCCMD*xx* member of SYS1.PARMLIB.
Migration action: Remove PATCH .MCVT.+433 from the ARCCMD*xx* parmlib member of SYS1.PARMLIB. Add the corresponding SETSYS DSSXMMODE command in the ARCCMD*xx* parmlib member of SYS1.PARMLIB.

DFSMShsm: Configure your security system to permit started procedures using new address space identifier (Required-IF, as of R12)
Required if DFSMSdss cross memory support is used for full-volume recovery from dump.
Before z/OS V1R12, when using the DFSMSdss cross memory API, the address space identifier for full-volume and data set recovery from dump was ARC*n*REST. Starting in z/OS V1R12, the address space identifier for full-volume recovery from dump is changed to ARC*n*RST*y* where *n* is the DFSMShsm host ID and *y* is the instance of the DFSMSdss started task (a number 1 - 4). Data set recovery from dump will still use ARC*n*REST. With the addition of multitasking volume recovery from DUMP in z/OS V1R12, the address space names for the DSS cross memory address spaces are separated so that four DSS volume recovery address spaces and only one DSS data set recovery address space can be created. ARC*n*REST is used for data set recovery from dump; ARC*n*RST*y* is used for volume recovery from dump.
Migration action: Before attempting full-volume recovery from dump, configure your security system to permit started procedures using the new address space identifier for DFSMSdss cross memory support for full-volume recovery from dump.

DFSMShsm: Update applications that depend on QUERY COPYPOOL output (Req-IF, as of R12)
Required if you will be using any FlashCopy technique that creates a full-volume FlashCopy relationship or if you will be specifying SETSYS FASTREPLICATION(FCRELATION(FULL)) and your applications depend on QUERY COPYPOOL output or message ARC1820I.
Before z/OS V1R12, the DFSMShsm QUERY COPYPOOL command output did not display FlashCopy "background copy percent-complete" information. Starting in z/OS V1R12, the QUERY COPYPOOL command output, in message ARC1820I, will display applicable "background copy percent-complete" (PCT-COMP) information for full-volume FlashCopy pairs with an incomplete background copy. Percent-complete information (a percentage) is available for full-volume FlashCopy pairs with an incomplete background copy only. A full-volume FlashCopy relationship is established when the FlashCopy technique (such as fast reverse restore or incremental) designates it, or when SETSYS FASTREPLICATION(FCRELATION(FULL)) has been specified. This change can affect applications that depend on QUERY COPYPOOL output.
Migration action: Update applications that depend on QUERY COPYPOOL output.

DFSMSdfp: Evaluate applications for sensitivity to changes to IDCAMS LISTCAT command output (Required-IF, as of R11)
Required if your application depends on the output generated by the IDCAMS LISTCAT ALL command.
Starting with z/OS V1R11, the IDCAMS LISTCAT command output is enhanced to include a new EATTR field indicating whether or not a VSAM data set can be defined with extended attribute DSCBs.

SHARE
Boston, MA
© 2010 IBM Corporation

Page 59 of 94

August 4, 2010

Session 7398

Migration action: Output from the IDCAMS LISTCAT command is not an intended programming interface. Evaluate applications that parse LISTCAT output to ensure that either there is no dependency on the order of the fields in the LISTCAT output or that they can accommodate the new EATTR fields in the output. To reduce future impact and maintenance, IBM suggests migrating the parsing routines or applications to use an IBM provided programming interface such as the Catalog Search Interface. See Catalog Search Interface User's Guide in *z/OS DFSMS Managing Catalogs*. The following example shows the EATTR field in LISTCAT ALL output:

```
LISTCAT LEVEL(ZTAMS002) ALL
NONVSAM ------- ZTAMS002.COMPRS.DATA01
IN-CAT --- SYS1.MVSRES.MASTCAT
HISTORY
DATASET-OWNER-----(NULL) CREATION--------2004.297
RELEASE----------------2 EXPIRATION------0000.000
ACCOUNT-INFO--------------------------------(NULL)
EATTR-------------(OPT)
SMSDATA
STORAGECLASS ---S1P03S12 MANAGEMENTCLASS---(NULL)
DATACLASS ------SRX00001 LBACKUP ---0000.000.0000
VOLUMES
VOLSER-----------1P0301 DEVTYPE------X'3010200F'
```

DFSMSdfp: Accommodate the change of IDCAMS DEFINE for RECORDS involving VSAM linear data sets (Required-IF, as of R11)

Required if you define a linear data set with CISIZE greater than 4096 bytes and specify the RECORDS parameter.
Before z/OS V1R11, when space for linear data sets was calculated in response to an AMS DEFINE command with the RECORDS parameter, an implied control interval size of 4096 was used when the control interval size was larger than 4096. As of z/OS V1R11, in the same scenario, the actual (larger) control interval size is used instead of 4096.
Note: This migration action was previously documented, incorrectly, as being introduced by APAR OA21369 on z/OS V1R9. The correct APAR is FIN APAR OA25988 and the correct release is z/OS V1R11.
Migration action: Consider adjusting the value of the RECORDS specification when a linear data set is defined with CISIZE greater than 4096 bytes. If you continue to use the same IDCAMS DEFINE CLUSTER job in such a situation, the allocated space will be larger in z/OS V1R11 because the actual CISIZE (instead of 4096 bytes) has an influence on the amount of space allocated to the data set. You need to decrease the value of the RECORDS specification to achieve an allocation amount similar to before migration.
Notes:
1. DB2 allocates VSAM linear data sets using the RECORDS parameter, so DB2 allocations would be affected by this change when CISIZE greater than 4 KB is used. To avoid this side effect, apply the PTF for DB2 APAR PK42840. With this PTF, the define and extend code in DB2 has been changed to use KB when issuing the IDCAMS DEFINE command.
2. In z/OS V1R8, SMS added a new code to validate the space quantity for volume selection. The space calculation is based on the number of records multiplied by the actual CISIZE instead of the 4 KB record size. If SMS fails the allocation because the primary quantity requested is larger than the total capacity of the largest available volume, the following error message is issued: IGD17279I (n) VOLUMES WERE REJECTED BECAUSE OF INSUFF TOTAL SPACE.
Reference information: For details about the change, see APAR OA25988.

DFSMSdfp: Permit DEVMAN to call ICKDSF (Required-IF, as of R11)

Required if you use RACF program control to restrict access to ICKDSF.
Starting with z/OS V1R11, support is added to the device manager (DEVMAN) to invoke ICKDSF for the new DEVMAN REFVTOC function. During startup of the DEVMAN address space, ICKDSF is loaded by DEVMAN regardless of whether or not you have enabled the REFVTOC function.
If you use program control in RACF to restrict access to ICKDSF, you must ensure the profile which protects the ICKDSF resource in the PROGRAM class allows the user ID associated with the DEVMAN address space to have READ access.
Message ICH408I will be issued when the DEVMAN address space is started if you use program control for ICKDSF and you have not allowed DEVMAN to call ICKDSF. You will not be able to exploit the new REFVTOC function. Note that DEVMAN initialization continues even if DEVMAN does not have sufficient authority.
Migration action: If you use RACF:
1. Define a user ID for the DEVMAN started task if one is not already established.
2. Define a profile for DEVMAN in the RACF STARTED class. Assign the user ID from Step 1 to the DEVMAN procedure in the STARTED class.

3. Permit the user ID for the DEVMAN started task to call ICKDSF by allowing read access to ICKDSF in the PROGRAM class. For example: `PERMIT ICKDSF CLASS(PROGRAM) ID(userid) ACCESS(READ)`
4. Refresh the storage RACF profiles by using the command `SETR WHEN(PROGRAM) REFRESH.`
5. Refresh the STARTED profiles, if necessary, using the command `SETR RACLIST(STARTED) REFRESH.`

DFSMSdfp: Update the storage size for jobs that invoke ADRDSSU (Required-IF, as of R11)
Required if the amount of storage requested for jobs that invoke ADRDSSU is 2M or smaller.
Beginning with z/OS V1R11, jobs invoking DFSMSdss ADRDSSU might require a storage size of greater than 2M. This could require changes to the JCL for jobs that invoke ADRDSSU, or to JES initialization statements that specify a default region size for jobs.
Migration action:
1. Check the JCL for jobs that invoke ADRDSSU. If the REGION parameter specifies a region of 2M or smaller, increase the region size. If the SIZE parameter on the EXEC statement specifies a size of 2M or smaller, increase the size.
2. **Check your JES initialization parameters. If the REGION parameter specifies a default region size of 2M or smaller, you may want to increase this.**

DFSMSdfp: Specify a control interval size for the IDCAMS DEFINE DATA component (Recommended, as of R11)
Recommended because specifying a control interval size other than 4096 (with the DEFINE command) can improve VSAM performance.
Before z/OS V1R11, IDCAMS DEFINE could be used to specify CONTROLINTERVALSIZE for the CLUSTER component when calculating the space for a data set allocation. If a CONTROLINTERVALSIZE was specified for the DATA component, the specified value was ignored and the CLUSTER CONTORLINTERVALSIZE used. A default value of 4096 is used for the DATA component if a size is not specified for the CLUSTER. Beginning with z/OS V1R11, IDCAMS DEFINE now processes the CONTROLINTERVALSIZE being specified for the DATA component. Existing JCL, which specifies a CONTROLINTERVALSIZE for the DATA component, can result in a different allocation amount when executed on z/OS V1R11.
Note: To preserve consistency in space allocations across releases, specify CONTROLINTERVALSIZE on the CLUSTER level of the DEFINE command and apply PTF UA36500 (APAR OA21369) on systems running z/OS V1R9. Jobs that do not specify a CONTROLINTERVALSIZE for the DATA component are unaffected. Specifying a value other than 4096 can improve VSAM performance.

Migration action: Review production JCL libraries for instances of an IDCAMS DEFINE of a linear data set and specify a control interval size value other than 4096. The actual allocation amount will use the specified control interval size value instead of 4096. Determine if the specified value is appropriate for the data set.

DFSMSdfp: Add columns ODSTATF, ODRETDT, and ODINSTID to OAM object directory tables (Required-IF, as of R11)
Required if you use OAM object support.
In z/OS V1R11, object access method (OAM) introduces 2 GB object support for tape media as well as archive retention enhancements. You must run the CBRSMR1B job to perform the migration from the z/OS V1R10 version of the object storage database to the z/OS V1R11 version. The job will add columns ODSTATF, ODRETDT, and ODINSTID to the existing object directory tables. Even if you do not take advantage of the new function added in this release, if you use OAM object support you must run the CBRSMR1B job to add the new columns to the tables.
Migration action:
1. If your installation's OAM object directory tables in DB2 do not contain ODSTATF, ODRETDT, and ODINSTID columns, edit the sample job CBRSMR1B to make the changes specified in the comments section, and run the job. The job adds the new columns to the OAM object directory tables and primes them ith default values.
2. Run OAM DB2 BIND and GRANT jobs. To determine which BIND and GRANT jobs you need to run, see *z/OS DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

DFSMSdfp: Ensure that the Language Environment run-time library is available for DLLs (Required-IF)
Required if your installation builds or references DLLs.
Language Environment provides common services and language-specific routines in a single run-time environment. You can use Language Environment to build and use dynamic link libraries (DLLs) for applications.

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

DFSMSdss: Build the IPLable stand-alone DFSMSdss image (Required-IF)
Required if you intend to use the Stand-Alone Services provided by DFSMSdss.
If you intend to use the Stand-Alone Services provided by DFSMSdss, you must use the DFSMSdss BUILDSA function to create the Stand-Alone Services IPL-capable core image. Starting with z/OS V1R12, DFSMSdss now uses BSAM instead of EXCP to read from and write to DFSMSdss dump data sets during DUMP, COPYDUMP, and RESTORE operations. To migrate to this support, you must rebuild the IPL-able core image for the Stand-Alone Services program. If this migration action is not performed, users of the DSS standalone restore will not be able to restore backups on tape created with greater than 65520 byte blocks. Message `ADRY3530I SEQUENCE ERROR ON RESTORE TAPE` is issued and the operation is terminated. Backups created with 65520 byte blocks will restore as they did in z/OS V1R11
Migration action:
1. Prepare for Stand-Alone Services by creating a Stand-Alone Services IPL-able core image with the BUILDSA command. With the BUILDSA command you can specify the device (card reader, tape drive, or DASD volume) from which Stand-Alone Services will be IPLed. You can also specify the operator console to be used for Stand-Alone Services. The BUILDSA function builds the IPL-able core image under the current operating system and determines a record size based on whether the IPL is from card, tape, or DASD.
2. Use RACF or another external security system to protect the SYS1.ADR.SAIPLD.Vvolser data set and the Stand-Alone Services modules.
3. If you haven't done so already, make a backup copy of your system that can be restored by this function. For information about backing up volumes, see *z/OS DFSMSdss Storage Administration Guide*.
Note: Message ADRY3530I SEQUENCE ERROR ON RESTORE TAPE might be issued with operation terminated if a user tries to restore a back up that was created with a block size greater than 65520 bytes, using the DSS stand-alone restore program from z/OS V1R10 or z/OS V1R11.

DFSMSdfp: Ensure your SYS1.IMAGELIB is customized for your printing environment (Required-IF)
Required if you are not using your old SYS1.IMAGELIB, you are installing with ServerPac or SystemPac, and you are using line mode printers such as the 3800 or 3900.
If you use line mode printers such as the IBM 3800 or the IBM 3900 running in line mode (not page mode), you must install library character sets, graphic character modification modules, and character arrangement tables in SYS1.IMAGELIB.   This migration action does not apply if you are using IBM 3900 printers that are driven by PSF.
Migration action:
1.   Run the LCSBLD1 job from the samplib data set to create character sets, graphic character modification modules, and character arrangement tables in SYS1.IMAGELIB.
2.   Copy customized or locally-written FCBs and UCS images from your old system's SYS1.IMAGELIB data set to the new system's SYS1.IMAGELIB data set.

DFSMSdfp: Remove user-defined trace points for VSAM record management trace (Required-IF, as of R11)
Required if you have previously added trace points for the VSAM record management trace function.
Before z/OS V1R11, DFSMS included an option to add your own trace points for the VSAM record management trace function. This option has been removed in z/OS V1R11. Therefore, any applications that have user-defined trace points for the VSAM record management trace function need to have them removed.
Migration action: Remove user-defined trace points for the VSAM record management trace function from your applications. If a VSAM trace definition contains any trace point ID that is not listed in *z/OS DFSMSdfp Diagnosis*, remove the user-defined trace point ID from the HOOK parameter in the DD statement. Failure to remove user-defined trace points could cause an ABEND0C1 when running with VSAM record management trace. You might be able to use new IBM-defined trace options in place of a user-defined one. See the list of IBM-defined trace points in *z/OS DFSMSdfp Diagnosis*.

DFSMSdfp: Accommodate changes to DEVSERV command output (Required-IF, as of R11)
Required if you have a dependency on the output of the DEVSERV QDASD or DEVSERV PATHS command.
z/OS V1R10 (and later) includes several changes to output from the DEVSERV command:
- DEVSERV QDASD output is changed as follows: the CYL column was expanded to accommodate a nine-digit number for cylinders. In addition, the extended function consistency check column EF-CHK is reduced to three characters. The new column heading is ECK.

- DEVSERV PATHS output is changed to add the number of cylinders, in a new column labeled CYL. To accommodate the largest possible number of cylinders for a device (nine digits) in this new column, the original ALT column is replaced with this new CYL column and unnecessary spaces around the PIN and DDC columns are removed.
  Note: The display in the header always contains the label "CYL", which replaces the label "ALT". The data in each row for a device always indicates the number of cylinders for a simplex device. For a device with an established or pending dual copy pair, the data in its row for this CYL-labeled column is the alternate device number prefixed by the "ALT" label in its row entry. These labels should satisfy any automation parsing requirements.
- The output for both DEVSERV QDASD and PATHS has been shifted one position to the right, leaving the first character position blank.

Note: In z/OS V1R11, or after the PTF for APAR OA25793 is installed on z/OS V1R10 and z/OS V1R9, DEVSERV PATHS and the QDASD command display 5-digit device numbers that consist of the subchannel set number and the normal device number. The blank that was in the first character position no longer exists and is replaced by 0 or 1 (subchannel set number). The unit field will be changed to "00ddd" from " 0ddd" in case of the 3-digits device number.

Migration action: Change any dependency you have on the output of DEVSERV QDASD or DEVSERV PATHS, as shown in the *z/OS R12 Migration* book.

DFSMSdfp: Verify the version set for ANTQFRVL (Required-IF, as of R11, and back to R10 with OA248009 and OA29249)
Required if ANTRQST REQUEST=QFRVOLS is used.
ANTQFRVL is a parameter list for the Query Fast Replication Volumes (QFRVOLS) request of the ANTRQST API. With the introduction of IBM Remote Pair FlashCopy support, the version field of the parameter list is updated. Prior to this update, the input version was not validated, and an improperly coded program could provide a value of ''b. After the update, without the PTFs for APARs OA24809 and OA29249 installed, a value of ''b for the input version field results in a value of 110 (version, release and level). If you are using the QFRVOLS request of the ANTRQST API, prior to compiling with the updated level of ANTQFRVL, you should ensure that the correct version is being set.
Migration action:
1. For programs that invoke ANTRQST REQUEST=QFRVOLS, verify that the version field is being set in the header.
2. If this field is not being set using the QFRVRLC constant, do not recompile with the updated ANTQFRVL macro; otherwise, a value of ''b will result in a version, release and level of 110 being assumed for ANTQFRVL.

DFSMShsm: Accommodate the change in ARCBDXT exit (Req-IF as of R11 APAR OA28948 and R10 APAR OA28136)
Required if you want to allow command backup but disallow autobackup for some data sets.
Prior to DFSMShsm z/OS V1R10, ARCBDEXT was called during volume-level backup operations (AUTOBACKUP for automatic and BACKVOL for command-based operations). Users examined information in the exit's input data structure to determine whether to allow or disallow backup of a data set. Starting with DFSMShsm z/OS V1R10, ARCBDEXT is called during individual data set backup operations, as well as volume-level backup operations. When called for individual data set backup, the exit's input data structure differs, but the return code values and meanings remain the same as for volume-level backups. Users must examine the level information in the input data structure, at offset x'04' , to determine whether ARCBDEXT is being invoked for volume-level or for individual data set backup.
Migration action: Examine the level information in the input data structure at offset x'04' to determine whether ARCBDEXT is being invoked for volume-level or for individual data set backup. The level field will contain '*EXPAND1' for volume-level requests and '*EXPAND2' for individual data set backup requests. For individual data set backup requests, the field at offset x'0C' contains additional status information about whether the backup request is the result of a backup data set command or the result of retry from a volume command.

DFSMShsm: Define facility class names to protect (H)BACKDS RETAINDAYS commands (Req-IF as of R11)
Required if you use the BACKDS RETAINDAYS and HBACKDS RETAINDAYS commands.
Starting in z/OS V1R11, new DFSMShsm facility class names protect the BACKDS RETAINDAYS and HBACKDS RETAINDAYS commands. The new facility class names must be defined to protect from unintended use of this support by unauthorized users. This support is only available on z/OS V1R11. Earlier systems will be able to recover backup versions created with the RETAINDAYS keyword.
The RETAINDAYS keyword is added to the DFSMShsm (H)BACKDS command. You can use the RETAINDAYS keyword to specify the number of days to retain a specific backup version of a data set.

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

DFSMShsm: Use ALL, DATE, or VERSIONS with the (H)BDELETE command and ARCHBDEL macro (Required-IF, as of R11)

Required if you use the (H)BDELETE command or the ARCHBDEL macro.
Before z/OS V1R11, you could use the (H)BDELETE *dsname* command to delete all backup versions of data set *dsname*. In z/OS V1R11, you must specify the ALL keyword on the (H)BDELETE *dsname* command to delete all backup versions of the data set.
With the PTF for APAR OA29178 installed, z/OS V1R10 and z/OS V1R9 are changed to allow the ALL keyword with HBDELETE, BDELETE, and ARCHBDEL so the command will complete as it does currently, as if the keyword were not present.
Migration action: In z/OS V1R11, you must use the (H)BDELETE command with one of these mutually exclusive keywords: ALL, DATE, or VERSIONS. The new keyword, ALL, specifies that DFSMShsm is to delete all backup versions of the specified data set, including both active and retained copies, except for the retired versions (if they exist). The DATE and TIME keyword deletes active or retained backup versions. The VERSIONS keyword deletes only active backup copies.

DFSMShsm: Maintain coexistence when encountering backup copies with RETAINDAYS value (Required-IF, as of R11)

Required if you are using DFSMShsm functions to process any data set backup copies with a RETAINDAYS value made on z/OS V1R11 DFSMShsm, you must apply toleration APAR OA26327.
In z/OS V1R11, the (H)BACKDS command is enhanced to allow users to specify a RETAINDAYS keyword to create a backup copy with a specified retention period. DFSMShsm maintains backup copies as active backup copies and retained backup copies. Active copies are the backup copies that have not yet rolled off. Retained copies are the backup copies that have rolled off from the active copies, but have not yet reached their retention periods.
Pre-z/OS V1R11 DFSMShsm functions that encounter a backup copy with a RETAINDAYS value, or a retained backup copy made on a z/OS V1R11 system, will have limited processing ability. The functions affected include (H)BACKDS, (H)RECOVER, EXPIREBV, (H)BDELETE, DELVOL, FREEVOL, AUTOBACKUP, RECYCLE, and AUDIT.
Refer to *z/OS R11 Migration* for coexistence considerations.
Migration action: Perform the following functions from a z/OS V1R11 system when data set backup versions are being created on z/OS V1R11 with the RETAINDAYS keyword: (H)BACKDS, (H)RECOVER, EXPIREBV, (H)BDELETE, DELVOL,RECYCLE, AUTO BACKUP, FREEVOL, and AUDIT.

✓ DFSMSrmm: Use DFSMSrmm parmlib options instead of USERMOD to specify how VRSEL processing handles duplicate GDGs (Required-IF, as of R11)

Required if a usermod is used to influence duplicate GDG handling.
Before z/OS V1R11, a USERMOD could be applied to the EDGVREC load module to influence how VRSEL processing handles duplicate generation data groups (GDGs). The USERMOD may be called RMDUPGD. Starting with z/OS V1R11, this USERMOD is no longer supported. You must now use the GDG operand of the OPTION command in parmlib member EDGRMM*xx* to specify how VRSEL processing handles duplicate generations.
Migration action:
1. If a USERMOD is used to influence duplicate GDG handling, do not install it on z/OS V1R11. Remove the USERMOD if already installed on z/OS V1R11. If the USERMOD is retained, it will be ignored and the following warning message will be issued to the MESSAGE file during VRSEL processing: EDGX004I USERMOD HAS BEEN DETECTED AND IGNORED – USE THE GDG(CYCLEBY(GEN) DUP(COUNT)) PARMLIB OPTION INSTEAD.
2. Specify the desired duplicate GDG handling by using the GDG operand of the OPTION command. To obtain consistent VRSEL processing results, you can:
    *   Run VRSEL either on your existing release or exclusively on z/OS V1R11.
    *   Ensure that the GDG parmlib option is specified to match the results obtained from your existing system and run VRSEL on any system. The parmlib option that closely matches the USERMOD is GDG(CYCLEBY(GENERATION) DUPLICATE(COUNT)).

Tip: Use IBM Health Checker for z/OS to determine whether you need to perform this migration action. The relevant check is named ZOSMIGV1R11_RMM_DUPLICATE_GDG and is available through PTF UA45968 on z/OS V1R9

(APAR OA26947) and PTF UA45967 on z/OS V1R10 (APAR OA26947). The check determines whether you have duplicate GDG processing defined.

DFSMSrmm: Update operator procedures and system automation for dynamic installation exits (Required-IF, as of R11)

Required if DFSMSrmm installation exits are in use.
Before z/OS V1R11, installation exits were loaded by DFSMSrmm during initialization. Starting with z/OS V1R11, z/OS dynamic exits services is used to load and activate the default (EDGUXn00) exit modules from LINKLIB during initialization. This change requires changes to DFSMSrmm operating procedures and system automation (if any), and might require changes to exit installation or activation.
Migration action:
* Update operator procedures to replace any mention of F DFRMM,REFRESH EXITS with use of the MVS operator commands that can be used with dynamic exits.
* If you use automation routines, ensure that any automated processing you have for the deleted messages EDG0301I, EDG0302I, EDG0303D, EDG0304I, EDG0305I, and EDG0306I is updated to instead process the EDG0311I through EDG0314I messages.
* If you use DFSMSrmm installation exits and they are:
    * Installed in LINKLIB, you do not need to change them in any way; just install them as you always have. DFSMSrmm automatically exploits the dynamic exit services using your installation exits as the default exit modules at the first startup of the DFRMM procedure.
    * **Not** installed in LINKLIB, you must either move them to LINKLIB or update PROG*xx* or issue the SETPROG command to get them loaded.
    In either case, you are not required to change the load module names used as exit routines with DFSMSrmm exits, although you may change the names if desired. But if you do, you must update PROG*xx* or issue the SETPROG command to get them loaded.
You can now have multiple exit modules associated with each of the exits. Other programs can use the CSVDYNEX macro to associate their exit modules with one or more of the DFSMSrmm exits, and can activate and deactivate as required. You can use the PROG*xx* member of parmlib or the operator SETPROG command to add and activate exit modules for each of the DFSMSrmm exits.

DFSMSrmm: Replace obsolete stem variables in REXX execs (Required-IF, as of R11)

Required if you have written any REXX execs that are used with any of the DFSMSrmm SEARCH or LISTCONTROL subcommands.
Before z/OS V1R11, when DFSMSrmm created REXX variables for SEARCH and LISTCONTROL subcommands, it returned these variables as stem variables and created a .0 stem variable for almost all stem variables. Starting with z/OS V1R11, DFSMSrmm creates a .0 stem variable only for the key variable for each SEARCH and LISTCONTROL command.
Migration action: Use the sample JCL from SAMPLIB (EDGJSTM0) to determine whether your execs must be updated. EDGJSTM0 runs ISRSUPC with the SRCHCMP parameter with SRCHFOR 'stem_variable' process statements to scan REXX exec libraries. If any REXX exec is identified as affected, you must replace the obsolete variable with the appropriate stem 0 variable for the subcommand processed.
Note: Once updated, the execs can be run on any supported release of DFSMSrmm because the key stem variables are always present on prior releases.
Tip: Use check ZOSMIGV1R11_RMM_REXX_STEM to determine if you need to perform this migration action. This check is available through PTF UA45967 on z/OS V1R10 (APAR OA26947). The check determines whether installation-written REXX execs that issue DFSMSrmm TSO/E subcommands use stem variables that are removed in z/OS V1R11 systems. To get the extended function described in this migration action, PTFs for APAR OA32028 need to be installed.

DFSMS Migration Actions Post-First IPL

DFSMSdfp: Run OAM DB2 BIND jobs (Required-IF)
Required if you use OAM object support.
When migrating to any new release of z/OS, you must run OAM DB2 BIND jobs if you are using OAM for object support. The BIND jobs update DB2 with new OAM DB2 code.
Migration action: Run the BIND jobs appropriate to your installation:
1. Update and execute the samplib job CBRPBIND (OAM DB2 Bind Package Job).
2. Do one of the following:

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

Note: The following list itemizes the changes to OAM BIND jobs in z/OS V1R10. If you choose to edit a previous version, you must incorporate any new changes as described in the header of each new samplib BIND job:
- CBRPBIND: CBRIEDBS added
- CBRABIND: CBRIEDBS added to CBRIDBS plan
- CBRHBIND: no changes
- CBRIBIND: CBRIEDBS added to CBRIDBS plan.

DFSMSdfp: Use indirect zFS file system data set catalog support (Recommended, as of R12)
Not required, but recommended to make deployment easier for zFS file system data sets.
Starting in z/OS V1R12, zFS file systems may be cataloged using a system symbol. This allows zFS file system data sets to be indirectly cataloged the same way as non-VSAM data sets.
Systems prior to z/OS V1R12 cannot process the indirectly-cataloged zFS file system data sets and will fail with volume not found errors. This support is limited to zFS file system data sets only. That is, all VSAM linear data sets are not included in this support; only data sets formatted as zFS file systems are included in this support.
Migration action: Use the reference information below to use this new support.
Reference information:
- For setting up the indirect catalog entry, see "Define Cluster" in *z/OS DFSMS Access Method Services for Catalogs*.
- For information about the steps required to establish an indirect catalog entry for zFS file system data sets and what the IDCAMS LISTCAT output will produce, see *z/OS DFSMS Access Method Services for Catalogs*.
- For cloning processes to use this support, see *z/OS Planning for Installation*.

DFSMSdss: Enable or disable the Catalog Search Interface (Required-IF, as of R11, and back to R10 with OA25644)
Required if you prefer the old method of finding cataloged data sets.
Before z/OS V1R11, DFSMSdss would use generic catalog locates to find cataloged data sets based on generic filter criteria on the INCLUDE keyword when no input volumes are specified. In z/OS V1R11, DFSMSdss is changed to use the Catalog Search Interface (CSI) to find cataloged data sets based on the generic filter criteria on the INCLUDE keyword when no input volumes are specified. Prior to z/OS V1R11, you could make use of this functionality on z/OS V1R10, z/OS V1R9, and z/OS V1R8 systems by installing the PTF for APAR OA25644 and patching the offset X'54' into the ADRPATCH module to X'11'.
Migration action:
- To use the new function of DFSMSdss using the Catalog Search Interface, no action is needed.
- If the old method of finding cataloged data sets is desired, a DFSMSdss patch byte can be set. The DFSMSdss patch byte at offset X'54' can be set to any value other than X'00' and X'11' to disable the new functionality. This patch byte can be set using the DFSMSdss SET PATCH command or submitting the AMASPZAP job.
Note: If you currently make use of this functionality on z/OS V1R10, z/OS V1R9, or z/OS V1R8 systems by installing the PTF for APAR OA25644 and patching the offset X'54' into the ADRPATCH module to X'11', you have two choices to expect the functionality be effective in z/OS V1R11.
1. Do nothing, because the patch continues to select the new feature even in z/OS V1R11.
2. Remove the patch or reset to X'00' and use the default behavior in z/OS V1R11.

Slide 30

DFS Migration Actions for z/OS R12 from z/OS R10

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

  - **Ensure that sysplex_admin_level=2 is available on all systems in shared file system environment** (Required-IF, as of R11)

    - Before starting R11/R12 zFS you **MUST** specify this level in your IOEFSPRM file on your R10 zFS systems, otherwise zFS will **not** initialize. This level is found in OA25026 (and an IPL with this APAR will put you at level 1 by default).
    - **sysplex_admin_level = 0** : no APAR OA25026 installed, can only coexist with level 1
    - **sysplex_admin_level = 1** : APAR OA25026 installed, default, can only coexist with level 0, OR level 2, but not both concurrently.
    - **sysplex_admin_level = 2** : APAR OA25026 installed AND IOEFSPRM updated, can only coexist with level 1, OR z/OS R11/R12, but not both concurrently. *Required before migrating to z/OS R12!*
    - **sysplex_admin_level = 3** : z/OS R11/R12 level, can only coexist with level 2.

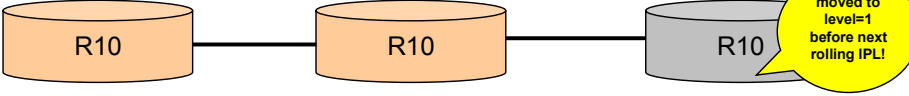30 | Migrating to z/OS 1.12: Part 2 of 2 | © 2010 IBM Corporation

Slide 31

DFS Migration Actions for z/OS R12 from z/OS R10
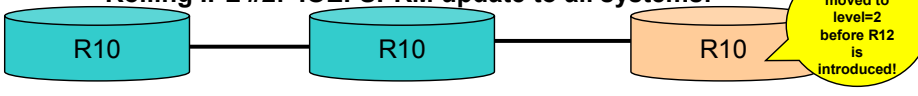
EXAMPLE: z/OS R10 without APAR OA25026 installed:

R10 — R10 — R10

sysplex_admin_level=0   sysplex_admin_level=0   sysplex_admin_level=0

Rolling IPL #1: roll APAR OA25026 to all systems:

R10 — R10 — R10   *(Must be moved to level=1 before next rolling IPL!)*

sysplex_admin_level=1   sysplex_admin_level=1   sysplex_admin_level=0

Rolling IPL #2: IOEFSPRM update to all systems:

R10 — R10 — R10   *(Must be moved to level=2 before R12 is introduced!)*

sysplex_admin_level=2   sysplex_admin_level=2   sysplex_admin_level=1

Rolling IPL with z/OS R12 introduced:

R12 — R12 — R10

sysplex_admin_level=3   sysplex_admin_level=3   sysplex_admin_level=2

31 | Migrating to z/OS 1.12: Part 2 of 2 | © 2010 IBM Corporation

DFS Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Modify programs or procedures that issue an unmount of zFS file systems (Required-IF, as of R11)**
  - Prior to sysplex_admin_level=2: zFS unmount would succeed on a quiesced, growing, or cloning zFS file system.
  - As of sysplex_admin_level=2: zFS unmount fails on a quiesced, growing, or cloning zFS file system, unless unmounted with force.
  - Handle the failure on an unmount, or issue unmount with the force option.

- **Permit the SMB server ID to BPX.DAEMON and BPX.SERVER (Required, as of R11)**
  - As of R11, SMB server must have READ access to these FACILITY class profiles. Otherwise SMB startup will fail.

32 | Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

---

DFS Migration Actions Between z/OS V1 R10 and z/OS V1 R12

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.*

DFS Migration Actions You Can Do Now

zFS: Discontinue use of multi-file system aggregates (Required-IF, as of R11)

Required if you are using zFS multi-file system aggregates.
Beginning in z/OS V1R11, you can no longer attach zFS multi-file system aggregates that are shared across systems in a sysplex. IBM has previously recommended that multi-file system aggregates not be shared in a sysplex environment. Any attempts to attach zFS multi-file system aggregates will fail in a z/OS UNIX shared file system environment with message IOEZ00046E. Attaching zFS compatibility mode aggregates, which have a single file system per data set, will continue to be supported in all environments. IOEZ00046E Error *ErrorCode* received while attaching aggregate *Name*

In a future release, IBM plans to withdraw support for zFS multi-file system aggregates. When this support is withdrawn, only zFS compatibility mode aggregates will be supported.
Migration action: Determine whether you are using zFS multi-file system aggregates by scanning your zFS IOEFSPRM configuration options file for define_aggr statements. You can also scan your /etc/rc file for any **zfsadm attach** commands. If you are using zFS multi-file system aggregates, copy the data from each file system into its own zFS compatibility mode aggregate.
Tip: Use check ZOSMIGV1R11_ZFS_RM_MULTIFS (or ZOSMIGREC_ZFS_RM_MULTIFS in a single-system or monoplex environment) to determine if you have any multi-file system aggregates attached on your system. This check is available for use on z/OS V1R10 systems by installing PTF UA49072 and PTF UA49073 (APAR OA27198).

zFS: Ensure that sysplex_admin_level=2 is available on all systems in a shared file system environment (Required-IF, as of R11)
Required if you have a shared file system environment with more than one system in that environment.
z/OS V1R11 adds a new XCF protocol between sysplex members for zFS administration functions. The new protocol requires the installation of a coexistence PTF on coexisting systems so that zFS can function properly in z/OS V1R11 in a shared file system environment.

If you have a problem on a z/OS V1R10 system running with zFS sysplex_admin_level=2, you can remove the specification (or specify sysplex_admin_level=1, which is equivalent to the default) and perform a rolling IPL or restart zFS on each system. This cannot be done after zFS on a z/OS V1R11 system has joined the sysplex. Also, if you try to start zFS in z/OS V1R11 on another system after you have backed out to sysplex_admin_level=1 on zFS z/OS V1R10, zFS on z/OS V1R11 will not come up because it requires all other systems to be at zFS sysplex_admin_level=2. If you try to bring in zFS z/OS V1R11 when syslex_admin_level=2 is not active on all systems, the message "`IOEZ00614A zFS has detected an incompatible interface level IntLevel for member Sysname`" will appear. (*Intlevel* will be 1 if APAR OA25026 is installed; 0 if it is not.)

The new zFS toleration support uses additional ENQs to keep track of zFS ownership of zFS aggregates. The new ENQs all have a qname of SYSIOEZ. The behavior of unmount changes for any zFS file systems that are owned on a system running with sysplex_admin_level=2.

Migration action:
1. Install the PTF for APAR OA25026 on all V1R10 systems. This is a conditioning function for zFS in z/OS V1R11. Make the PTF available on all systems through a rolling IPL. You are now running with zFS sysplex_admin_level=1.
2. Specify the sysplex_admin_level=2 configuration option in your IOEFSPRM file and make this level available on all z/OS V1R10 zFS systems through another rolling IPL or by restarting zFS. This allows zFS on z/OS V1R10 to tolerate running with zFS on z/OS V1R11. (The default for sysplex_admin_level is sysplex_admin_level=1).

Notes:
1. You cannot specify the sysplex_admin_level configuration option dynamically through the **zfsadm config** command.
2. You cannot skip Step 1 and do only Step 2 in "Steps to take" above if you are running z/OS V1R10 on any systems that do not have the PTF applied and are activated by IPL.

When you configure sysplex_admin_level=1 or 2, the MODIFY ZFS,QUERY,LEVEL operator command returns a new line, at the end of the output, that shows the sysplex admin interface level. The following example shows z/OS V1R10 displaying interface level 1:
```
f zfs,query,level
IOEZ00639I zFS kernel: z/OS zSeries File System
Version 01.10.00 Service Level OAnnnnn - HZFS3A0.
Created on Tue Jan 6 20:02:25 EST 2009.
sysplex(admin-only) interface(1)
```
When you run zFS on z/OS V1R11, the sysplex_admin_level option is ignored and the MODIFY ZFS,QUERY,LEVEL operator command displays interface level 3. When zFS initializes in a shared file system environment, the following message is displayed (as of z/OS V1R10): IOEZ00617I zFS is running sysplex admin-only with interface level 1. This is normal. It shows the sysplex admin interface level that zFS is running.

Tip: Use IBM Health Checker for z/OS to help determine whether you are running zFS at the correct interface level. You need level 2 in order to start zFS on z/OS V1R11. The check is available through APAR OA27198 and is named IBMZFS,ZOSMIGV1R11_ZFS_INTERFACELEVEL.

DFS Migration Actions Pre-First IPL
zFS: Modify programs, execs, or procedures that issue an unmount of zFS file systems (Required-IF, as of R11, and APAR OA25026 with sysplex_admin_level=2)
Required if you have programs that assume that unmount is successful.
Beginning with z/OS V1R11, and for z/OS V1R10 systems that are running zFS with sysplex_admin_level=2, zFS fails an unmount of a quiesced, growing, or cloning zFS file system unless unmount force is specified. In previous releases (without sysplex_admin_level=2), zFS successfully returned in these cases.

Here is an example of the failure message that is issued when a file system is quiesced:
```
unmount filesystem(plex.jms.aggr006.lds0006)
RETURN CODE 00000072, REASON CODE EF0969B5. THE UNMOUNT FAILED FOR FILE SYSTEM
PLEX.JMS.AGGR006.LDS0006.
```
Return code X'72'' indicates that the file system is busy with an administration command.
Migration action: Check your programs, execs and procedures to see if they can handle a failure on unmount. If they cannot, modify them to handle a possible unmount failure or to issue unmount with the FORCE option.
Note: An unmount force can cause a grow or clone to be terminated, causing a partial grow or a partial clone. The partial grow should not cause a mount failure. The grow can be issued again and completed. A partial clone must be deleted and issued again. An unmount force of a quiesced aggregate could cause a backup to fail if the quiesce was issued by the backup program. (It will fail when it tries to issue the unquiesce.)

zFS: Permit the SMB server user ID to BPX.DAEMON and BPX.SERVER (Required, as of R11)
In z/OS V1R11, the SMB server requires access to BPX.DAEMON and BPX.SERVER. The new settings provide greater security.
Migration action: Define the BPX.SERVER and BPX.DAEMON FACILITY classes in RACF (if not already defined) using the following steps:
1. Create the BPX.SERVER and BPX.DAEMON FACILITY class profile:
   ```
   RDEFINE FACILITY BPX.SERVER UACC(NONE)
   RDEFINE FACILITY BPX.DAEMON UACC(NONE)
   SETROPTS RACLIST(FACILITY) REFRESH
   ```
2. Permit the DFS user ID to the BPX.SERVER and BPX.DAEMON RACF FACILITY class profiles:
   ```
   PERMIT BPX.SERVER CLASS(FACILITY) ACCESS(READ) ID(DFS)
   PERMIT BPX.DAEMON CLASS(FACILITY) ACCESS(READ) ID(DFS)
   SETROPTS RACLIST(FACILITY) REFRESH
   ```
3. Follow your installation's procedure for allowing the SMB daemon control task (IOEPDCT) to run if your system is set up for RACF program control protection. If your system is not set up for RACF program control protection, no action is necessary.

If this security setup is not done prior to SMB startup, the following message occurs: `IOEN00514A`
```
ThisProgram: Not permitted to BPX.DAEMON. Severity: svc_c_sev_fatal.
```

Various Migration Actions for z/OS R12 from z/OS R10

- **ICSF HCR7770:** **Modify ICSF startup procedure (Required as of HCR7770, incorporated into R12)**
  - **Switch from PGM=CSFMMAIN to CSFINIT.**

- **HCM:** **Install the new HCM (Required, as of R11)**
  - **As of R11, the installation of HCM is new – it is an MSI package and is installed via Windows Installer.**
  - **Uninstall previous HCM. Install R11 HCM. An upgrade of your older level is not possible.**

- **IBM Tivoli Directory Server:** **Determine if you will be using IBM TDS WLM support (Required, as of R11)**
  - **As of R11, IBM TDS is enhanced to use WLM performance goals for work within the LDAP server (by default). It is based on client IP addr or the bound user's distinguished name assoc. with LDAP transaction names.**
    - **Ensure the srvStarUpError option is terminate (default).**
    - **Permit the LDAP server userid to BPX.WLMSERVER facility class.**
    - **Use WLM ISPF panels for classifications rules.** *If a default service class for the LDAP subsystem is not configured, it will be discretionary!*
  - **You may turn off the WLM support if you like.**

33   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

Various Migration Actions Between z/OS V1 R10 and z/OS V1 R12

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*.

ICSF: Modify ICSF startup procedure (Required, as of HCR7770 web deliverable, incorporated into R12)

The program that started ICSF in earlier releases was named CSFMMAIN. In ICSF FMID HCR7770 (which was made available in the Web deliverable *Cryptographic Support for z/OS V1R9-R11* and has been incorporated into z/OS R12), the CSFMMAIN program is replaced by the CSFINIT program. If your ICSF startup procedure is not modified to run this new program, the procedure will not start the HCR7770 level of ICSF.

Migration action: In your startup procedure for ICSF:

1. Find the job step that identifies the ICSF startup program (CSFMMAIN) that was used in earlier releases.
   For example: `CSFSTART EXEC PGM=CSFMMAIN,REGION=0M,TIME=1440`
2. Modify the PGM parameter on this EXEC statement to identify the new startup program (CSFINIT):
   `CSFSTART EXEC PGM=CSFINIT,REGION=0M,TIME=1440`
3. Save your changes to the startup procedure.

Tip: Member CSF in SYS1.SAMPLIB contains a sample JCL code for an ICSF startup procedure.

More details about this migration action (from WSC Flash FLASH1062):

ICSF is being made non-swappable and non-cancelable using standard z/OS conventions. Prior versions of ICSF are non-swappable however this new support is implemented using standard conventions. Current versions of ICSF are cancelable via an operator command. If the operator issues the cancel command multiple times, the address space may be terminated before cleanup can be completed and this can sometimes cause problems to other ICSF started tasks that are sharing the key data sets (CKDS, PKDS, TKDS). By making ICSF non-cancelable, operations will have to issue an explicit command to stop the address space, driving it through normal termination routines.

Making the ICSF started task non-cancelable and non-swappable requires that the ICSF initialization module be

added to the Program Properties Table (PPT). To avoid impacting prior versions of ICSF this main module name is being changed from CSFMMAIN (PGM=CSFMMAIN for HCR7751 and earlier) to CSFINIT. Simply adding CSFMMAIN to the PPT, would cause current versions of ICSF to work differently than they do today (i.e. cancel operations would fail).

The CSFINIT entry in the PPT will be automatically generated as part of the install of HCR7770, however existing users of ICSF will need to modify the ICSF started task to use the new initialization module. CSFMMAIN will still exist with HCR7770, however it will simply generate error message CSFM022E explaining that CSFINIT is required and then ICSF will terminate.

Since this version of ICSF can be installed on older versions of z/OS, the migration from CSFMMAIN to CSFINIT will need to be carefully coordinated in shops running multiple versions of z/OS and ICSF. Systems that require earlier versions of ICSF will have to continue to use //CSF EXEC PGM=CSFMMAIN,REGION=0M,TIME=1440 in the started task. However, to use HCR7770, the started task will need to be modified to specify
//CSF EXEC PGM=CSFINIT,REGION=0M,TIME=1440

Installations that are running multiple LPARs with various versions of ICSF across those LPARs, might want to consider using a system variable for the startup program name. For example, the ICSF started task could be modified to specify
//CSF EXEC PGM=&ICSFPROG,REGION=0M,TIME=1440
and the system symbolic ICSFPROG could be set to
SYMDEF(&ICSFPROG='CSFINIT') on the HCR7770 systems and
SYMDEF(&ICSFPROG='CSFMMAIN') on the earlier systems. No matter whether you change the started task directly to point to the new startup module, or use system symbols, the change will have to be made before HCR7770 can be started. If using system symbols, the system symbol will have to be refreshed before ICSF can be restarted.

Alternatively, the ICSF started task could be moved to a release specific PROCLIB. On the HCR7770 systems the version of ICSF that uses
//CSF EXEC PGM=CSFMINIT,REGION=0M,TIME=1440
would come before the library that contains the JCL with
//CSF EXEC PGM=CSFMMAIN,REGION=0M,TIME=1440. On the pre-HCR7770 systems, the concatenation order would be reversed.

HCM: Install the new HCM (Required, as of R11)
In z/OS V1R11, the installation process for HCM is new. The code is delivered as an MSI package and is installed using the Windows® Installer. If you have an older version of HCM installed, it is strongly recommended that you uninstall it before you install the new HCM, which uses the Windows Installer. An upgrade of your older level is not possible.
Migration action: Uninstall your previous HCM using the Windows uninstall facility or use the **Unistall** function provided by HCM from the following path: All Programs-->IBM Hardware Configuration Manager-->Uninstall. Do this before installing z/OS V1R11 HCM.
Note: With z/OS V1R11, the location of the EEQHCM.INI file changes from C:\WINDOWS to the user-specific application data directory. This does not affect your migration because the first invocation of HCM automatically copies the old INI file to the new location or creates a new one if no INI file is found.

The default name of the user-specific directory, for example, for Windows XP, is C:\Documents and Settings\<*user*>\Application Data\IBM\Hardware Configuration Manager. Due to changes in the user authorization methods for Windows XP and later versions, the change of the location is necessary to allow users who are not administrators to update this file.

IBM TDS: Determine if you will be using IBM TDS WLM support (Required, as of R11)
Before z/OS V1R11, the IBM TDS did not support using Workload Manager (WLM) to allow an installation to set performance goals for work within the LDAP server. Beginning with z/OS V1R11, the IBM TDS is enhanced to use

WLM performance goals for work within the LDAP server, based on the client IP address or the bound user's distinguished name (DN) associated with LDAP requests. If a client application is spamming the LDAP server with numerous LDAP requests, a lower priority can be assigned to those requests. This support can prevent the LDAP server's resources from being consumed by spamming LDAP client applications.

Migration action: If you want to use the IBM TDS WLM support, which is the default:

1. Set the srvStartUpError option to terminate in the LDAP server configuration file. This is the default setting for this configuration option.
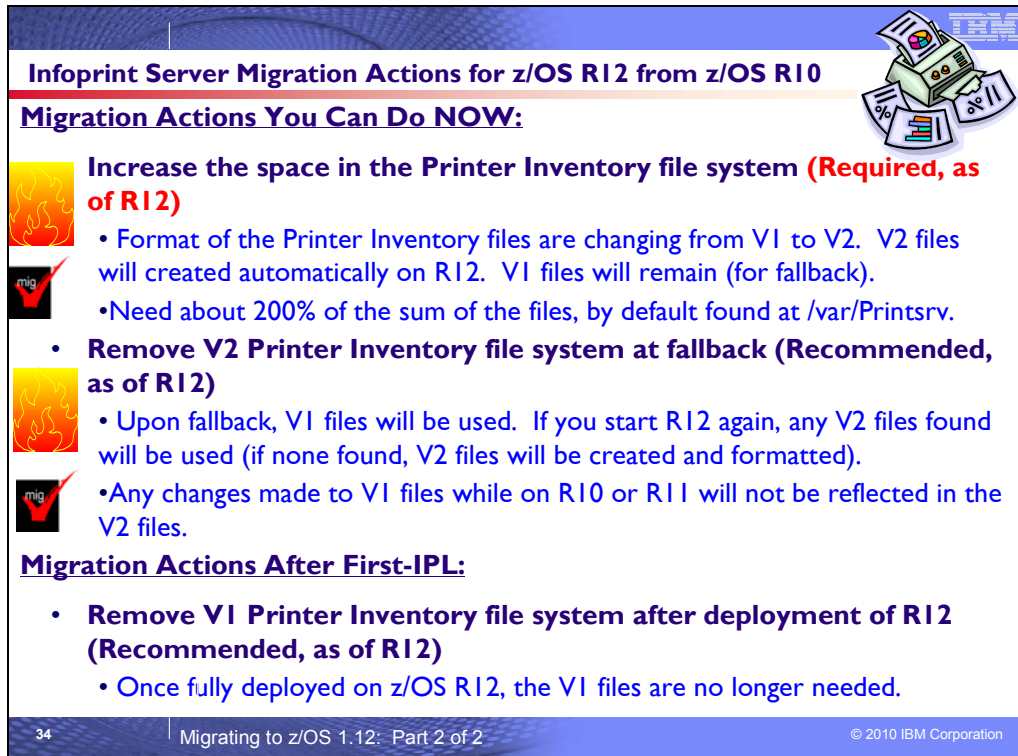2. Issue the following RACF commands before starting IBM TDS:
   ```
   RDEFINE FACILITY BPX.WLMSERVER UACC(NONE)
   PERMIT BPX.WLMSERVER CLASS(FACILITY) ID (userid) ACCESS(READ)
   SETROPTS RACLIST(FACILITY) REFRESH
   ```
   where *userid* is the user ID of the LDAP server.
3. In the WLM ISPF panels, define a new classification rule using the subsystem type of LDAP. If you had a subsystem type of LDAP defined previously (for the Integrated Security Services LDAP Server), it will now be used for the IBM TDS. Review any classification rules that you had for the Integrated Security Services LDAP Server, and ensure that they are desirable for the IBM TDS.

   Define classification rules for the LDAP subsystem type, using a transaction name of GENERAL (for LDAP work), and any other user-defined WLM transaction names used for the IBM TDS. Refer to *z/OS MVS Planning: Workload Management* for all the workload qualifiers that the LDAP subsystem type supports. ***If a default service class for the LDAP subsystem is not configured in WLM, all LDAP server operations will run under the discretionary goal and receive a low priority.***

4. If you do not want to use the IBM TDS WLM support, set the srvStartUpError option to ignore in the LDAP server configuration file.

Infoprint Server Migration Actions for z/OS R12 from z/OS R10

Migration Actions You Can Do NOW:

- **Increase the space in the Printer Inventory file system (Required, as of R12)**
  - Format of the Printer Inventory files are changing from V1 to V2. V2 files will created automatically on R12. V1 files will remain (for fallback).
  - Need about 200% of the sum of the files, by default found at /var/Printsrv.
- **Remove V2 Printer Inventory file system at fallback (Recommended, as of R12)**
  - Upon fallback, V1 files will be used. If you start R12 again, any V2 files found will be used (if none found, V2 files will be created and formatted).
  - Any changes made to V1 files while on R10 or R11 will not be reflected in the V2 files.

Migration Actions After First-IPL:

- **Remove V1 Printer Inventory file system after deployment of R12 (Recommended, as of R12)**
  - Once fully deployed on z/OS R12, the V1 files are no longer needed.

34   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

---

Infoprint Server Migration Actions Between z/OS V1 R10 and z/OS V1 R12

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.*

Infoprint Server Migration Actions You Can Do Now

Increase space in the Printer Inventory file system (Required, as of R12)

In z/OS V1R12, the format of the Infoprint Server Printer Inventory files has changed from Version 1 to Version 2 format. When you start Infoprint Server on z/OS V1R12 for the first time, Infoprint Server reformats the Version 1 Printer Inventory files and creates Version 2 Printer Inventory files. The Version 1 Printer Inventory files are not removed so that if you need to fall back to the previous z/OS release, Infoprint Server can use the Version 1 Printer Inventory files. Therefore, the Printer Inventory file system requires more space in z/OS V1R12 than in previous releases. You might need to increase space in the Infoprint Server Printer Inventory file system.

On a z/OS V1R11 or z/OS V1R10 system, Infoprint Server cannot read or export the Version 2 Printer Inventory. If you want to use the Version 2 Printer Inventory on z/OS V1R11 or z/OS V1R10, use the **pidu** command to export the Version 2 Printer Inventory while running on z/OS V1R12 and then use the **pidu** command to import the exported copy to z/OS V1R11 or z/OS V1R10.

Migration action:
1. Do one of these:
    a. Run the IBM Health Check ZOSMIGV1R12_INFOPRINT_INVSIZE available with APAR OA32093..
    b. Run the **df** command to display the current utilization of the Printer Inventory file system. Printer Inventory files are located in the Infoprint Server base directory. The default base directory name is /var/Printsrv. You might have changed the base directory name in the **base-directory** attribute in the aopd.conf configuration file. The aopd.conf default location is /etc/Printsrv/aopd.conf. However, you might have specified a different location in environment variable AOPCONF. The free space required is 200% of the sum of the Version 1 Printer Inventory and historical Printer Inventory files (master.db, jestoken.db, pwjestoken.db, hinv/hinv.db, and logdb/log.db). If the "Capacity" is greater than 33%, increase the size of the file system.

2. To increase the size of the file system, you can use the z/OS UNIX **zfsadm grow** (zFS) or **confighfs** (HFS) command.

Set the **aggrfull** (zFS) or FSFULL (HFS) file system option so that warning messages are issued if the Infoprint Server base directory (/var/Printsrv) is getting full.

Remove Version 2 Printer Inventory files at fallback (Recommended, as of R12)

Not *required, but recommended if you want Infoprint Server to reformat the Version 1 Printer Inventory files after a fallback.*
In z/OS V1R12, the format of the Infoprint Server Printer Inventory files has changed from Version 1 to Version 2 format. When you start Infoprint Server on z/OS V1R12 for the first time, Infoprint Server reformats the Version 1 Printer Inventory files and creates Version 2 Printer Inventory files. Both Version 1 and Version 2 Printer Inventory files exist in the Infoprint Server base directory.   Infoprint Server on z/OS V1R12 uses the Version 2 Printer Inventory files. If you fall back to z/OS V1R11 or V1R10, Infoprint Server uses the Version 1 Printer Inventory files.

If you start Infoprint Server on z/OS V1R12 a second time after falling back to a previous z/OS release, Infoprint Server uses the existing Version 2 Printer Inventory files that it created the first time you started Infoprint Server on z/OS V1R12. It does not reformat the Version 1 Printer Inventory files again. If you want Infoprint Server to reformat the Version 1 Printer Inventory files again, remove the Version 2 Printer Inventory files before you start Infoprint Server on z/OS V1R12. Because the Version 2 Printer Inventory files no longer exist, Infoprint Server reformats the Version 1 Printer Inventory files and creates a new set of Version 2 Printer Inventory files.

In most cases, you should remove the Version 2 Printer Inventory files if they exist. If you do not remove the Version 2 Printer Inventory files, any changes that the administrator made to the Version 1 Printer Inventory on z/OS V1R11 or z/OS V1R10 are not in the Version 2 Printer Inventory. In addition, Infoprint Central on z/OS V1R12 cannot display historical information for jobs that Infoprint Server processed on z/OS V1R11 or z/OS V1R10.
Migration action:
1. Run the IBM Health Checker for z/OS check INFOPRINT_V2DB_CHECK.
2. If Version 2 Printer Inventory files exist after falling back to z/OS V1R11 or z/OS V1R10, remove them from the Infoprint Server base directory. Be careful not to remove any Version 2 files while running z/OS V1R12 because Infoprint Server on z/OS V1R12 requires Version 2 Printer Inventory files. Version 2 files have the extension "v2db". The default base directory is /var/Printsrv. You might have changed the base directory name in the **base-directory** attribute in the aopd.conf configuration file. The aopd.conf default location is /etc/Printsrv/aopd.conf. However, you might have specified a different location in environment variable AOPCONF.
   Example: These z/OS UNIX commands switch to an effective UID of 0, remove all files with the "v2db" extension from directory /var/Printsrv, and switchback to the original UID:
   su
   rm -f $(find /var/Printsrv/ -name "*.v2db")
   exit

Note: To remove Printer Inventory files, you must have an effective UID of 0 or be a member of the RACF AOPADMIN group.

Upgrade Java support for IPP Server (Required-IF, as of R12)
Required if you use IPP Server and specify the JAVA_HOME environment variable. You are using IPP Server if the start-daemons={ippd} attribute is specified in the Infoprint Server configuration file. The configuration file's default location is /etc/Printsrv/aopd.conf. However, you might have specified a different location in environment variable AOPCONF.
In z/OS V1R12, the Internet Printing Protocol (IPP) Server component of Infoprint Server requires Java V6.0. If the JAVA_HOME environment variable specifies the location of an earlier version of Java, you must update the JAVA_HOME environment variable.
Migration action:
1. Install IBM 31-bit SDK for z/OS, Java 2 Technology Edition, V6 (5655-R31).
2. If you use the IPP Server, edit the **aopstart** EXEC to update the directory path specified in the JAVA_HOME environment variable. IPP Server requires the 31-bit version of Java V6.0.
3. If you use the z/OS HTTP Server, update the setting of JAVA_HOME in the z/OS HTTP Server environment variables file httpd.envvars.

Note: If you installed Java V6.0 in the default Java directories, you do not need to specify the JAVA_HOME environment variable. If JAVA_HOME is not specified, IPP Server looks for Java files in the /usr/lpp/java/J6.0 directory.

Infoprint Server Migration Actions Pre-First IPL

Remount the Printer Inventory and copy files that were customized (Required)
When migrating to z/OS V1R12 Infoprint Server, you must bring forward the customized data from your previous system.
Migration action: Follow the instructions in *z/OS R12 Migration* for bringing forward the required files.

Upgrade XML for Infoprint Central (Required-IF, as of R12)
Required if you use Infoprint Central. You are using Infoprint Central if the start-daemons={ssid} attribute is specified in the Infoprint Server configuration file. The file's default location is /etc/Printsrv/ aopd.conf. However, you might have specified a different location in environment variable AOPCONF.
In z/OS V1R12, the Infoprint Central component of Infoprint Server,
which you can use to work with IP PrintWay extended mode print jobs and
printers, requires the IBM XML Toolkit for z/OS V1.10 (5655-J51) product.
Migration action:
1. Install IBM XML Toolkit for z/OS V1.10 (5655-J51).
2. Specify the XML Toolkit for z/OS V1.10 libraries in the LIBPATH environment variable in your z/OS IBM HTTP Server environment variables file (default location is /etc/httpd.envvars). After z/OS V1R12 is installed, Infoprint Central requires the XML Toolkit for z/OS V1.10 libraries:
   - LIBPATH: change /usr/lpp/ixm/IBM/xml4c-5_6/lib to /usr/lpp/ixm/IBM/xml4c-5_7/lib
   - LIBPATH: change /usr/lpp/ixm/IBM/xslt4c-1_10/lib to /usr/lpp/ixm/IBM/xslt4c-1_11/lib
   - ICU_DATA: You can remove this variable because XML no longer uses this variable.
3. Restart the z/OS IBM HTTP Server to pick up the changes to the environment variables file.

Infoprint Server Migration Actions Post-First IPL

Remove Version 1 Printer Inventory files after deploying z/OS V1R12 (Recommended, as of R12)
Not required, but recommended to free up space in the Infoprint Server base directory.
In z/OS V1R12, the format of the Infoprint Server Printer Inventory files has changed from Version 1 to Version 2 format. When you start Infoprint Server in z/OS V1R12, Infoprint Server reformats the Version 1 Printer Inventory files and creates Version 2 Printer Inventory files. The Version 1 Printer Inventory files are not removed so that if you need to fall back to the previous z/OS release, Infoprint Server can use the Version 1 Printer Inventory files. After you have fully deployed z/OS V1R12 and are sure that you will not need to fall back to the previous z/OS release, you can remove the Version 1 Printer Inventory files to free up space in the Infoprint Server base directory. If you need to fall back to a previous z/OS release after removing the Version 1 Printer Inventory files, use the **pidu** command to export the Version 2 Printer Inventory on the z/OS V1R12 system and import the exported copy to the z/OS V1R11 or z/OS V1R10 system.
Migration action: Remove the Printer Inventory Version 1 database files in the Infoprint Server base directory. Version 1 files have a "db" extension. The default base directory is /var/Printsrv. You might have changed the base directory name in the **base-directory** attribute in the aopd.conf configuration file. The aopd.conf default location is /etc/Printsrv/aopd.conf. However, you might have specified a different location in environment variable AOPCONF.

Example: These z/OS UNIX commands switch to an effective UID of 0, remove all files with the "db" extension from directory /var/Printsrv, and switch back to the original UID:
```
su
rm -f $(find /var/Printsrv/ -name "*.db")
exit
```
Note: To remove Printer Inventory files, you must have an effective UID of 0 or be a member of the RACF AOPADMIN group.

JES2 Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Update code remove references to PDBLENG (Required-IF, as of R12)**
    - As of R12, variable size PDDBs. Exits need to use a runtime length field, not compile time length. Compile time length equate PDBLENG has been deleted.
- **Ensure calls to JES Property Information Services SSI can handle multiple members (Required-IF, as of R12)**
    - As of R12, if information on multiple members is requested by wildcard on the member filter, all members that match will be returned. Not just the local member only, which was the prior behavior.

Migration Actions Post-First IPL:

- **Migrate to z11 mode (Recommended, as of R11)**
    1. Determine your z11 checkpoint activation readiness with the $D ACTIVATE command. This command indicates if activation to z11 mode will succeed.
        - Review your current utilization of BERT data to determine if there are sufficient BERTS. Additional BERTs are needed for each SYSOUT data set that has transaction data associated with it.
    2. Run the JES2 $ACTIVATE command to activate z11 mode following the considerations for this command found in *z/OS JES2 Commands*.

35   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

JES2 Actions Between z/OS V1 R10 and z/OS V1 R12

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

JES2 Migration Actions You Can Do Now
Update code to remove references to PDBLENG (Required-IF, as of R12)

Required if installation exits use PDBLENG equate.

Starting with z/OS V1R12, JES2 supports variable size PDDBs, though the PDDBs generated in this release remain a fixed size. Installation exits that examine PDDBs or step through PDDBs using the compile time length of the PDDB need to be updated to use a run time length field. To facilitate locating an exit code that is assuming a fixed PDDB size, the compile time length equate PDBLENG has been deleted. Code that used this compile time length should be updated to use the run time field PDBSIZE to determine the size of the PDDB. The field PDBSIZE has correctly contained the length of the PDDB since z/OS V1R7 (the field existed in earlier releases but was not consistently set).

Migration action: Before installing z/OS V1R12 JES2, review installation exits for references to the field PDBLENG. If any references are found, the code needs to be updated to use the run time PDDB length field PDBSIZE.

Ensure calls to JES Property Information Services SSI can handle multiple members (Required-IF, as of R12)

Required if you are using the Initiator information function of the JES Property Information Services SSI.

In z/OS V1R11 JES2, the Initiator information function of the JES Property Information Services SSI (SSI 82) returned information for the local member only, even if multiple members matched the value that was specified on the member filter. In z/OS V1R12 JES2, if information for multiple members is requested by specifying wildcards on the member filter, the Initiator information function will return information for all members that match the filter request.

Migration action: Before installing z/OS V1R12 JES2, ensure that all calls to the Initiator information function of the JES Property Information Services SSI (SSI 82) that request information for multiple members can correctly handle information being returned for multiple members.

JES2 Migration Actions Pre-First IPL
Update uses of JES2 macros and exits (Required-IF, as of R11)

Required if using various JES2 macros or exits mentioned in z/OS Migration.

With z/OS V1R11 enhanced transaction SYSOUT selection for the SYSOUT API and extended status SSIs, a variable extension is added to the JOE data area. This extension is stored in the BERT data area in the JES2 checkpoint. To manage this data area in a release independent manner, a check out and check in service ($DOGJOE) is available to manage JOE data areas. This service must be used when JOE data areas are being modified. It is also recommended that this service be used to examine fields in a JOE data area.

Installation exits that update JOE data areas need to be updated to use the $DOGJOE service to check out (fetch) an update mode JOA. The checked out JOA (also referred to as an artificial JOE) is a composite of the work JOE, the characteristics (char) JOE, and the JOE extension in the BERTs. The $DOGJOE service also insulates the exit code from the current $ACTIVATE level of JES2.

Various z/OS V1R11 JES2 macros changed as a result of the $DOGJOE service. Certain macro interfaces now require a JOA instead of a work JOE or characteristics JOE. In other scenarios, the rules for using these macros have changed.

Migration actions: Before upgrading to z/OS V1R11 JES2, certain conditions must be met if your installation is using any of the following JES2 exits and/or macros. The actions described below must be considered regardless of checkpoint activation level.

- Be aware of the $DOGJOE macro if you reference JOE fields in any of your exits. This new macro enables creation of read and update mode artificial JOEs termed JOAs.
- Determine which fields need to be replaced for specific exits. Before z/OS V1R11 JES2, a real work and sometimes a characteristics JOE were passed to JES2 exits 1,15,38,46, or 56. Starting with z/OS V1R11 JES2, an artificial JOE (JOA) will be passed to each of these exits. The table in *z/OS Migration* explains which fields need to be replaced for specific exits.
- Make required code updates if you use JES2 macros $#ADD, $#ALCHK, $#BLD and $#BUSY. In many cases, the interface has changed to require that a JOA be passed into the macro instead of a work JOE or the work/characteristics JOE combination. In the case of $#BUSY and $#ALCHK, additional rules must be followed.
- Be aware that the default for $#JOE has been changed to return a read mode JOA. It is the responsibility of the user to return a JOA upon early exit from the $#JOE processing loop. The $#JOE macro returns a real JOE or a read mode JOA. Before z/OS V1R11 JES2, this macro only returned a real JOE.
- Change exit routines that examine field CBMTTR to examine field CBMQTR, which is now a 6-byte MQTR. Prior to z/OS V1R11 JES2, $CBIO used 4-byte MTTRs internally to determine what record is being read or written. Starting with z/OS V1R11 JES2, $CBIO uses 6-byte MQTRs internally. Though not formally passed to exit 7 ($CBIO exit), some exits are known to locate the track address being processed.

JES2 Migration Actions Post-First IPL

Migrate to z11 mode (Recommended, as of R11)

Not required, but recommended to activate the full-function level of z/OS V1R11 JES2 processing.

If you wish to take advantage of the full-function level of z/OS V1R11 JES2, you must be in z11 mode. Activating z11 mode upgrades the JES2 checkpoint and enables JES2 functionality that is introduced in z/OS V1R11, including JOE data area extensions supported by BERTs. For more information on the JES2 functionality introduced in z/OS V1R11, see the reference links below.

Migration action:

- After migrating to z/OS V1R11 JES2 on all systems in your MAS, determine your z11 checkpoint activation readiness:
    1. Use the $D ACTIVATE command. This command indicates if activation to z11 mode will succeed.
    2. Review your current utilization of BERT data to determine if there are sufficient BERTS, as detailed in "**Check BERT utilization**" below.
- Run the JES2 $ACTIVATE command to activate z11 mode following the considerations for this command found in *z/OS JES2 Commands*.

By default, JES2 restarts in the same mode (z2 or z11) as other members of the MAS (if any are active) or the mode the last active JES2 member was in when it came down. To restart JES2 in z2 mode, specify UNACT on PARM=. On a cold start JES2 starts in z11 mode unless overridden by OPTSDEF COLD_START_MODE.

Tip: Use IBM Health Checker for z/OS health check JES2_Z11_Upgrade_CK_JES2. This check determines if the system is ready to upgrade the JES2 checkpoint to z11 mode.

Check BERT utilization

Before issuing the $ACTIVATE,LEVEL=z11 command, review the current utilization of BERT data to determine whether there are sufficient BERTs. Additional BERTs are needed for each SYSOUT data set that has transaction data associated with it. These SYSOUT data sets can be seen using SDSF by setting APPC ON and examining SYSOUT data sets on the H and O panels; SYSOUT data sets with transaction data have nontraditional JES2 job IDs. Consider increasing the number of BERTs to correspond to two times the maximum number of transaction SYSOUT data sets on the system. BERT utilization should be monitored after the $ACTIVATE to z11 mode to ensure there are sufficient BERTs for the jobs and SYSOUT in the MAS. There are several ways to determine your current BERT usage:

- The $D CKPTSPACE,BERTUSE command displays a table of the types of control blocks in BERTs and how many BERTs are used by each control block type. The example below shows the output of the command:

```
$HASP852 CKPTSPACE CURRENT BERT UTILIZATION
$HASP852 TYPE COUNT CB COUNT
$HASP852 -------- --------- ---------
$HASP852 INTERNAL 11 1,
$HASP852 JQE 211 108,
$HASP852 CAT 114 38,
$HASP852 WSCQ 1 1,
$HASP852 DJBQ 0 0,
$HASP852 JOE 0 0,
$HASP852 FREE 763 0
```

In the example, there are 108 JQEs that have a total of 211 BERTs associated with them. This example is for a system in z2 mode and does not have any BERTs associated with JOEs.

- The $D ACTIVATE command displays the number of BERTs that are needed for activation to z11 mode. This is the number of BERTs that will be associated with JOEs after the $ACTIVATE. The example below shows the output of the $D ACTIVATE command.

```
$HASP895 $DACTIVATE
$HASP895 JES2 CHECKPOINT MODE IS CURRENTLY Z2
$HASP895 THE CURRENT CHECKPOINT:
$HASP895 -- CONTAINS 1100 BERTS AND BERT UTILIZATION IS 30
$HASP895 PERCENT.
$HASP895 -- CONTAINS 158 4K RECORDS.
$HASP895 z11 CHECKPOINT MODE ACTIVATION WILL:
$HASP895 -- EXPAND CHECKPOINT SIZE TO 165 4K RECORDS.
$HASP895 -- REQUIRE 22 ADDITIONAL BERTS AND UTILIZATION
$HASP895 WOULD REACH 32 PERCENT.
$HASP895 z11 ACTIVATION WILL SUCCEED IF ISSUED FROM THIS MEMBER.
```

In the example, there are 22 additional BERTs that will be used after the $ACTIVATE to z11 mode, for transaction data associated with JOEs.

- A general history of BERT usage can be obtained by using the $JD HISTORY(BERT) command or by using the SDSF RM panel. This displays the usage of BERTs after the system was IPLed. The example below shows the output of the $JD HISTORY(BERT) command:

```
$HASP9130 D HISTORY
$HASP9131 JES2 BERT USAGE HISTORY
DATE TIME LIMIT USAGE LOW HIGH AVERAGE
-------- -------- -------- -------- -------- -------- --------
2009.086 16:00:00 1100 337 337 337 337
2009.086 15:50:09 1100 337 125 337 192
```

Reference information:
- For a list of the enhancements introduced in z/OS V1R11 for z11 mode, see *z/OS Introduction and Release Guide*.
- For $ACTIVATE, $D ACTIVATE, $D CKPTSPACE and $JDHISTORY command details, see *z/OS JES2 Commands*.

JES3 Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

Identify the JES3 auxiliary address space to your security product (Required-IF, as of R11)

•Verify the name of the address space (JES3AUX, for instance). Check for a profile defined in the STARTED class that covers the name of the JES3AUX address space. If none is defined, create a profile for JES3AUX. Make sure JES3AUX has the trusted attribute.

• **Update exit IATUX72 (Required-IF, as of R11)**

•As of R11: when you use SSI 70 to move a data set from the hold queue to the TCP or BDT queues, SSI 70 calls exit 72. SSI 70 is a new caller of exit 72, so the parameter list is changed.

•Reassemble your IATUX72 against the new parameter list, possibility update your exit to handle the possibility of SSI 70 calling this exit.

• **Modify code that depends on the value of STVSTOD returned by extended status (Required-IF, as of R11)**

• Before R11: extended status includes a timestamp, STVSTOD, for each data set returned. It represented when the output data set became available.

•As of R11: the value now represents when the data set is created. Reflects when a data set is available for spool browse.

•Ensure any dependency on STVSTOD tolerates a timestamp when data set is created, instead of when output is available.

36 Migrating to z/OS 1.12: Part 2 of 2

Complete migration list is here!

JES3 Migration Actions Between z/OS V1 R10 and z/OS V1 R12

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.*

JES3 Migration Actions You Can Do Now
Modify code that depends on the value of STVSTOD returned by extended status (Required-IF, as of R11)
Required if the installation has a code dependency on the value of STVSTOD corresponding to the time that output is available and not before. It is expected that this dependency is unlikely.
Extended status includes a timestamp, STVSTOD, for each dataset returned. Prior to z/OS V1R11, the timestamp value represented when that output dataset became available. Beginning with z/OS V1R11, the value is changed to represent when the dataset is created. This new value reflects when a dataset is available for spool browse.
Migration action:
Ensure that any dependency on STVSTOD will tolerate a timestamp corresponding to when the dataset is created intead of when output is available

JES3 Migration Actions Pre-First IPL

Identify the JES3 auxiliary address space to your security product (Required-IF, as of R11)
Required if your security product is configured to monitor started tasks.
The JES3 auxiliary address space provides critical functional support for JES3 and is started during JES3 initialization. Before z/OS V1R11, a limited-function address space for JES3AUX was created by using a link to IEEMB881. Beginning with z/OS JES3 V1R11, the ASCRE macro is used to create the JES3 auxiliary address space. Therefore, you need to identify this address space to your security product (for example, RACF) if it is configured to monitor started tasks. JES3 will not start unless this action is taken.
Migration action:
1. Determine the JES3 auxiliary address space name. This name consists of the primary subsystem name combined with AUX. The primary subsystem name is defined in the IEFSSN*xx* member of SYS1.PARMLIB. For example, if the primary subsystem name is JES3, then the address space will be JES3AUX.
2. Verify that the JES3AUX address space is identified to your security product. For RACF, check for a profile defined in the STARTED class that covers the name of the JES3AUX address space. If none is defined, create a profile for JES3AUX. Make sure JES3AUX has the trusted attribute. For example:

```
        RDEFINE STARTED JES3AUX.* STDATA( USER(JES3ID) GROUP(SYS1) TRUSTED(YES) )
```

Update JES3 exit IATUX72 **(Required-IF, as of R11)**
Required if you use exit IATUX72.
Beginning with z/OS V1R11, you can use SSI 70 to change the characteristics of a SYSOUT data set. If your changes are to move a data set from the hold queue to the TCP or BDT queues, SSI 70 calls exit 72 (IATUX72). Because SSI 70 is a new caller of exit 72, the parameter list of the exit, IATYUX72, is changed. This change consists of a new bit, YUX73SJF, which identifies SSI 70 as the caller, and a new pointer, UX72SMW, which addresses a work area, IATYSMW, that is used by SSI 70.
Migration action: Reassemble your IATUX72 exit against the new IATYUX72 parameter list. In addition, if your IATUX72 exit bases its actions either on its caller or on the caller specific data that is provided in the IATYU72 parameter list, update your exit to handle the possibility of SSI 70 calling the exit.

JES3 Migration Actions Post-First IPL

Language Environment Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Determine the impact of added and changed runtime options (Required-IF, as of R11)**
  - As of R11: four suboptions were added to HEAPCHK for serviceability of heap pools. These suboptions allow you to specify a heap pools trace table size that is greater or less than the current hardcoded size of 1024, and to limit heap pools tracing to a single pool instead of always tracing all pools.
  - As of R12: none.
  - If you use CEEPRMxx: *no migration action* if you want to use the defaults of the newly added options. The newly added options do not change default behavior.
  - If you use assembler language modules: update and reinstall your usermods, or better yet – stop using them. **Note that these usermods will no longer be supported in the future!**

- **Set runtime options as overrideable or nonoverrideable in CEEPRMxx parmlib member (Recommended, as of R12)**
  - As of R12: NONOVR and OVR are supported in CEEPRMxx, removing a barrier to using CEEPRMxx.

Language Environment Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Examine programs that read output from a CICS CLER transaction (Required-IF, as of R12)**
  - Report output is changed: LAST WHERE SET is wider, OPTIONS is now called OPTION.
- **Examine programs that read output when a D CEE command is issued (Required-IF, as of R12)**
  - Command response is changed: only suboptions specified with a parmlib member or SETCEE command (not those not explicitly set, as before) are displayed. Commas are now displayed for suboptions not explicitly set.
- **Use Unicode Services to create conversation tables (Req-IF, as of R12)**
  - C/C++ runtime library no longer includes an ucmap source code or genxit source code for character conversions being performed by Unicode Services.
  - If you use customized conversation table, generate custom Unicode Services conversation tables.
  - If you use iconv() testing to a "known conversion result" and experience failures, update your expected results to the new results.
  - If you want to create custom conversation tables involving any CCSIDs related to the conversion table source no longer being shipped, generate custom Unicode conversion tables instead of LE conversion tables.

Language Environment Migration Actions Between z/OS V1 R10 and z/OS V1 R12

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

Language Environment Migration Actions You Can Do Now

<none>

Language Environment Migration Actions Pre-First IPL

Update the CICS CSD based on the newest CEECCSD sample (Required)

Each release, Language Environment adds or deletes load modules in the CICS system definition (CSD) file. Thus, you should update the file each release using the program definitions found in member CEECCSD and, if using CICS Transaction Server (TS) for z/OS V3 (5655-M15), in member CEECCSDX..
Migration action:
- Update the CSD file using the program definitions in member CEECCSD (and member CEECCSDX if using CICS TS V3) found in the *hlq*.SCEESAMP data set.
- Complete the following steps so the CSD can be shared by CICS regions that are running on different releases of z/OS (this is an example of upgrading from z/OS V1R10 to z/OS V1R12):
    1. Enter CEDA COPY ALL GROUP(CEE) TO(LEZ110) to copy the z/OS V1R10 supplied CEE group to a group called LEZ110. Do the same for the z/OS V1R12 CEE group, but call it something like LEZ112.
    2. Enter CEDA ADD GROUP(LEZ110) LIST(*listname*) to include the old z/OS V1R10 definition group in your grouplist so all program definitions that are not included in the new z/OS group (LEZ112) will be included in your regions.
    3. Enter CEDA ADD GROUP(LEZ112) LIST(*listname*) to include the new z/OS V1R12 definition group after the old definition group (LEZ110) in your grouplist so all updated program definitions will be used in your regions.
    4. Enter CEDA REMOVE GROUP(LEZ110) LIST(*listname*) to remove the old z/OS definition group from the grouplist when all regions in the sysplex are upgraded to the new version of z/OS.

Note: The group containing the Language Environment runtime routines must be in the group list used during CICS startup.

Review Language Environment load modules in LPA (Required-IF)

Required if you need to make modules accessible through the link pack area (LPA).
Each release you must update the Language Environment load modules that you make accessible through the link pack area (LPA). In addition, each release you should review your list of load modules in the LPA to determine if it's still suitable.
Migration action: Review Language Environment load modules in the LPA. To move load modules into the LPA, use the following sample members in the CEE.SCEESAMP data set:
- *AFHWMLP2:* This is a sample of all Language Environment Fortran component modules eligible for the LPA.
- *CEEWLPA:* This is a sample of a PROG*xx* member of SYS1.PARMLIB that includes all Language Environment CEE-prefixed runtime modules eligible for the LPA (that is, all Language Environment base modules) except the callable services stubs.
- *CELQWLPA:* This is a sample for AMODE 64 runtime support.
- *EDCWLPA:* This is a sample of a PROG*xx* member of SYS1.PARMLIB that includes all Language Environment EDC-prefixed and CEH-prefixed runtime modules eligible for the LPA (that is, all XL C/C++ component modules) except locales and code page converters.
- *IBMALLP2 (or IBMPLPA1 for Enterprise PL/I for z/OS):* This is a sample of all Language Environment PL/I component modules eligible for the LPA.
- *IGZWMLP4:* This is a sample of all Language Environment COBOL component modules eligible for the LPA.

If you are considering placing the modules listed in *z/OS Language Environment Customization* in the LPA or the ELPA, then IBM recommends that you place the SCEELPA data set in the LPA list (LPALST*xx*). SCEELPA contains Language Environment load modules that are reentrant, that reside above the 16 MB line, and that are heavily used by z/OS.

In *z/OS Language Environment Customization* you will also see tables of modules eligible for the LPA and the ELPA above and beyond what is found in the SCEELPA data set. You will need to use the dynamic LPA or MLPA approach to move these modules into the LPA or ELPA. You do not need to include recommended modules if they contain functions your installation does not use. Language Environment modules not listed in these tables can be moved into the LPA or ELPA at your discretion.

Examine programs that read output when a D CEE command is issued (Required-IF, as of R12)
Required if you have an application that reads the output of a D CEE command.
Starting in z/OS V1R12, changes are made to the Language Environment runtime options report when a D CEE command is issued. Before z/OS V1R12, the Language Environment runtime options report displayed all suboptions, even if they were not explicitly set for any runtime option that was specified. Starting in z/OS V1R12, when a valid option is specified with a parmlib member or a SETCEE command, only the suboptions specified are displayed when a D CEE command is issued. A comma is displayed as a placeholder for those suboptions not specified.
Migration action: Examine any programs that read the output of a D CEE command to ensure compatibility with the updated runtime options report. Commas are now displayed for any suboptions that are not explicitly specified in a parmlib member or with a SETCEE command.

For example, if the following SETCEE command is issued:
```
SETCEE CEEDOPT,ALL31,ANYHEAP(4K),FILETAG(,AUTOTAG)
```
a subsequent D CEE,CEEDOPT command displays the following:
```
CEE3745I 09.32.13 DISPLAY CEEDOPT
NO MEMBERS SPECIFIED
LAST WHERE SET OPTION
----------------------------------------------------------------------
SETCEE command ALL31()
SETCEE command ANYHEAP(4096,,,)
SETCEE command FILETAG(,AUTOTAG)
```

Set runtime options as overrideable or nonoverrideable in CEEPRMxx parmlib member (Recommended, as of R12)
Not required, but recommended so you can eliminate use of the assembler language USERMODs to specify installation-wide runtime options, and use parmlib member CEEPRMxx instead.
Before z/OS V1R12, all runtime options specified in a CEEPRM*xx* parmlib member member were overrideable by default. Beginning with z/OS V1R12, you can set runtime options as overrideable or nonoverrideable in the CEEPRM*xx* parmlib member or with a SETCEE command using the OVR or NONOVR attribute. The ability to specify an option as overrideable or nonoverrideable removes a barrier to using CEEPRM*xx*.
Migration action: Set runtime options as overrideable or nonoverrideable in the CEEPRM*xx* parmlib member or by issuing the SETCEE command using the OVR or NONOVR attribute.
Now that runtime options can be specified as overrideable or nonoverrideable in a CEEPRM*xx* parmlib member, and with a SETCEE command, you can eliminate the use of assembler language USERMODs to specify installation-wide runtime options.

Determine the impact of added and changed run-time options (Required-IF, as of R11)
Required you are using assembler modules to specify your installation-wide runtime options (CEEDOPT, CEECOPT, or CELQDOPT). If you are using CEEPRMxx (introduced in z/OS V1R7) and the default specification for the newly added options is acceptable to you, then no migration action is required.
In z/OS V1R11:
- No options were added.
- Four suboptions were added to HEAPCHK to enhance the serviceability of heap pools. The suboptions allow you to specify a heap pools trace table size that is greater or less than the current hardcoded size of 1024, and to limit heap pools tracing to a single pool instead of always tracing all pools.. The suboptions apply to the main user heap in an application and the 31-bit heap (__malloc31()) in a 64-bit environment:
  - Number of Entries: Specifies the number of entries to be recorded in one heap pool trace table for the main user heap in the application. Each pool has its own trace table. If the number of entries is 0, the heap pool trace table is not generated.

- o Pool Number: Specifies which pools are traced for the main user heap in the application. You can either trace one pool or all pools. The value should be a valid pool number from 1 to 12. If the pool number is 0, all pools will be traced.
- o Number of Entries 31: Specifies the number of entries to be recorded in one heap pool trace table when an AMODE 64 application is using heap storage from 31–bit addressable storage __malloc31()). Each pool has its own trace table. If the number of entries is 0, the heap pool trace table is not generated. This value is only supported in an AMODE64 environment.
- o Pool Number 31: Specifies which pools are traced when an AMODE 64 application is using heap storage from 31-bit addressable storage (__malloc31()). You can trace either one pool or all pools. The value should be a valid pool number from 1 to 12. If the pool number is 0, all pools will be traced. This value is only supported in an AMODE64 environment.

In z/OS V1R12: there are no new options, or suboptions.

Because of these changes, you might have to perform migration actions.
Migration action:
- • If you are using CEEPRM*xx* to specify your installation-wide runtime options:
  - o The four suboptions added to HEAPCHK in z/OS V1R11 do not change the default behavior. You have no migration action.
  - o The HEAPPOOLS runtime option, which was added to CELQDOPT in z/OS V1R10, does not change the default behavior. You have no migration action.

  Note: In a future release, IBM plans to remove the capability to change the default Language Environment runtime options settings through SMP/E installable USERMODs. IBM recommends using the CEEPRM*xx* parmlib member to change the default Language Environment runtime options for the system.
- • If you are using the assembler modules to specify your installation-wide runtime options, compare your existing source for the installation-wide runtime options CSECT, CEEDOPT (non-CICS environment), CEECOPT (CICS environment), or CELQDOPT (AMODE 64) with the new samples in *hlq*.SCEESAMP to determine whether you need to change the defaults. Then, update (and reinstall) your usermods with the following new options:
  - o The four suboptions added to HEAPCHK in z/OS V1R11.
  - o The HEAPPOOLS runtime option, which was added to CELQDOPT in z/OS V1R10. But note that this function, which is integrated in z/OS V1R10, was rolled back to prior releases (APAR PK49427 on z/OS V1R9 and APAR PK57579 on z/OS V1R8 and V1R7), so you might have already updated this usermod.

Language Environment Migration Actions Post-First IPL
Examine programs that read output from a CICS CLER transaction (Required-IF, as of R12)
Required if if you have an application that reads the output of a CICS CLER transaction.
Starting with z/OS V1R12, the Language Environment runtime options report displayed from the CICS CLER transaction is changed. The report is modified to have a wider LAST WHERE SET column to accomodate longer values, such as "Installation Non-overrideable." In addition, the report heading OPTIONS is changed to OPTION to match the other runtime options reports.
Migration action: Examine programs that read the output of a CICS CLER transaction to ensure compatibility with the updated CLER runtime options report. The LAST WHERE SET column is now wider and the OPTIONS heading is changed to OPTION. The following is a subset of the new report to show the formatting changes:

```
LAST WHERE SET                 OPTION
---------------------------    ----------------------------------------------
Installation default           ABPERC(NONE)
Installation default           ABTERMENC(ABEND)
Installation default           NOAIXBLD
```

Use Unicode Services to create conversion tables (Required-IF, as of R12)
Required if you use the iconv() family of functions to test to a "known conversion result"and experience testcase failures. Also, if you use custom conversion tables replacing those listed in either ucmapt.lst or genxlt.lst.
Beginning with z/OS V1R12, the C/C++ runtime library will no longer include any ucmap source code or genxlt source code for character conversions now being performed by Unicode Services.
Migration action:
- • If you use customized conversion tables, you should now generate custom Unicode Services conversion tables.

- If you use the iconv() family of functions testing to a "known conversion result" and experience test case failures, you need to update your expected results to the new conversion results.
- If you want to create custom conversion tables involving any of the CCSIDs related to the conversion table source no longer being shipped, you should now generate custom Unicode Services conversion tables instead of custom Language Environment conversion tables.

The *installation prefix*.SCEEUMAP data set will no longer be shipped. The /usr/lib/nls/locale/ucmap HFS directory will no longer be shipped.

Note: The _ICONV_TECHNIQUE environment variable must be set to the same technique search order value used for the customized Unicode Services table in order for the iconv() family of functions to use the customized Unicode Services table. For example, if you want the iconv() family of functions to use a user-defined Unicode Services table with a technique search order of 2, the _ICONV_TECHNIQUE environment variable should be set to 2LMREC.

z/OS UNIX Migration Actions for z/OS R12 from z/OS R10

Migration Actions You Can Do Now:

- **Accommodate new Shell and Utilities version of the tsocmd command (Required-IF, as of R12)**
  - Before R12: tsocmd command was a Tools and Toys download.
  - As of R12: it is part of the Shell and Utilities support in z/OS.
  - Decide if you which version of tsocmd to use. Remove the "old" tsocmd if you want. R12 tsocmd download to be made available, so you can start using this version on old releases.
- **Migrate from HFS to zFS file systems (Recommended)**
  - HFS has been stabilized, and zFS is strategic.
  - Convert you file systems from HFS to zFS.
  - 1) Use ServerPac (during install, for the shipped file systems), 2) use BPXWH2Z (for after install, and other file systems you have), and 3) look at R10's F OMVS,NEWROOT= support for the sysplex root
  - **Planned for z/OS R12**: zFS data sets to be indirectly cataloged with a system symbol!

z/OS UNIX Migration Actions for z/OS R12 from z/OS R10

Migration Actions Pre-First IPL:

- **Use the BPX.UNIQUE.USER profile instead of BPX.DEFAULT.USER (Recommended, as of R11)**
  - Before R11: BPX.DEFAULT.USER were assigned a default OMVS segments for the length of the user session. They used the same UID/GID.
  - As of R11: If BPX.UNIQUE.USER is used, users will have unique UID/GID saved into their profiles in the RACF data base.
  - Follow steps in z/OS UNIX System Services Planning.

Migration Actions Post-First IPL:

- **Consider skulker invocations due to updated restriction (Required-IF, as of R12)**
  - Before R12: if there were objects whose path name included single quotes or newline character, those objects were not deleted, a warning message was issued, and processing stopped.
  - As of R12: there is no warning message, those objects are ignored, and processing continues.

z/OS UNIX Migration Actions for z/OS R12 from z/OS R10

Migration Actions Post-First IPL:

- **Remove MAXSOCKETS values from AF_UNIX in the BPXPRMxx parmlib member (Recommended, as of R12)**
  - Before R12: MAXSOCKETS was required in the NETWORK statement for AF_UNIX, if default of 100 was not acceptable.
  - As of R12: the maximum of 10,000 will be set. It will be ignored if specified on R12. Remove it when not sharing the BPXPRMxx member with pre-R12 systems.

- **Use man instead of OHELP (Required-IF, as of R11)**
  - Before R11: OHELP command was used to display online reference information for many commands and services.
  - As of R11: Use the z/OS UNIX **man** command instead. You do not need to delete the IBM-supplied /samples/ohelp.ENU file or the local /etc/ohelp.ENU file.

41   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Corporation

---

z/OS UNIX Migration Actions Between z/OS V1 R10 and z/OS V1 R12

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

z/OS UNIX System Services Migration Actions You Can Do Now

Accommodate new Shell and Utilities version of the tsocmd command (Required-IF, as of R12)
Required if you currently use the Tools and Toys version of the tsocmd command.

Before z/OS V1R11, the **tsocmd** command was obtained from the Tools and Toys section of the z/OS UNIX Web site. Starting with z/OS V1R12, Shell and Utilities support of the **tsocmd** command has been added. The supported version differs from the Tools and Toys version in a number of ways. For example:
- Exit values are consistently sent if the issued TSO/E command fails.
- The tsoin and tsoout environment variables are not supported. Instead, stdin and stdout are supported, as is done for most other Shell and Utilities commands.
- The TSOPROFILE environment variable is supported.

Migration action: Look for current use of the Tools and Toys version of **tsocmd**. If there is no current use of the Tools and Toys version of **tsocmd**, then no actions or changes are required.

If there is current use of the Tools and Toys version of **tsocmd**, determine if the command is located in /bin or in another directory. Also, determine if you want to preserve the Tools and Toys version in addition to the officially shipped version.
1. If you want to preserve the Tools and Toys version, ensure the Tools and Toys version of **tsocmd** is not located in /bin prior to the installation of z/OS V1R12.
2. If you do not want to preserve the Tools and Toys version, and it is located in /bin, then the installation of z/OS V1R12 will automatically replace the Tools and Toys version with the new officially supported version. If the Tools and Toys version is not located in /bin, remove it from its current location. In either case, you will also need to remove the Tools and Toys **tsocmd** load module from the authorized load library (either prefix.TSOCMD.LOADLIB or SYS1.LINKLIB) as described in the **tsocmd** Tools and Toys README documentation available at ftp://ftp.software.ibm.com/s390/zos/tools/ tsocmd/tsocmd.readme.txt.

Note: A new version of **tsocmd** that matches the version shipped with z/OS V1R12 will be made available on the Tools and Toys Web site. If you have multiple systems at different releases and you want to have the same version of the tool on all releases, or if you want to try out the new command prior to z/OS V1R12, you can download the new version to the earlier systems and replace the previous Tools and Toys version of **tsocmd** if appropriate. If this action is taken, you will need to clean up the authorized load library as described in Step 2 prior to downloading the new tool.

Migrate from HFS file systems to zFS file systems (Recommended)

Not required, recommended because the action is planned to become a requirement in a future release. zFS is the strategic file system for z/OS UNIX and continues to be enhanced to provide superior performance, reliability, and data integrity. IBM has not announced removal of support for the HFS file system.

Before z/OS V1R7, the HFS file system was the primary hierarchical file system. As of z/OS V1R7, you can use any combination of HFS and zFS file systems. Because zFS has higher performance characteristics than HFS and is the strategic file system, you should migrate your HFS file systems to zFS.

The HFS and zFS file system types in mount statements and command operands are now generic file system types that can mean either HFS or zFS. Based on the data set type, the system will determine which is appropriate. But note that you must still specify a type (HFS or zFS and it cannot be defaulted), and if the type you specify is not correct for the file system being mounted, any associated parameter string setting in the mount statement or command is ignored, even though the system sets the type correctly and processes the mount.

Understand the zFS recommendations and limits. For more information, see Minimum and maximum file system sizes in *z/OS Distributed File Service zSeries File System Administration*

Migration action:

1. Before beginning the migration, do the following:
   - Establish backout procedures.
   - Decide on naming conventions.
   - Decide on unavailability.
   - Understand any cloning or deployment changes required by zFS systems being linear data sets. Considerations would include any copy utility invocations, BPXPRMxx specifications for symbolics, and placement of zFS file systems on system volumes.

2. Perform the conversion from an HFS to zFS file system.

Tip: Use the BPXWH2Z tool to perform the conversion. It is an ISPF-based tool that migrates HFS file systems to zFS file systems. Using its panel interface, you can alter the space allocation, placement, SMS classes, and data set names. A HELP panel is provided. With this tool, you can:
   - Migrate HFS file systems (both mounted and unmounted) to zFS file systems. If the HFS being migrated is mounted, the tool automatically unmounts it and then mounts the new zFS file system on its current mount point.
   - Define zFS aggregates by default to be approximately the same size as the HFS. The new allocation size can also be increased or decreased.
   - Have the migration run in TSO foreground or UNIX background.

You can use the JCL sample ISPBTCH in SYS1.SAMPLIB to invoke BPXWH2Z as an ISPF batch job. Before you run the job, be sure to read the Notes section. When you run BPXWH2Z on your z/OS V1R12 system, it uses the z/OS V1R12 level of the **pax** command. This level was enhanced in z/OS V1R7 for sparse file support and other characteristics that are of concern when migrating from an HFS to zFS file system. You can manually migrate from an HFS to zFS file system without using the tool. However, you would need to allocate and format the target zFS file systems.

Requirement: The BPXWH2Z tool requires the zFS address space to be operational. Therefore, before attempting to migrate existing HFS to zFS file systems using BPXWH2Z, make sure that the zFS address space has been successfully configured and initialized.

Tip: Consider migrating the sysplex root file system to zFS after you IPL z/OS V1R12. You can dynamically migrate the HFS sysplex root in a shared file system configuration to zFS while the root is in use, without disrupting workloads. Before z/OS V1R10, you could not migrate without disrupting active workloads. Although the shared file system configuration is required, the sysplex can be a single system. For the steps involved in migrating the sysplex root file system from HFS to zFS, see below.

Tip: Use check USS_HFS_DETECTED to verify all file systems mounted. This check issues exception message BPXH068E if any HFS file systems are found.

SHARE
Boston, MA
© 2010 IBM Corporation

Page 89 of 94

August 4, 2010

Session 7398

Planned for z/OS R12! DFSMSdfp is planned to allow a zFS data set to be recataloged with an indirect volume serial or system symbol. This is designed to allow the zFS file systems used for z/OS system software files (called version root file systems) to be cataloged using an indirect volume serial or a system symbol the same way as non-VSAM data sets to make cloning and migration easier.

3. Change policies and scripts, and so forth, to reflect the change from the HFS file system to zFS file system.
Tip: Use the RMF Monitor III option to report on zFS activity.

Migrating the sysplex root file system from HFS to zFS after IPLing z/OS V1R12:
Before you begin the migration:
- Ensure that the following requirements have been met:
    - All systems in the sysplex are at the V1R12 level.
    - The current sysplex root file system PFS, and the new sysplex root file system PFS, are up in all the systems in shared file system configuration.
- Be aware of the following restrictions:
    - The current sysplex root file system must be mounted as a read-only file system.
    - The systems that do not meet the requirements for this migration action cannot join the sysplex during the sysplex root file system migration processing, but they can join the sysplex after the sysplex root migration has completed.
    - The current sysplex root and the new sysplex root must be either HFS or zFS in any combination. If the new sysplex root is zFS, then it must be HFS-compatible.
    - The sysplex root or any directories on it cannot have been exported by the DFS or SMB server.
- Note the following:
    - Remote NFS mounts of the sysplex root or any directories on it are considered active use of the current sysplex root file system.
    - During the migration, the new zFS sysplex root file system must not be HSM-migrated, mounted, or in use.
    - Mount parameters are preserved during the migration or replacement of the sysplex root file system of the same file system type (PFS). They are dropped if the file system type is different.
    - Directories, data, files, and links are not copied from one file system to another.

Perform the migration as follows:
1. Ensure that a file system has been mounted read-only as the current sysplex root file system. When the root is mounted read-only, there are no function-shipping clients as long as physical paths to the DASD are available to each system. To verify that there are no function-shipping clients, issue: D OMVS,F,NAME=root_file_system_name
You should see CLIENT=N on each system.

2. Allocate and set up the new zFS sysplex root file system:
- Create a new zFS file system to be used as the new sysplex root file system. *z/OS Distributed File Service zSeries File System Administration* discusses creating and managing zFS file systems.
    Rules:
    - The UID, GID and the permission bits of the root directory in the new sysplex root file system must be same as the root directory in the current sysplex root file system.
    - If the SECLABEL class is active and the MLFSOBJ option is active, the security label for the new zFS file system must match the assumed security label of the current sysplex root file system.
- On the new sysplex root file system, set up the active mount points and the symbolic links. The mount points and symbolic links must be the same as the ones on the current sysplex root file system. You can set them up either (1) manually or (2) by using the **pax** shell command to populate the new sysplex root file system using the existing sysplex root as a source. To do it manually, create a mount point in the existing sysplex root (for example, /newroot) and mount the new sysplex root file system in the MODE(RDWR) on that mount point. After mounting the new sysplex root file system, manually issue MKDIRs and ln -s to create the mount point directories and symbolic links similar to the existing sysplex root file system. Note that the new sysplex root file system must contain all active mount points and symbolic links exactly as on the existing sysplex root file system.
- Use the **pax** shell command to populate the new file system, using the existing sysplex root as a source.
    Example:
    cd /
    pax -wr -pe -XCM ./ /newroot

For more information about using **pax** to copy data from an HFS file system to a zFS file system, see
z/OS Distributed File Service zSeries File System Administration.
- Unmount the new zFS file system.

3. Before proceeding with this step, ensure PTF UA49782 for APAR OA29567 is installed on your z/OS V1R10 systems. Then, on any system in the shared file system configuration, issue:
F OMVS,NEWROOT=new.root.file.system.name,COND=<Yes|No>

> **YES**   Proceed conditionally. The system checks for active usage in the current sysplex root file system and reports the active usage in a BPXF245I message. If file activity is found, the command fails with EBUSY return code and JrActivityFound reason code. If file activity is not found, the command continues processing to replace the sysplex root. YES is the default.
> **NO**   Proceed unconditionally. The system checks for active usage in the current sysplex root file system and reports the active usage in a BPXF245I message. Replacement of the sysplex root file system will continue.

The migration of the sysplex root file system will begin. During the migration, active connections to files and directories in the current sysplex root file system are broken.
After the migration completes:
- The root CWD('/') is updated on all systems in the sysplex to point to the new sysplex root file system.
- New opens go to the new sysplex root file system. The current sysplex root for the root directory is replaced for all processes in all systems. The current directory for root directory is replaced for any processes using it.
- Old connections in the previous sysplex root file system might get EIO errors.

4. Update the TYPE parameter and name of the sysplex root file system in the BPXPRM*xx* member of SYS1.PARMLIB.

z/OS UNIX System Services Migration Actions Pre-First IPL
Update automated actions for z/OS UNIX messages (Required-IF, as of R11)
Required if you have automation that is affected by the message changes.
The following z/OS UNIX message have changed:
- Before z/OS V1R11, message BPXO040I was issued in response to one of the following DISPLAY OMVS commands: D OMVS,A, D OMVS,U, D OMVS,PID=
  Beginning with z/OS V1R11, message BPXO070I is issued instead.
- Before z/OS V1R11, message BPXO046I was issued in response to the DISPLAY OMVS,PFS command. As of z/OS V1R11, message BPXO068I is issued.

z/OS UNIX System Services Migration Actions Post-First IPL
Consider skulker invocations due to updated restriction (Required-IF, as of R12)
Required if you are concerned about not getting a warning message if objects in the specified directory or subdirectories whose path name includes single quotes or a newline character are found.
Before z/OS V1R12, if the **skulker** command was issued and there were objects in the specified directory (and in any subdirectories if the **-r** option was used) whose path name included single quotes or a newline character, those objects were not deleted, a warning message was issued, and all processing stopped. Starting with z/OS V1R12, a warning message is not issued, those objects are ignored, and processing continues.
Migration action: Use the **find** command to determine if there are any objects in the specified directory or subdirectories whose path name includes single quotes or a newline character. Do not use the **find** command to locate the objects whose path name include single quotes or a newline character as a standard practice before using the **skulker** command if you want the directories removed because issuing the **find** command will modify the access times of directories that are part of that **find** request.

Remove MAXSOCKETS values from AF_UNIX in the BPXPRMxx parmlib member (Recommended, as of R12)
Not required, but recommended to avoid confusion if users do not know that a value does not need to be specified for the MAXSOCKETS keyword on the NETWORK statement for AF_UNIX for z/OS V1R12.
Before z/OS V1R12, a value had to be specified for the MAXSOCKETS keyword in the NETWORK statement for AF_UNIX in the BPXPRM*xx* parmlib member of SYS1.PARMLIB if the maximum number of AF_UNIX sockets for the system needed to be greater than the default of 100. As of z/OS V1R12, the value does not need to be specified

because a maximum value of 10,000 has been set for MAXSOCKETS for AF_UNIX. The MAXSOCKETS keyword is still allowed on the NETWORK statement for AF_UNIX, but will be ignored on a z/OS V1R12 system.

If you share your BPXPRM*xx* parmlib member with lower level systems, only remove your MAXSOCKETS statement from your AF_UNIX domain if you will accept the default MAXSOCKETS value for AF_UNIX for those lower level systems. (The default on z/OS V1R11 and z/OS V1R10 for MAXSOCKETS in the AF_UNIX domain is 100.) Otherwise, remove the MAXSOCKETS statement from the AF_UNIX domain when all systems that share the BPXPRM*xx* parmlib member are at z/OS V1R12 or later.
Migration action: Remove any MAXSOCKETS statement from NETWORK statements for DOMAINNAME(AF_UNIX). In a configuration with a shared BPXPRM*xx* parmlib, MAXSOCKETS should only be removed when all systems are at z/OS V1R12.

Use man instead of OHELP(Required-IF, as of R11)
Required if you use the OHELP command.
Previously, the TSO/E OHELP command was used to display online reference information for shell commands, TSO/E commands, C functions, callable services, and messages issued by the shell and **dbx**. It required the use of bookshelf provided by the *z/OS Collection Kit*. In V1R11, the OHELP command is no longer available.
Migration action: Use the z/OS UNIX **man** command instead. You do not need to delete the IBM-supplied /samples/ohelp.ENU file or the local /etc/ohelp.ENU file.

Use the BPX.UNIQUE.USER profile instead of BPX.DEFAULT.USER (Recommended, as of R11)
Recommended because in the future, support for the BPX.DEFAULT.USER profile might be removed. The use of BPX.UNIQUE.USER increases security because shared IDs are not used.
Before z/OS V1R11, if the BPX.DEFAULT.USER profile in the FACILITY class was defined, users who accessed z/OS UNIX services who did not have an OMVS user or group segment were assigned the default OMVS segments for the length of the user session. All users of the default OMVS segments shared the same UID and GID. As of z/OS V1R11, if BPX.UNIQUE.USER has been defined, users who access z/OS UNIX services who do not have an OMVS user or group segment are automatically assigned an OMVS segment with a unique UID and GID. The new OMVS segments are added to the user and group profiles in the RACF database.
Migration action: Follow the steps in *z/OS UNIX System Services Planning* to set up the BPX.UNIQUE.USER profile. If BPX.DEFAULT.USER has not been deleted, BPX.UNIQUE.USER takes precedence when default OMVS segments are used.
To remove the BPX.DEFAULT.USER profile, use the following RACF commands:
        RDELETE FACILITY BPX.DEFAULT.USER
        SETROPTS RACLIST(FACILITY) REFRESH

Update automation that handles ISPF Edit and ISPF Browse (Required-IF, as of R11)
Required if you have automation that depends on the display of OEDIT and OBROWSE panels.
Before z/OS V1R11, the TSO/E commands OEDIT and OBROWSE displayed panels that were different than the conventional ISPF Edit and ISPF Browse panels and used the ISPF EDIF and BRIF services. As of z/OS V1R11, OEDIT and OBROWSE will no longer display their unique panels and will directly invoke the ISPF EDIT and BROWSE commands.
Migration action: Update any automation that depends on the display of panels by OEDIT and OBROWSE.

Marna's "Big Migs" for z/OS R12 from z/OS R10

Migration Actions You Should NOT Overlook:

1. **Install coexistence and fallback PTFs (Required)**
2. **Add or change volumes to keep file systems in a single data set (Recommended)**
3. **Accept CustomPac default file system of zFS (Recommended)**
4. **Ensure your ESQA specification in IEASYSxx is adequate (Req-IF, R11)**
5. **Review the list of WTORs in parmlib member AUTOR00 (Req-IF, as of R12)**
6. **Reassemble the standalone dump program (Required, changed in R12)**
7. **Update automation for some deleted SMF messages (Req-IF, as of R12)**
8. **Accommodate DEFAULT route usage changed for CINET environments (Required-IF, as of R11)**
9. **Run the syslog daemon in the daemon in the background when starting from UNIX (Required-IF, as of R11)**

Marna's "Big Migs" for z/OS R12 from z/OS R10

Migration Actions You Should NOT Overlook:

10. **ICSF HCR7770: Modify ICSF startup procedure (Required, as of HCR7770, incorporated into R12)**
11. **Determine if you will be using the IBM TDS WLM support (Req-IF, R11)**
12. **Install the new HCM (Required-IF, as of R11)**
13. **Ensure that sysplex_admin_level=2 is available on all systems (Required-IF, as of R11)**
14. **Permit the SMB server ID to BPX.DAEMON and BPX.SERVER (Required-IF, as of R11)**
15. **Identify the JES3 auxiliary address space to your security product (Required-IF, as of R11)**
16. **Increase space in the Printer Inventory file system (Required-IF, as of R12)**
17. **Remove Version 2 Printer Inventory files at fallback (Recommended, as of R12)**

Migrating to z/OS 1.12: Part 2 of 2 Summary

- **General Migration Actions for Everyone**:
  - HWIBCPII and ARC*n*RST*y* are new address spaces; SYS1.SCUNIMG is removed; TCPIP.SEZAEXEC is new; Changed Health Checks.
- **BCP Migration Actions:**
  - GRS BUFSIZE default change; IFASMFDL and IFASMFDP changes; zAAP on zIIP; LCCA and PCCA default location; REUSEASID default change; Sample IEFACTRT field length change; PFA install script specification.
- **Communications Server Migration Actions:**
  - Removed functions; Telnet stopping when OMVS is shutdown; Syslogd archiving in a unique location; Syslogd job name change; Dump data sets big enough for VIT; Increase region size for VTAM; new resolver behavior for DNS queries, monitoring name server responsiveness, and UDP timeouts; Netstat message catalog to include timestamp; Telnet messages have changed.

44   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IBM Co...

Migrating to z/OS 1.12: Part 2 of 2 Summary

- **DFSMS Migration Actions:**
  - Remove old VSAM attributes; EDGRMMxx VRSEL(OLD) is removed; Dss applications to handle larger I/O buffers; Remove patch for dss in ARCCMDxx; Keyword on (H)BDELETE; Rmm parmlib instead of usermod for duplicate GDGs; ADRDSSU region size; DEVMAN permission for ICKDSF; Rmm dynamic exits; Indirect zFS catalog support; security setup for (H)BACKDS RETAINDAYS.
- **DFS Migration Actions:** Unmount may fail on some file systems.
- **Infoprint Server:** Remove V1 files after fully on R12.
- **JES2 Migration Actions:** References to PDBLENG; Info on multiple members; Migrate to z11 mode (more BERTs needed?).
- **JES3 Migration Actions:** IATUX72 changes; STVSTOD value change.
- **LE Migration Actions:** New runtime sub-options; NONOVR and OVR support in CEEPRMxx; Report and response changes; Unicode conv tables.
- **z/OS UNIX Migration Action:** Tsocmd; Move from HFS to zFS; Use BPX.UNIQUE.USER instead of BPX.DEFAULT.USER; Skulker changes; MAXSOCKETS can be removed; Use man instead of OHELP.

45   Migrating to z/OS 1.12: Part 2 of 2   © 2010 IB... ...on