z/OS PARMLIB Successful Practices - User Experience

Mary Anne Matyaz
Base Technologies Inc, A CA Technologies Company, for U.S. Customs and Border Protection

Thursday, August 11, 2011
Session 9800
D PARMLIB

IEE251I 10.07.58 PARMLIB DISPLAY 446
PARMLIB DATA SETS SPECIFIED AT IPL
ENTRY FLAGS VOLUME DATA SET

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>S</td>
<td>SYSCTT</td>
<td>USCT.PARMLIB</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>S</td>
<td>SYSX04</td>
<td>SYS2.PARMLIB</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>D</td>
<td>SYSR31</td>
<td>SYS1.PARMLIB</td>
<td></td>
</tr>
</tbody>
</table>

On system-specific cat pack

Shared across all LPARs

IBM Vanilla, shared across lpars
How do we specify multiple parmlibs?

SYS1.IPLPARM(LOADTT)
SYSCAT SYSCTT113CMCAT.VSYSCTT
SYSPARM 00
IEASYM (TT,00,VS,L)
PARMLIB USCT.PARMLIB
PARMLIB SYS2.PARMLIB
PARMLIB SYS1.PARMLIB

• LOADPARM 524ATTM
Symbol Usage

- We HEAVILY use symbols in our shop

- IEASYM (TT,00,VS,L)

- TT is the System (LPAR) specific
- 00 is shared by all LPARs, it sets up the TT
- VS is shared by all LPARs, unless we’re in release transition
IEASYM00 - Shared by all LPARs

/* Static Symbol Definitions for all systems */

SYSDEF SYMDEF(&TZONE.='&EDT') /* Set current time zone */
SYMDEF(&EDT.='W.04') /* GMT offset for EDT */
SYMDEF(&EST.='W.05') /* GMT offset for EST */
SYMDEF(&SYSCDN.='1') /* XCF and CFRM CDS num 1 */
SYMDEF(&GRSMOD.='STAR') /* GRS=STAR for NDC */

/* Set system ID related symbols. Examples assume SYSNAME='USCT' */
SYMDEF(&SYSCPX.='&SYSNAME(1:3)') /* USC */
SYMDEF(&SYSSFX.='&SYSNAME(4:4).&SYSNAME(4:4)') /* TT */
SYMDEF(&SYSSID.='&SYSNAME(4:4)') /* T */
IEASYMTT - USCT LPAR Specific

/* Static Symbol Definitions for system: USCT */
SYSDEF SYSNAME(USCT)
SYSPARM(T1,TT) /* Sysplex=CSPLEX T1 Sysid=USCT */
SYMDEF(&AALEVL.='V2110D2') /* Level of ABENDAID */
SYMDEF(&CA11LVL.='R11SP00C') /* Level of CA-11 */
SYMDEF(&CA7LEVL.='R113SP0D') /* Level of CA-7 */
SYMDEF(&CCSLEVL.='R12SP00K') /* Level of Common Services */
SYMDEF(&CICSLVL.='C320T01') /* Level of CICS TS */
SYMDEF(&CICSVER.='V320') /* Version of CICS */

There are 64 lines in this member!
IEASYMVS – z/OS Version/Release Symbols

/* Static Symbol Definitions for z/OS 1.12 */
SYSDEF SYMDEF(&OSFMID.='HBB7770') /* OS FMID */
    SYMDEF(&OSNAME.='ZOS') /* OS Name */
    SYMDEF(&OSVER.='1') /* OS Version */
    SYMDEF(&OSREL.='C') /* OS RELEASE */
    SYMDEF(&OSVR.='&OSVER.&OSREL') /* OS Version / Release */
    SYMDEF(&OSVR2.='&OSVER..&OSREL') /* OS Version / Release */

SO…Where do we use all these dang symbols anyway??
Everywhere we can! - In other parmlib members:

```
LNKLST ADD NAME(PROGLT) DSN(CCSSYS.&CCSLEVL..CAILOAD)
LNKLST ADD NAME(PROGLT) DSN(CCSSYS.&CCSLEVL..CAIDCM)
LNKLST ADD NAME(PROGLT) DSN(SYNCSYS.&SYNCLVL..SORTLIB)

PCOUPLE(SYSPLEX.CSPLPX.&SYSCDN..CDS.PRI,SPXP11)

MOUNT FILESYSTEM('CICS.&SYSNAME..&CICSLVL..ZFS')
    MOUNTPOINT('/$SYSNAME/cicsts')
    TYPE(ZFS) MODE(RDWR) SYSNAME(&SYSNAME) UNMOUNT

DAE=START, DSN(&SYSNAME..DAE)

CICSSYS.&SYSNAME..BETA.SDFHLPA(LNKLST),  CICS 42 LPA MODULES
TSSSSYS.&TSSLEVL..CAKOLPA,   TSS LPA MODULES

In JES2PARM: INCLUDE MEMBER=HASEXIT&OSREL
```
Everywhere we can! - In Procs:

//CAS9     PROC DCOM=DCOMB0,RIMPARM=RIMPARM&SYSSID.
//CAIRIM    EXEC PGM=CAIRIM,TIME=1440
//STEPLIB   DD DISP=SHR,DSN=DISP.PROD.&DISPLVL..CAILIB
//          DD DISP=SHR,DSN=CA11.&CA11LV..CAL7LOAD
//          DD DISP=SHR,DSN=DATAcom.PROD.&DCOM..CAILIB
//          DD DISP=SHR,DSN=ROSCOE.&ROSEVL..RO60LIB
//PARMLIB   DD DISP=SHR,DSN=CCS.PROD.PPOPTION(&RIMPARM)

//SYSSIN   DD DISP=SHR,DSN=CICS.&REL..SYSIN(&SYSNAME) SIT-OVERRIDS
//          DD DISP=SHR,DSN=CICS.&REL..SYSIN(&CICS) SIT-OVERRIDES
//STEPLIB   DD DISP=SHR,DSN=CICS.&REL..ALIAS.&SYSNAME..SDFHAUTH
//          DD DISP=SHR,DSN=CICS.&REL..ALIAS.&SYSNAME..SDFJAUTH

//SYSTCPD   DD DISP=SHR,DSN=CNMP.TCP32.TCPDATA(&SYSNAME)
Everywhere we can! - TSO Logon Procs

//         DD DISP=SHR, DSN=OPSS.PROD.FBCLIST     <== CA OPS/MVS
//         DD DISP=SHR, DSN=OPSS.&OPSLEVL..FBCLIST  <== CA OPS/MVS
//         DD DISP=SHR, DSN=FDR.PROD.USRCLIB        <== FDR CLIST
//         DD DISP=SHR, DSN=INT.&INTBLVL..CAICLIB    <== INTERTEST

In extended aliases:

DEFINE ALIAS (NAME(TMS.PROD.CTAPOPTN) -
SYMBOLICRELATE(TMS.&TMSLEVL..CTAPOPTN))

3.4 shows:

Command - Enter "/" to select action       Message       Volume
---------------------------------------------------------------------
I          TMS.PROD.CTAPOPTN                        *ALIAS

“I” Shows:
Data Set Information
Command ==> Data Set Name . . . :TMS.V120SP0B.CTAPOPTN
One change product upgrades

• So for example, if I put fixes on CA1, I create a new level of extended alias, and make one change, the system symbol:

**FROM:**
SYMDEF(&TMSLEVL.= 'V120SP0A') /* Level of TMS */

**TO:**
SYMDEF(&TMSLEVL.= 'V120SP0B') /* Level of TMS */

I IPL, and I’m ready to go. Or I could just use IEASYMUP SETPROG for LINKLIST and APF, and rerun TMSINIT.
So what's in USCT.PARMLIB?

- Not much!

- Some APPC members for LUDEL of MQ ACB’s

- The odd SCHEDxx or ERBRMFxx that someone is working on, but only on our test systems.

- We look forward to batch jobs being able to use system symbols. ☺
IEASYS00

- **CMD= (00, &SYSSFX),**
- **LOGREC= &SYSNAME ..LOGREC,**
- **LPA= (&SYSSFX, L),**
- **PROG= (00, A&SYSSID, L&SYSSID),**
- **SMF= &SYSSFX,**
z/OS Bi-Monthly maintenance

• We put two months of RSU’s and PUT’s on, six weeks back. (IE, on June 14, we put 1103 and 1104 PUTs on)
• We clone a new sysres, and change any necessary symbols, then IPL our little system. The changes run there for two weeks. We do have CICS/MQ/DB2/Datacom here.
• After two weeks, we IPL the other two, higher activity test lpars off the same sysres.
• After two more weeks, we begin rolling production IPLs, with the new SYSRES and symbol changes. This takes about four weeks to complete.
• We ‘stage’ the symbol changes the day before, and IPL with a new Load address.
z/OS Maintenance – Backout

- We’ve not had to backout in the three years I’ve been here.
- Another sysprog remembers backing out once, but can’t recall the reason
- If we did, we’d simply undo the parmlib changes and IPL with the previous load address.
### Implementation Plan: Page 1

#### Update SYS2.PARMLIB (IEASYMMM)

<table>
<thead>
<tr>
<th>FROM:</th>
<th>TO:</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;AALEVL.='V2110D1'</td>
<td>&amp;AALEVL.='V2110D2'</td>
</tr>
<tr>
<td>&amp;CICSLVL.='C320T03'</td>
<td>&amp;CICSLVL.='C320T01'</td>
</tr>
<tr>
<td>&amp;FDRPLVL.='V5475B'</td>
<td>&amp;FDRPLVL.='V5475C'</td>
</tr>
<tr>
<td>&amp;HIPLVL.='V8000D0'</td>
<td>&amp;HIPLVL.='V8000D1'</td>
</tr>
<tr>
<td>&amp;JCKLEVL.='R11SP01E'</td>
<td>&amp;JCKLEVL.='R11SP01F'</td>
</tr>
<tr>
<td>&amp;MIMLEVL.='R117SP2D'</td>
<td>&amp;MIMLEVL.='R117SP2E'</td>
</tr>
<tr>
<td>&amp;QWKLEVL.='R73A'</td>
<td>&amp;QWKLEVL.='R74A'</td>
</tr>
<tr>
<td>&amp;SYNCLVL.='ZOSR14A'</td>
<td>&amp;SYNCLVL.='ZOSR14B'</td>
</tr>
<tr>
<td>&amp;SYSVLVL.='R127SP0G'</td>
<td>&amp;SYSVLVL.='R130SP0A'</td>
</tr>
<tr>
<td>&amp;TMOCVLVL.='T310T01'</td>
<td>&amp;TMOCVLVL.='T310T03'</td>
</tr>
<tr>
<td>&amp;TPXLEVL.='TPX530B'</td>
<td>&amp;TPXLEVL.='TPX530C'</td>
</tr>
<tr>
<td>&amp;TSSDB2L.='V13SP00A'</td>
<td>&amp;TSSDB2L.='V13SP00B'</td>
</tr>
<tr>
<td>&amp;TSSLEVL.='R1SP00H'</td>
<td>&amp;TSSLEVL.='R15SP00K'</td>
</tr>
<tr>
<td>&amp;VPSVLVL.='V2R10C'</td>
<td>&amp;VPSVLVL.='V2R10D'</td>
</tr>
<tr>
<td>&amp;ZENLVL.='V210F'</td>
<td>&amp;ZENLVL.='V210G'</td>
</tr>
</tbody>
</table>

#### Update SYS2.PARMLIB (PROGLM)

<table>
<thead>
<tr>
<th>FROM:</th>
<th>TO:</th>
</tr>
</thead>
</table>

---

*SHARE in Orlando 2011*
Update SYS2.PARMLIB(PROGLM)  
z/OS System Staff - 703-921-xxxx  
Implementation Date / 02:00:00

Remove the following entry after SYS2.LINKLIB:

  LNKLIST ADD NAME(PROGLD) DSN(DBSYS.SUBSYS.DBLOAD) VOL(DBSY04)

Update SYS2.PARMLIB(BPXPRMMM)  
z/OS System Staff - 703-921-xxxx  
Implementation Date / 02:00:00

To be determined – will depend on which HFSs are replaced by ZFSs.

After making the changes, perform a syntax check of this member by issuing the following SDSF log

SETOMVS SYNTAXCHECK=(MM)
Catalogs

- Each LPAR has its own master catalog. Though we are going to look at sharing.
- The master catalog uses indirect addressing with sysres, and stays the same across maintenance and even z/OS releases. (We do a manual check to see if we need to add any SYS1 datasets or remove any)
- All UCATs are connected to all MCATs and as such, all dasd and datasets are shared (across two sysplexes). This DOES cause an issue with PDSE’s.
Incidentals

• We have great users. They are diligent testers! That helps! All the way from CICS/MQ/DB2/Datacom to end users.

• Our implementation plan runs 40 pages. Most of it is change control junk, but we do have backout procedures for each product.

• We didn’t build this in a day, it evolved over years.
Questions?