z/OS 1.13 UNIX System Services Latest Status and New Features

Richard Theis (rtheis@us.ibm.com)
IBM Rochester, MN

Session 9721
August 9, 2011

Trademarks and Disclaimers

- **See http://www.ibm.com/legal/copytrade.shtml for a list of IBM trademarks.**
- **The following are trademarks or registered trademarks of other companies**
  - UNIX is a registered trademark of The Open Group in the United States and other countries
  - CERT® is a registered trademark and service mark of Carnegie Mellon University.
  - ssh® is a registered trademark of SSH Communications Security Corp
  - X Window System is a trademark of X Consortium, Inc
- **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.

Agenda

Nonprivileged user mount

Additional mount enhancements

File system access check

New script command

ASCII option for vi command

D OMVS,W enhancement

Lost message detection

__console2() enhancements

IPv4 IP_PKTINFO support

Symbolic links to shared root

Appendix

Nonprivileged user mount

- **Problem Statement**
  - Mounting a UNIX file system is a privileged operation. Yet it would be nice to move the management of user data away from the system administrators and into the hands of the users that own the data.
- **Solution**
  - Provide capability to permit nonprivileged users to mount and unmount file systems based on the user's authority to the mount point and the file system root.
  - Only supported in BPX2MNT (mount2) interface (e.g.: /usr/sbin/mount, TSO Mount commands).

Nonprivileged user mount

- **Benefits**
  - System administrators will no longer have the burden of end user file system data management.
  - Application developers/End users will now be able to manage their own data.

Nonprivileged user mount

- **Mount Security Requirements:**
  - Read access to SUPERUSER.FILESYS.USERMOUNT UNIXPRIV profile
  - Read-Write-Execute (RWX) access permission to the mount point directory
    - If Sticky bit is set, then the user must be the owner of mount point directory
  - Mount point directory must be empty
  - Read-Write-Execute(RWX) access permission to the file system root directory
    - If Sticky bit is set, then the user must be the owner of file system root

Nonprivileged user mount

- **Unmount Security Requirements:**
  - Read access to SUPERUSER.FILESYS.USERMOUNT profile
  - Must be the user who mounted the file system

Nonprivileged user mount

- **Restrictions:**
  - Supported file system types are HFS, zFS and NFS
  - SYSNAME option is **not** supported
  - NOSECURITY option **cannot** be specified
  - NOSETUID option **must** be specified
  - chmount is **not** supported for nonprivileged users
  - Remount is **not** supported for nonprivileged users
  - Use of /// as a file system name placeholder is **not** supported
  - BPX1MNT callable service is **not** supported for nonprivileged users
  - Mount operation is bounded by MAXUSERMOUNTSYS & MAXUSERMOUNTUSER BPXPRMxx PARMLIB limits.

Nonprivileged user mount

- **Setup:**
  - Permit users to the new UNIXPRIV SUPERUSER.FILESYS.USERMOUNT profile
  - Set following new keywords in BPXPRMxx:
    - MAXUSERMOUNTSYS( ):
      - *Use the MAXUSERMOUNTSYS statement to specify the maximum number of nonprivileged user mounts in the system or in shared file system configuration.*
    - MAXUSERMOUNTUSER( ):
      - *Use the MAXUSERMOUNTUSER statement to specify the maximum number of nonprivileged user mounts allowed for any nonprivileged user in the system or in shared file system configuration.*

Nonprivileged user mount

- Display nonprivileged user mount information
  - **D OMVS,FILE**

```
BPXO045I 13.37.38 DISPLAY OMVS 589
OMVS     000E ACTIVE              OMVS=(Y2,3Z)
TYPENAME   DEVICE ----------STATUS----------- MODE  MOUNTED    LATCHES
ZFS             19 ACTIVE                     RDWR  07/22/2010  L=27
  NAME=MYFS1.ZFS                                    13.37.25    Q=0
  PATH=/u/myzfs1/mntzfs
  UID=295
ZFS              3 ACTIVE                     RDWR  07/22/2010  L=15
  NAME=ZOS113.VAR.ZFS                               12.46.09    Q=0
  PATH=/SYSTEM/var
```

Nonprivileged user mount

- Display nonprivileged user mount information
  - **MODIFY BPXOINIT,FILESYS=DISPLAY,FILESYSTEM=**

```
BPXF035I 2010/07/22 14.50.59 MODIFY BPXOINIT,FILESYS=DISPLAY 847
-------------NAME-------------------------        DEVICE  MODE
ZOS113.SYSPLEX.ROOT.ZFS                                1   RDWR
  PATH=/
  STATUS=ACTIVE                LOCAL STATUS=ACTIVE
  ........
  LOCAL FLAGS=40000212  LOCAL LFSFLAGS=22000000
  ACTIVECHK  =00000000  LFSFLAGS2     =D8000000
MYFS1.ZFS                                             16   RDWR
  PATH=/u/myzfs1/mntzfs
  UID=295
  STATUS=ACTIVE                LOCAL STATUS=ACTIVE
  ........
  LOCAL FLAGS=40000400  LOCAL LFSFLAGS=02000000
  ACTIVECHK  =00000000  LFSFLAGS2     =D8000000
```

Nonprivileged user mount

- Display nonprivileged user mount information using filters
  - D OMVS,FILE,UID=<euid|USER|PRIV>
  - **D OMVS,F,UID=USER**

```
BPXO045I 13.37.38 DISPLAY OMVS 589
OMVS     000E ACTIVE              OMVS=(Y2,3Z)
TYPENAME   DEVICE ----------STATUS----------- MODE  MOUNTED    LATCHES
ZFS             19 ACTIVE                     RDWR  07/22/2010  L=27
  NAME=MYFS1.ZFS                                    13.37.25    Q=0
  PATH=/u/myzfs1/mntzfs
  UID=295
HFS             20 ACTIVE                     RDWR  07/22/2010  L=23
  NAME=MYFS1.HFS                                    13.37.28    Q=0
  PATH=/u/myhfs1/mnthfs
  UID=47
```

Nonprivileged user mount

- Display nonprivileged user mount information using filters
  - **D OMVS,F,UID=PRIV**

```
BPXO045I 13.38.38 DISPLAY OMVS 592
OMVS     000E ACTIVE              OMVS=(Y2,3Z)
TYPENAME   DEVICE ----------STATUS----------- MODE  MOUNTED    LATCHES
ZFS             3 ACTIVE                      RDWR  07/22/2010  L=15
  NAME=ZOS113.VAR.ZFS                               12.46.09    Q=0
  PATH=/SYSTEM/var
ZFS             2 ACTIVE                      RDWR  07/22/2010  L=14
  NAME=ZOS113.ETC.ZFS                               12.46.09    Q=0
  PATH=/SYSTEM/etc
```

Nonprivileged user mount

- Display nonprivileged user mount information and settings
  - **DISPLAY OMVS,USERMOUNTS**

```
BPXO072I 13.28.20 DISPLAY OMVS 544
OMVS     000E ACTIVE             OMVS=(Y2,3Z)
NONPRIVILEGED USER MOUNTS SUMMARY
          UID   CURRENT MOUNTS
          295           1
           47           2
           25           1
```

Nonprivileged user mount

- Display nonprivileged user mount information and settings
  - **D OMVS,OPTIONS**

```
BPXO043I 09.56.54 DISPLAY OMVS 513
OMVS     000E ACTIVE              OMVS=(Y4)
CURRENT UNIX CONFIGURATION SETTINGS:
MAXPROCSYS       =        256    MAXPROCUSER     =        300
 ........
AUTHPGMLIST      = NONE
SWA              = BELOW
SERV_LINKLIB     =
SERV_LPALIB      =
MAXUSERMOUNTSYS  = 100          MAXUSERMOUNTUSER = 10
```

Nonprivileged user mount

- Display nonprivileged user mount settings and high-water marks
  - **D OMVS,LIMITS**

```
SY1  BPXO051I 19.35.21 DISPLAY OMVS 896
OMVS     000E ACTIVE              OMVS=(Y8,MZ)
SYSTEM WIDE LIMITS:          LIMMSG=NONE
                   CURRENT  HIGHWATER     SYSTEM
                     USAGE       USAGE      LIMIT
MAXPROCSYS               7           9        900
MAXUIDS                  2           2        200
.......
SHRLIBRGNSIZE            0           0   67108864
SHRLIBMAXPAGES           0           0       4096
MAXUSERMOUNTSYS         15          20        100
MAXUSERMOUNTUSER         7           8         10
```

Nonprivileged user mount

- Display nonprivileged user mount settings and high-water marks
  - **MODIFY BPXOINIT,FILESYS=DISPLAY,GLOBAL**

```
SY2  BPXM027I COMMAND ACCEPTED.
SY2  BPXF040I MODIFY BPXOINIT,FILESYS PROCESSING IS COMPLETE.
SY2  BPXF242I 2010/07/22 14.44.10 MODIFY BPXOINIT,FILESYS=DISPLAY,GLOBAL
SYSTEM   LFS VERSION ---STATUS--------------------- RECOMMENDED ACTION
.......
MAXIMUM MOUNT ENTRIES=     500   MOUNT ENTRIES IN USE=      14
MAXIMUM AMTRULES=           50   AMTRULES IN USE=            0
MAXUSERMOUNTSYS=           100   HIWATER MAXUSERMOUNTSYS=   20
MAXUSERMOUNTUSER=           10   HIWATER MAXUSERMOUNTUSR=    8
MAXSYSTEM=                   8
ALTROOT=  N/A
```

Nonprivileged user mount

- ISHELL display for nonprivileged user mounted file system
  - **ISHELL Panel of File System Attributes**

```
                          Work with Mounted File Systems
.-------------------------------------------------------------.
| BPXWP22           File System Attributes                    |
|                                                             |
| File system name:                                           |  Row 1 of 5
| MYFS.ZFS                                                    |
| Mount point:                                                |
| /u/zfs1                                                     |
|                                                  More:  - + |
| Data blocks read . . . : 0                                  |
| Data blocks written  . : 0                                  |
| Dir blocks r/w . . . . : 0                                  |
| User . . . . . : WELLIE(25)                                 |
| Char Set ID/Text flag  :                                    |
| Seclabel . . . . . :                                        |
| Mount parameter:                                            |
|    _____________________________________________________    |
|    _____________________________________________________    |
|    _____________________________________________________    |
|    _____________________________________________________    |
|    _____________________________________________________    |
|                                                             |
|  F1=Help        F3=Exit        F4=Name      F6=Keyshelp     |
| F12=Cancel                                  F7=Backward     |
'-------------------------------------------------------------'
```

Nonprivileged user mount

- ISHELL display for nonprivileged user mounted file system
  - **ISHELL Panel of Mounted File Systems**

```
                    Work with Mounted File Systems


Select one or more file systems with / or action codes.

U=Unmount   A=Attributes   M=Modify   R=Reset unmount or quiesce

  Type  Owner     A/M  R/O  User          F  Row 1 of 9
_ ZFS   SY1       U    No                   ZOS113.SY1.ETC.ZFS
_ ZFS   SY1       U    No    WELLIE         MYFS.ZFS
_ ZFS   SY1       U    No                   ZOS113.SY1.VAR.ZFS
```

Nonprivileged user mount

- df command display for nonprivileged user mounted file system
  - **df –v /u/rtheis**

```
Mounted on     Filesystem                Avail/Total    Files       Status
/u/rtheis      (OMVS.ZFS.RTHEIS)         778020/3867840 4294949652 Available
ZFS, Read/Write, Device:429465, ACLS=Y
File System Owner : AQFT        Automove=Y      Client=N
Filetag : T=off   codeset=0
Aggregate Name : OMVS.ZFS.RTHEIS
User : rtheis(2821)
```

Additional mount enhancements

- NONEMPTYMOUNTPT (NOWARN|WARN|DENY)
  - Provides a new BPXPRMxx PARMLIB statement to control non-empty mount point directory contents overlay during mount operations
  - Can be dynamically changed using SET OMVS or SETOMVS commands

Additional mount enhancements

- Display non-empty mount settings
  - **D OMVS,OPTIONS**

```
BPXO043I 09.56.54 DISPLAY OMVS 513
OMVS     000E ACTIVE              OMVS=(Y4)
CURRENT UNIX CONFIGURATION SETTINGS:
MAXPROCSYS       =        256    MAXPROCUSER     =        300
 ........
AUTHPGMLIST      = NONE
SWA              = BELOW          NONEMPTYMOUNTPT = DENY
SERV_LINKLIB     =
SERV_LPALIB      =
MAXUSERMOUNTSYS  = 100            MAXUSERMOUNTUSER = 10
```

Additional mount enhancements

- **/usr/sbin/mount Shell Command Updates:**
  - If waiting for an asynchronous mount to complete, failures will now be reported to the user. (This also applies to TSO Mount)
  - File system type is dynamically determined if "–t" (type option) not used and "–o" (fsoptions) was specified. This is for HFS and zFS only.
  - Now verifies file system name length and fails the mount if larger than 44 characters
  - Does not check for existence of path so mount syscall is issued that will fail and create a mount failure record
  - File system name is uppercased if –t (type option) not used and type determined to be zFS

Additional mount enhancements

- **/usr/sbin/unmount Shell Command Updates:**
  - Default behavior changed to unmount a file system only if the path specified is a mount point
  - New –m option created to retain default behavior (path specified can be any file/directory contained in the file system)

Additional mount enhancements

- **Migration and Coexistence Considerations**
  - Default behavior changed for /usr/sbin/unmount to require path name specified to be a mount point
  - Mounts via /usr/sbin/mount of zFS file systems may start to fail if –t option (type) not used and –o option (fsoption) was used and the fsoption were HFS parameters

File system access check

- **Problem Statement**
  - Ability for the RACF administrator to control access to UNIX System Services mounted file systems without needing to use UNIX semantics.
- **Solution**
  - Implement a new FASTAUTH call in ck_access that checks user's permission to file system profile in a new class (FSACCESS).
  - Supported for zFS.

File system access check

- **Benefits**
  - Provides a coarse-grained file system access control for use by the RACF administrator.
  - Does not require UNIX command expertise (setfacl) for administration.
  - Provides compliance and audit verification for RACF-centric customers.

File system access check

- **Mount point traversal triggers one-time check to the container**
  - Access failure prevents any operation within file system, regardless of permission bits, acls, file ownership, or UID(0)
  - Successful access (or no covering profile) simply continues with existing UNIX-style checks which may or may not allow access to file system object
  - RACF AUDITOR attribute bypasses FSACCESS check
- **UPDATE access required to (new) FSACCESS class resource name which equals the containing data set name**
  - Only performed if FSACCESS class is active
  - Use of generics for resource name is supported

File system access check

- **RACF setup:**

```
RDEFINE FSACCESS IBMUSER.ZFS.SRC UACC(NONE)
PERMIT IBMUSER.ZFS.SRC CLASS(FSACCESS)
   ID(IBMUSER)ACCESS(UPDATE)
SETROPTS CLASSACT(FSACCESS)
SETROPTS RACLIST(FSACCESS)
```

- New function available in Releases 12 and 13.
- New function enabled when FSACCESS class is activated.
- The following set of APARs are required:
  - OA35973 RACF
  - OA35974 SAF
  - OA35970 UNIX

New script command

- **Problem Statement**
  - As a z/OS S&U user, I want to record my shell session activity so that I can use the record for troubleshooting and documenting purposes.
- **Solution**
  - Provide the /bin/script command for z/OS 1.13. The **script** command makes a typescript of everything displayed on the terminal.

New script command

- **Benefits**
  - Records shell session activity in a way that is similar to other UNIX and Linux platforms.
  - Works with the /bin/sh and /bin/tcsh shells.
  - Works in the OpenSSH, rlogin, telnet and OMVS environments.

New script command

- The **script** command isn't listed in the UNIX standards (SUSv2 or SUSv3).
- z/OS **script** command is similar to the AIX **script** command.
- **Command usage: script [-aq] [file]**
  - 'a' option: Appends the typescript to the file (default is to overwrite).
  - 'q' option: Quiet mode. In quiet mode, all diagnostic messages are suppressed.
  - 'file' parameter: Typescript is written to the file specified by the file parameter (default is ./typescript).

New script command

- **Usage notes**
  - **script** forks and executes a shell according to the value of the SHELL environment variable. If the environment variable is not set, script uses the /bin/sh shell. script ends when the shell process exits. Use either exit or Ctrl-D to exit the shell process.
  - Because **script** writes everything in the typescript to the file including backspaces and prompts, commands that modify terminals such as **vi** might create unexpected data in the typescript.
  - Before and after running **script**, ensure that access to the file containing the typescript is properly controlled because the file might contain sensitive data.

New script command

- **Usage notes**
  - **script** does not support setting 3270 passthrough mode during the shell session. As a result, OEDIT, OBROWSE, and other utilities requiring 3270 passthrough mode will fail.
  - **script** creates a new session and controlling terminal for the shell process. A login accounting entry is not added to /etc/utmpx for this session and terminal.
  - **script** cannot be run in a background process.
  - Do not access the typescript file in use by **script** during the shell session, or unexpected results might occur.

New script command

- **Example session activity**

```
RCHSHUT3: /tst/usr/wellie6> script
Script command is started.  The file is typescript.
RCHSHUT3: /tst/usr/wellie6> rm -v removeme*
removeme1
removeme2
removeme3
RCHSHUT3: /tst/usr/wellie6> exit
Script command is complete.  The file is typescript.
```

- **Example typescript generated**

```
Script command is started on Mon Jan 11 13:52:04 2011.
RCHSHUT3: /tst/usr/wellie6> rm -v removeme*
removeme1
removeme2
removeme3
RCHSHUT3: /tst/usr/wellie6> exit
Script command is complete on Mon Jan 11 13:52:21 2011.
```

ASCII option for vi command

- **Problem Statement**
  - As a z/OS S&U user, I want manual control of the text conversion status of files that I edit or browse using the **vi** and **ex** editors.
- **Solution**
  - New **–W filecodeset=***codeset*,**pgmcodeset=***codeset* option on the **vi** and **ex** editors to enable text conversion
  - New **–B** option on the **vi** and **ex** editors to disable automatic text conversion – this is consistent with other commands that already have this override support.

ASCII option for vi command

- **Benefits**
  - More detailed control of text conversion
    - No file tagging required
    - No environment or system setup required
  - Easily override the system's automatic text conversion

ASCII option for vi command

- New **–W** option[,option]... option added to the **vi** and **ex** commands

  Specifies z/OS-specific options. The option keywords are case sensitive. Possible options are: **filecodeset=***codeset* and **pgmcodeset=***codeset*

- New **–B** option added to the **vi** and **ex** commands

  Disables the automatic text conversion of tagged files. This option is ignored if the **filecodeset** or **pgmcodeset** options (**-W** option) are specified.

ASCII option for vi command

- New **–W filecodeset=***codeset* option details

  Performs text conversion from one code set to another when reading from or writing to the file. The coded character set of the file is *codeset*. *codeset* can be a code set name known to the system or a numeric coded character set identifier (CCSID). The **filecodeset** and **pgmcodeset** options can be used on files with any file tag.

  If **pgmcodeset** is specified but **filecodeset** is omitted, then the default file code set is ISO8859-1, even if the file is tagged with a different code set.

  Restriction: The only supported values for **filecodeset** are ISO8859-1 and 819.

ASCII option for vi command

- New **–W pgmcodeset=***codeset* option similar to the **filecodeset=***codeset* option except for the following:
  - Represents the editor program's code set
  - Default program code set is IBM-1047
  - Only supported values for **pgmcodeset** are IBM-1047 and 1047

ASCII option for vi command

- Note the following precedence rules:
  - The **–W filecodeset=***codeset*,**pgmcodeset=***codeset* option overrides the **–B** option and the system's automatic text conversion
  - The **–B** option disables the system's automatic text conversion
  - If the **–W filecodeset=***codeset*,**pgmcodeset=***codeset* and **–B** options aren't specified then the system's automatic text conversion rules apply
- The **vi** and **ex –r** option (i.e. recover) must use the same text conversion options to ensure proper file recovery

D OMVS,W enhancement

- **Problem Statement**
  - File latch contention isn't displayed in its own table like file system latch contention. File latch activity will currently only show up in the other waiters table. This makes it hard to correlate the waiters and holders causing file latch contention. The file latch holder isn't necessarily displayed either unless it is waiting on something else.
  - On a live system, the information displayed on the D OMVS,W command can include many waiters. There is currently no way to limit the output being displayed. A filtering option is needed on the command to view only what is needed.

D OMVS,W enhancement

- **Solution**
  - D OMVS,W display has been enhanced to include a table for file latch activity. This table is similar to the existing one for file system latches and also displays the USER, ASID and TCB of the holder or waiter, whether the latch is held exclusive or shared, the AGE etc.
  - A set of filtering options have been added to the command in order to limit what gets displayed.

D OMVS,W enhancement

- **Benefits**
  - Easier to identify and debug file latch contention.
  - Easily limit the amount of data being displayed to help in problem determination.

D OMVS,W enhancement

- **DISPLAY OMVS,WAITERS – system command**

```
BPXO063I 16.22.32 DISPLAY OMVS 586
OMVS     000E ACTIVE              OMVS=(2W,DN)
MOUNT LATCH ACTIVITY: NONE

FILE SYSTEM LATCH ACTIVITY:
    USER   ASID     TCB              SHR/EXCL                      AGE
-------------------------------------------------------------------------
Latch 14 FILE SYSTEM: ZOS113.VAR.ZFS
HOLDER(S):
  TC0      0028  008E6D90             EXCL                    00.00.15
     TIME: 2011/02/08 16.22.17
     IS DOING: Running
WAITER(S):
  TC0      0029  008E6D90             SHR                     00.00.07
     TIME: 2011/02/08 16.22.24
```

D OMVS,W enhancement

```
FILE LATCH ACTIVITY:
    USER   ASID     TCB              SHR/EXCL                    AGE
-------------------------------------------------------------------
LATCH 66 LSET 01 TYPE REGFILE   DEVNO 2 INO 204
FILE: myfile
FILE SYSTEM: ZOS113.ETC.ZFS
HOLDER(S):
  TC0      0026  008E6D90            EXCL                   00.00.56
     TIME: 2011/02/08 16.21.36
     IS DOING: ZFS MKDirCall

WAITER(S):
  TC0      0027  008E6D90            SHR                    00.00.53
     TIME: 2011/02/08 16.21.38

OTHER WAITING THREADS:
    USER   ASID     TCB              PID                      AGE
-------------------------------------------------------------------
  CEA      0016  008E2D90        16777218                   00.06.51
     TIME: 2011/02/08 16.15.40
     IS DOING: Osi Wait
```

D OMVS,W enhancement

- DEVNO - indicates the device number (decimal) of the file system in which the file resides
- INO - indicates the inode number (decimal) of the file
- LSET - indicates the latch set identifier. File latches are created in the SYS.BPX.A000.FSLIT.FILESYS.LSN.XX latch set where XX corresponds to LSET.
- TYPE - indicates the type of file (DIR, CHARSPEC, REGFILE, FIFO)
- FILE - indicates the name of the file (if known). The name can be up to 16 characters long (truncated on the left).
- FILE SYSTEM - indicates the file system that owns the file.

Note: CINET sockets use the same latch set as files. This display will not show any sockets.

D OMVS,W enhancement

- **DISPLAY OMVS,WAITERS – filtering**
  - **LATCHES | L** - this will only display the latch activity tables. These include the mount, file system and file latch activity tables. The cross system messages and other waiters tables will be suppressed.
  - **MESSAGES | M** - this will only display the sent and received cross system messages table. The mount latch, file system latch, file latch and other waiters tables will be suppressed. This option is only valid in a shared file system environment.
  - **OTHER | O** - this will only display the other waiters table. The mount latch, file system latch, file latch and cross system messages tables will be suppressed.
  - **AGE | A** - this will only display waiters that have been waiting for more than five minutes. A table will be displayed if there are waiters that meet the criteria.
  - **SPECIAL | S** - this will allow special files to be displayed in the other waiters table that are otherwise filtered out. These are: character special files, pipes and sockets. This option will be allowed by itself, with the A option or when the O option has also been specified.

D OMVS,W enhancement

- **Filtering Examples**
  - D OMVS,W,L,O – display only the latch and other waiters tables
  - D OMVS,W,A – display any activity in any of the tables where we have been waiting for more than five minutes
  - D OMVS,W,L,A – display only activity in the latch tables that have been waiting for more than five minutes
  - D OMVS,W,S – display any waiters in all tables and allow other waiters table to also include waiters on character special files, sockets and pipes
  - D OMVS,W,O,S – display only waiters in the other waiters table and also include waiters on character special files, sockets and pipes

D OMVS,W enhancement

- **Filtering Example: D OMVS,W,O**

```
BPXO063I 16.22.32 DISPLAY OMVS 586
OMVS     000E ACTIVE              OMVS=(2W,DN)
OTHER WAITING THREADS:
    USER   ASID     TCB               PID                          AGE
-------------------------------------------------------------------------
  CEA      0016  008E2D90        16777218                       00.06.51
     TIME: 2011/02/08 16.15.40
     IS DOING: Osi Wait
```

D OMVS,W enhancement

- Data from D OMVS,W can be obtained programmatically using BPXEKDA and the BPXZODMV mapping
  - New filters: Set the following filters in ODMVINBYTE2 when ODMVWAITERS has been set in ODMVINBYTEM3

```
ODMVWLATCHES EQU X'80'     D OMVS,W,LATCHES
ODMVWMESSAGES EQU X'40'    D OMVS,W,MESSAGES
ODMVWOTHER EQU X'20'       D OMVS,W,OTHER
ODMVWAGE EQU   X'10'       D OMVS,W,AGE
ODMVWSPECIAL EQU X'08'     D OMVS,W,SPECIAL
```

  - Output data mapped by ODMVDWHEADER and ODMVDWELEMENT

Lost message detection

- **Problem Statement**
  - Using a shared file system configuration, need increased diagnostic capability to detect lost XCF messages between z/OS UNIX members.
- **Solution**
  - Utilize XCF ordered message delivery along with assigning sequence numbers to messages. Each message sequence number will be checked.
  - A lost or duplicate message results in a two system dump.
- **Benefits**
  - Timely service data.

Lost message detection

- **New PARMLIB keyword:**
  `LOSTMSG(ON|OFF)`
- **SETOMVS console command update:**
  `SETOMVS LOSTMSG=ON|OFF`
- LOSTMSG(ON) incurs a performance penalty in high z/OS UNIX traffic environments.
  - zFS SYSPLEX=FILESYS environment reduces z/OS UNIX traffic, however.
- LOSTMSG(ON) enabled for z/OS 1.13.

__console2() enhancements

- **Problem Statement**
  - The __console2() (BPX1CCS) syscall interface does not support supplied or returned console identifiers. This limits applications to communicating with only one console at a time.
- **Solution**
  - Add new fields to the __console2() MsgAttributes parameter (mapped by BPXYCCA) to support supplied and returned console identifiers.

__console2() enhancements

- **Benefits**
  - Applications will be able to interleave communications with multiple consoles by exploiting the new supplied and returned console identifiers.

__console2() enhancements

- Expanded BPXYCCA parameter mapping macro for __console2(). New section “Ver3”.

```
CCASTARTVER3 DS 0CL40    Start of version 3
CCAMODCARTPTRG DS 0CL8   +72 Pointer for amode 64 use
CCAMODCARTPTR DS A       +72 Pointer to 8 byte CART returned for       *
                         MODIFY/STOP command
         DS    CL4       +76 Not used in amode 31
CCAMODCONSOLEIDPTRG DS 0CL8 +80 Pointer for amode 64 use
CCAMODCONSOLEIDPTR DS A  +80 Pointer to 4 byte ConsoleID returned for  *
                         MODIFY/STOP cmd
         DS    CL4       +84 Not used in amode 31
CCAMSGCART DS  CL8       +88 Supplied - CART to be specified on WTO    *
                         when message is issued
CCAMSGCONSOLEID DS CL4   +96 Supplied - ConsoleID to be specified on   *
                         WTO when message is issued
         DS    CL12      +100 Reserved for future use
CCAENDVER3 DS  0C        End of Version 3
```

IPv4 IP_PKTINFO support

- **Problem Statement**
  - Currently under CINET environment with multiple TCPIP stack configuration, when a server system has multiple home addresses with multiple routes back to the client, the UDP reply packet might not flow on the same interface where the UDP request packet arrived.
- **Solution**
  - Provide a new z/OS IPv4 external interface to obtain request's inbound interface information and use it on the reply.
  - Similar to the existing IPv6 external interface.

IPv4 IP_PKTINFO support

- **Benefits**
  - Server can send the UDP reply packet to a client request out on the same inbound physical interface that the client's UDP request packet arrived.

IPv4 IP_PKTINFO support

- **Expected IPv4 IP_PKTINFO Usage Protocol:**
  - Application requests inbound interface info to be part of recvmsg()
    - Use setsockopt() with new IP_RECVPKTINFO option
  - TCP/IP stack includes the inbound interface info in a new IN_PKTINFO structure as an ancillary data item on the recvmsg()
  - Returned inbound interface info is used untouched on the subsequent sendmsg()

IPv4 IP_PKTINFO support

- **IPv4 IP_PKTINFO External Interface:**
  - Ancillary data type for the new IN_PKTINFO mapping structure:

```
IP_PKTINFO     EQU   101
```

  - setsockopt() call option to receive inbound interface information as ancillary data on recvmsg() call:

```
IP_RECVPKTINFO      EQU   102   (IPPROTO_IP Level
option)
```

  - Ancillary data mapping structure:

```
IN_PKTINFO     DS    0F
IPI_ADDR       DS    CL4    IPv4 Address
IPI_IFINDEX    DS    F      Interface index
```

IPv4 IP_PKTINFO support

- **Notes:**
  - The program must to call setsockopt() with IP_RECVPKTINFO option to have the TCPIP stack pass the client's return information in a IN_PKTINFO structure as ancillary data item on the recvmsg() call, and that IP_PKTINFO data is used, untouched, on the subsequent sendmsg() to have the reply flow out the same interface the request arrived.
  - z/OS NFS server exploits this support.

Symbolic links to shared root

- **Problem Statement**
  - As a z/OS S&U user, I want to be able to mount my version-root file system read-only, without having to make post-installation actions for some utilities (cron, mail, and uucp) after each new release.
- **Solution**
  - Eliminate the post-installation actions, by defining the symbolic links these utilities need to function in a read-only version-root file system at installation time.
  - Provide a new Migration Health Checker to determine if the user will be affected by a one-time migration action due to this change.

Symbolic links to shared root

- **Benefits**
  - These post-installation actions will be eliminated, if the user takes advantage of the provided symbolic links.

Symbolic links to shared root

- As of z/OS V1R13, ServerPac is delivered with the /usr/lib/cron, /usr/mail and /usr/spool directories as symbolic links to /var. ServerPac users will not need to take any action to create these symbolic links.
- For CBPDO Users, the required directories and symbolic link structure are created during execution of the BPXMKDIR REXX exec in SYS1.SAMPLIB.

Symbolic links to shared root

- The symbolic links are directed to /var directories as follows:

Directory	Linked to…
`/usr/lib/cron`	`../../var/cron`
`/usr/spool`	`../var/spool`
`/usr/mail`	`../var/mail`

Symbolic links to shared root

- For files used by uucp, these files are delivered as symbolic links that are directed to the /var directories (see below). The /var file system provided by ServerPac will not contain these files, they are provided when the utilities are configured and used.

File	Linked to…
/usr/lib/uucp/Systems	../../../var/uucp/System
/usr/lib/uucp/Devices	../../../var/uucp/Devices
/usr/lib/uucp/Dialers	../../../var/uucp/Dialers
/usr/lib/uucp/Dialcodes	../../../var/uucp/Dialcodes
/usr/lib/uucp/Permissions	../../../var/uucp/Permissions
/usr/lib/uucp/config	../../../var/uucp/config

Symbolic links to shared root

- **CBPDO considerations -** The following issues will cause a failure in BPXMKDIR, the Migration Health Check will verify both.
  - (1) Symbolic link targets created ahead of time must be created exactly as shown in the previous slides, otherwise BPXMKDIR and the Health Check will fail. For example:

Directory	Linked to…	Acceptable
`/usr/mail`	`../var/mail`	`Yes`
`/usr/mail`	`/var/mail`	`No`

Symbolic links to shared root

- **CBPDO considerations (continued):**
  - (2) Customer content must be moved before running BPXMKDIR.
    - **/usr/lib/cron** - Must be empty.
    - **/usr/mail** - Must be empty.
    - **/usr/lib/uucp** - Should contain the IBM files uucc, uucico, uuxqt and contains the IBM subdirectory.
    - **/usr/spool** - May contain the following empty IBM subdirectories: cron, locks, cron/atjobs, cron/crontabs, uucp, uucppublic, uucp/.Xqtdir, uucp/.Sequence, uucp/.Status

Symbolic links to shared root

- The verification used in the V1R13 BPXMKDIR and the V1R11 and V1R12 Migration Health Check is essentially the same.
- The Migration Health Check will be written in System REXX, so the System REXX service, as well as the Health Checker started task and Unix System Services, must be available on the system to run this check.

Symbolic links to shared root

- A user will be notified by the Health Check that a one-time migration action will need to be taken if either of the following is true:
  - You have performed the post-installation activities to make uucp, cron, or mail supported for a read-only z/OS version-root. You do not necessarily have to be running with the z/OS version-root as read-only, but only have the post-installation customization as described in z/OS UNIX System Services Planning.
  - You have used uucp, cron, or mail facilities and have not performed the post-installation customization as described in z/OS UNIX System Services Planning.

Symbolic links to shared root

- While this migration action should be performed before the first IPL of z/OS V1R13, the changes to use /var for this support can be done at any time.
  - Although previous documentation had shown the use of /etc in examples, we now recommend using /var for these utilities.
- If you use /etc or another directory today for your post-installation customization:
  - You may continue to use this directory, though it is not recommended.
  - Moving to the /var structure that is provided with z/OS V1R13 is recommended.

Symbolic links to shared root

- **Why use /var structure provided with z/OS V1R13?**
  - (1) You can minimize any subsequent post-installation customization, since the symbolic links to /var will be provided for you by IBM.
    - Continued use of non-/var directories may mean post-installation work for every new release, to remove the delivered structure and replace it with your own.
    - Use of non-/var directories negate the benefits of the provided symbolic links, because the post-installation actions will be changed, but not eliminated.

Symbolic links to shared root

- **Why use /var structure provided with z/OS V1R13?**
  - (2) Continued use of /etc (or another directory), requires you to manage and maintain the symbolic links required from /var to that directory, which is “double symlinking.” This “double symlinking” might be confusing for those that maintain the system.
    - In order to use a non-/var directory, you will have to create an additional symbolic link from the /var directory to your chosen target directory. For example, if you chose to use a directory other than /var/mail (such as /etc/mail), you will need to create an additional symbolic link from /var/mail to this directory. These “double symlinks” can be reused for future releases, if you continue to provide the intermediate links for each release.

Symbolic links to shared root

- If you have any user files under /usr/spool, /usr/lib/cron, or /usr/mail in the version-root file system (that is, they have not been symlinked under /var or another directory to support the read-only version-root), then those files must be moved to /var/spool, /var/cron, or /var/mail (or another directory besides the /var that you choose).
- For uucp: If you have any files, except those listed below, under /usr/lib/uucp in the version-root file system (that is, they have not been symlinked under /var or another directory to support the read-only version-root), then those files must be moved to /var/uucp (or another directory besides the /var that you choose)
  - **Files:** uucc, uucico, uuxqt
  - **Subdirectory:** IBM

Symbolic links to shared root

- For both ServerPac and CBPDO Users, the Health Check ZOSMIGV1R13_RO_SYMLINKS provided in APARS OA35605 & OA35636 should be used to check if a one-time migration action will need to be taken when migrating from z/OS V1R11 or V1R12.
  - Install the APARS OA35605 & OA35636 on V1R11 or V1R12.
  - Activate the ZOSMIGV1R13_RO_SYMLINKS Health Check.
  - Examine the results.

Symbolic links to shared root

- **Health Check Success Messages:**

```
BPXH910I The directory /usr/mail is not customized.

BPXH920I The directory /usr/lib/cron is customized to the ../../var/cron
directory.
```

- **Overall Success Message:**

```
BPXH913I All directories verified were found to be acceptable for the new
symlinks added in z/OS V1R13. A migration action is not required.
```

Symbolic links to shared root

- **Health Check Error Messages:**

```
BPXH912I The directory /usr/lib/cron has additional files, directories, or
symbolic links found as follows:
file1
file2
directory1

BPXH911I The directory /usr/mail has a symlink to /var/mail.
```

- **Overall Failure Message:**

```
BPXH915E One or more of the directories verified were found to contain
post-install customization that is expected to be affected by the new
symlinks added in z/OS V1R13, or there were problems accessing the
directory. A migration action is required.
```

Questions

- **Any questions, concerns or comments?**

Appendix

- z/OS V1R13.0 UNIX System Services Command Reference (SA22-7802-14)
- z/OS V1R13.0 UNIX System Services Planning (GA22-7800-19)
- z/OS V1R13.0 UNIX System Services Programming: Assembler Callable Services (SA22-7803-14)
- z/OS V1R13.0 UNIX System Services Messages and Codes (SA22-7807-12)
- z/OS V1R13.0 UNIX System Services File System Interface Reference (SA22-7808-13)
- z/OS V1R13.0 UNIX System Services User's Guide (SA22-7801-14)
- z/OS V1R13.0 Using REXX and z/OS UNIX System Services (SA22-7806-14)

Appendix

- z/OS V1R13.0 MVS Initialization and Tuning Reference (SA22-7592-22)
- z/OS V1R13.0 MVS System Commands (SA22-7627-26)
- z/OS V1R13.0 MVS Diagnosis: Reference (GA22-7588-16)
- z/OS V1R13.0 MVS Authorized Assembler Services Reference ALE-DYN (SA22-7609-12)
- z/OS V1R13.0 MVS Data Areas Volume 1 (GA32-0853-01)
- z/OS V1R13.0 MVS System Messages, Vol 3 (ASB-BPX) (SA22-7633-21)

Appendix

- z/OS V1R13.0 XL C/C++ Run-Time Library Reference (SA22-7821-13)
- z/OS V1R13.0 TSO/E Programming Services (SA22-7789-09)
- z/OS V1R13.0 IBM Health checker for z/OS: User's Guide (SA22-7994-12)
- z/OS V1R13.0 Security Server RACF Callable Services (SA22-7691-15)
- z/OS V1R13.0 Security Server RACF Auditor's Guide (SA22-7684-13)
- z/OS V1R13 Program Directory (GI10-0670)
- z/OS V1R13 Migration Guide (GA22-7499-15)

Appendix

- USS: http://www.ibm.com/systems/z/os/zos/features/unix/
- “Tools and Toys”: http://www.ibm.com/systems/z/os/zos/features/unix/tools/
- SUSv2 (UNIX98): http://www.unix.org/version2/
- SUSv3 (UNIX03): http://www.unix.org/version3/