#158
MVS Audit and parmlib Workshop

Mark S. Hahn
I/T Consultant
IBM

June 10-11, 1999

Abstract

OS/390 (the operating system formerly known as MVS) is a dynamic, growing operating system. But more than a group of independent pieces, it is a collection of interacting components. In order to effectively audit MVS, a basic understanding of how these building blocks work together is needed. This workshop will consist of two segments: MVS concepts and components and building a self-documenting MVS system.

More than a collection of checklist items, this will be an opportunity to understand how the puzzle fits together. Methodologies of auditing, procedures and real-life examples will be provided.

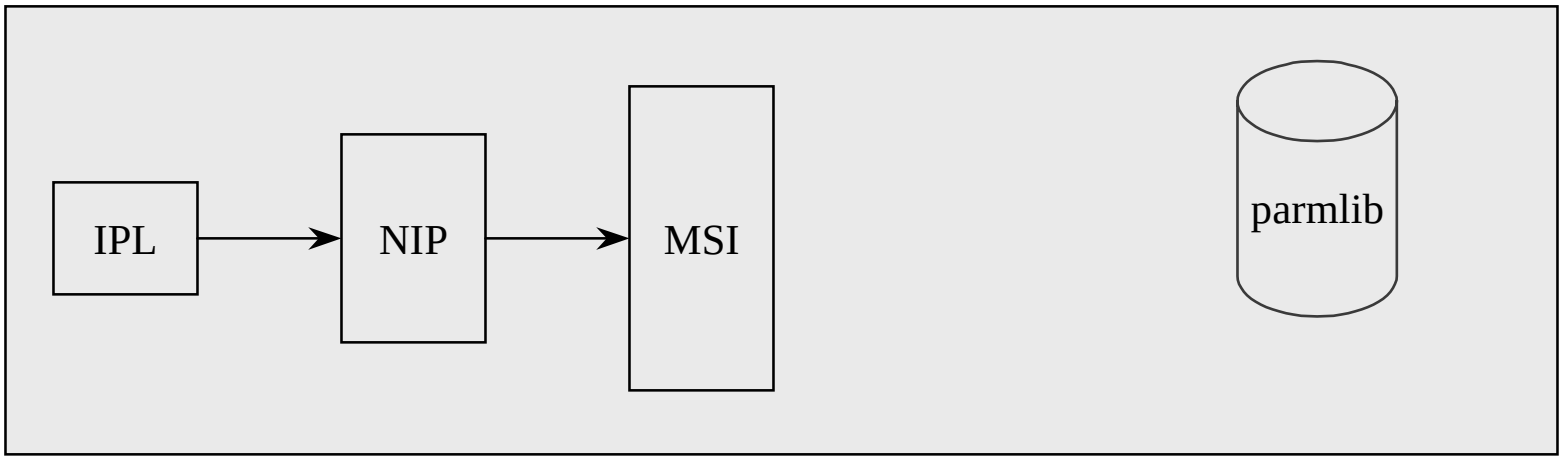
WARNING: This session will contain some basics, some advanced topics and a lot of MVS concepts.

Note: *Since OS/390 is the next evolutionary step of MVS, any reference to "MVS" applies to OS/390, except where specifically noted. In like fashion, with OS/390 R2's allowing multiple parmlib data sets, references to SYS1.PARMLIB include the parmlib data sets, except where so stated.*

OS/390 releases now number 2.7, with the next 2.8 due later this year.

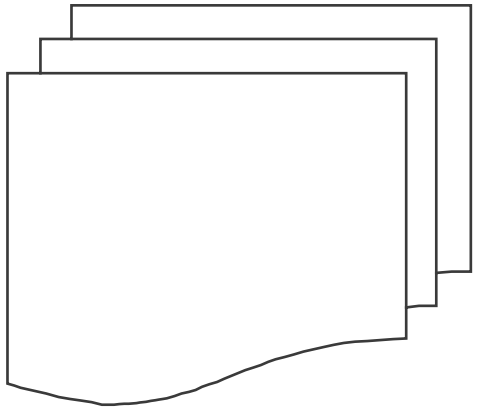
Workshop Plan

- Introduction to MVS
- Walk through an IPL
- Investigate *parmlib*
- Review other MVS components
- Building a Self-Documenting System
- Summary



Workshop Objectives

- Understand MVS components
- Develop *self-sufficient* tools for effortless review
- Blueprint for a self-documenting system
- Know how to address inconsistencies
- Produce desired hard copy of your findings



Basic Concepts

Basic concepts

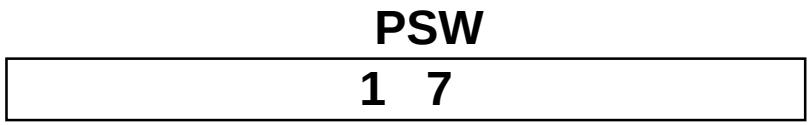
- Areas of storage (LPA, CSA, nucleus)
- Storage keys
- Address space

MVS releases

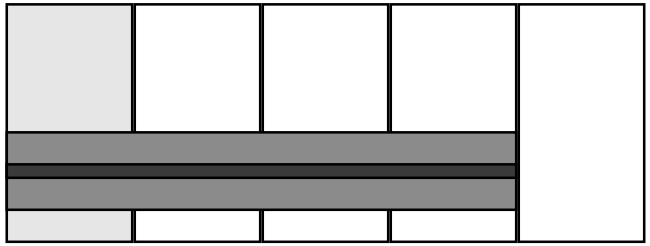
- MVS/370 - MVS SP1
- MVS/XA - Extended Architecture - MVS/XA
- MVS/ESA - Enterprise System Architecture - MVS/ESA
- OS/390 - Operating System - R1.1-R2.7 (anticipated R2.8)

Basic Concepts

- **370/390 architecture**
  - two states: supervisor & problem
  - 16 keys: 0-7 system; 8-F user



- **MVS architecture**
  - address space
  - system / common area
    - Nucleus
    - Link Pack Area
    - Common System Area

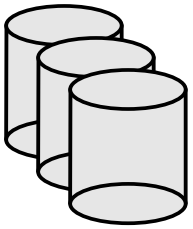


- **Address Spaces**
  - Private area (only accessible by user)
  - Shared areas: LPA, CSA and nucleus

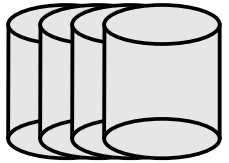
- **Data Space**
  - ALL private, no common

IPLing the System

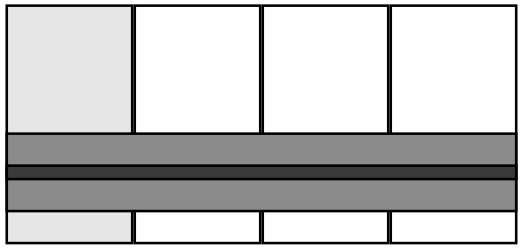
- What is IPL?
- Why IPL?
- Who controls IPL?
- What are the steps of IPL?
  - Initial Program Load (IPL)
  - Nucleus Initialization Program (NIP)
  - Master Scheduler Initialization (MSI)
- SYSLOG details member usage in R1.2

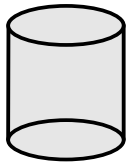


parmlib



LPALIB





SYS1.NUCLEUS

IPL

NIP

MSI

LOAD the System

- Push the “big blue button” or ‘O7X’

LOAD

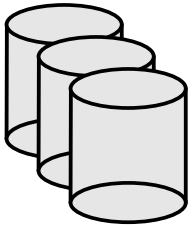
IPL

NIP

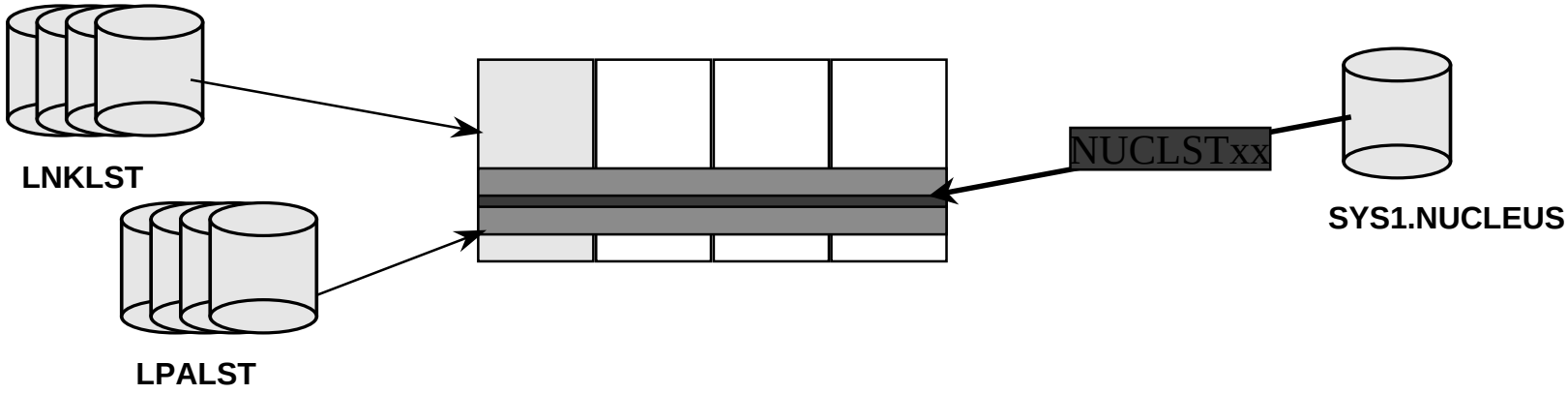
MSI

Slide Title????

- Resident routines shared by all users
- Heart and soul of MVS
- Basic definition is in SYS1.NUCLEUS
- Modifications in parmlib: LOADxx and NUCLSTxx
  - LOADxx: which IEANUC0x to load
  - LOADxx: define parmlib(s) <==
  - NUCLSTxx: module replacement - TEST system
- Build LINKLIST and LPALIST data set lists (using PROGxx)
- Type 1 and Type 2 SVCs (resident SVCs)



parmlib



IBM®

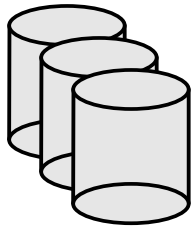
IPL

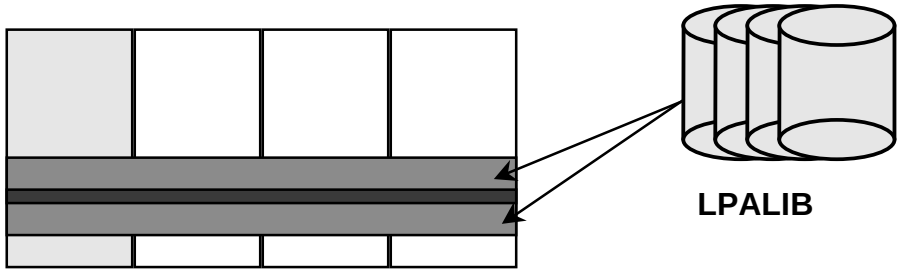
NIP

MSI

Pageable Link Pack Area

- PLPA: shared routines, not nucleus
  - Exits (including I/O appendages)
  - Type 3, 4 and 6 SVCs
  - Shared product modules
- Defined by parmlib members: LPALSTxx, IEASVCxx
  - list modified by PROGxx
- Routines are NOT modifiable





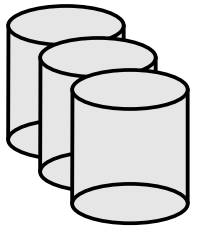
IPL

NIP

MSI

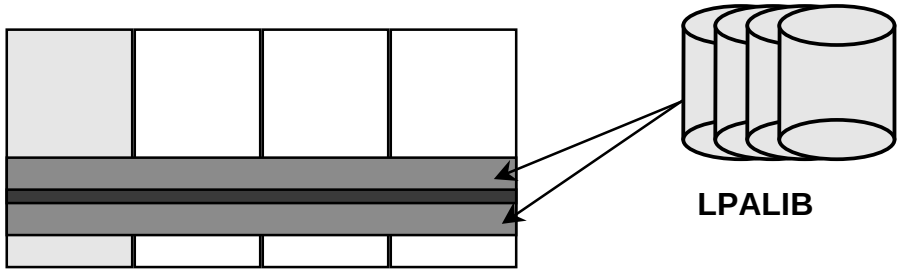
Fixed / Modified LPA

- Both can exist in system
- Fixed Link Pack Area - searched first for LPA modules
  - Done for tuning and performance concerns
  - Defined by IEAFIXxx and IEAPAKxx members of parmlib



parmlib

- Modified Link Pack Area - checked next for LPA modules
  - Entries include both data set name and member name(s)
  - Used for testing system exits (easy back out)
  - Defined by IEALPAxx member of parmlib



LPALIB

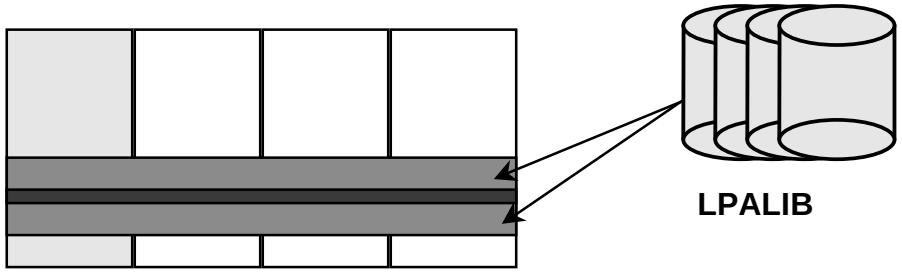
IPL

NIP

MSI

Common Storage Area [(E)CSA]

- Shared among all users (system and problem alike)
- Modifiable - therefore location for front-ended routines
  - One of your red flags: system routines residing in CSA are likely front-ended. Is the front-end “legal”?
  - Sometimes subsystems store data here -- but it can (likely) be read and / or modified by any user.



IPL

NIP

MSI

System Services

- Console definitions read from (CONSOLxx)
  - Operator signon could be required at this point
- Program Properties Table (PPT) built
  - merge of IBM table & installation entries (SCHEDxx)
- 3490 tape device compression (DEVSUPxx)
- Product licensing (IFAPRDxx and IGDDFPKG)
- DASD volume attributes set via VATLSTxx member

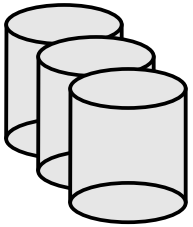
IPL

NIP

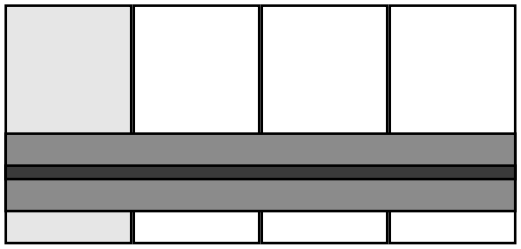
MSI

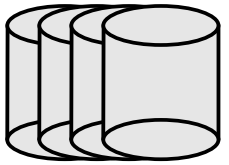
System Address Spaces Subsystems

- MSI reads MSTJCLxx in parmlib or from LINKLIB.
- IEACMD00 and COMMNDxx are read and passed to console services
  - Some services start even before JESx is started
  - SUB=MSTR enables cross-JES starts; restricted!
- Subsystem initialization begins: perform services for multiple users
  - JES -- proclibs, parm / INIT deck
  - DB2, RACF, SMS, etc..
  - **Usually** defined in IEFSSNxx



parmlib





LINKLIB

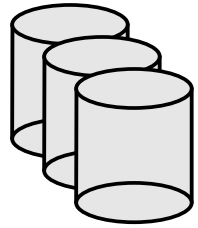
IPL

NIP

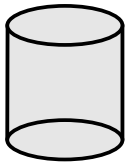
MSI

System Address Spaces
System Tasks

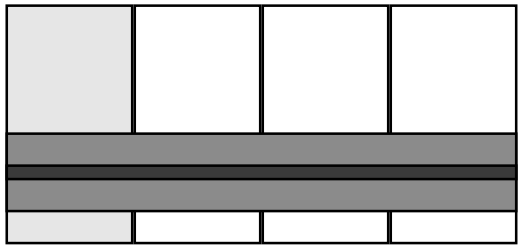
- VTAM (Virtual Terminal Access Method)
- TSO (Time Sharing Option)
- Advanced Program to Program Communication (APPC) uses APPCPMxx and ASCHxx
- Link Library Lookaside <CSVLLAxx>
- Hiperbatch (Data Lookaside Facility) <COFDLFxx>
- VLF (Virtual Lookaside Facility) <COFVLFxx>

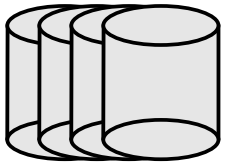


parmlib



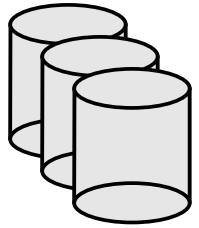
➢**SYS1.VTAMLIB**
SYS1.VTAMLST





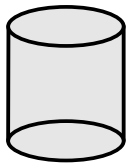
LLA Libraries

Review the System

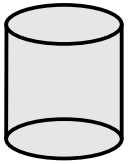


PARMLIB

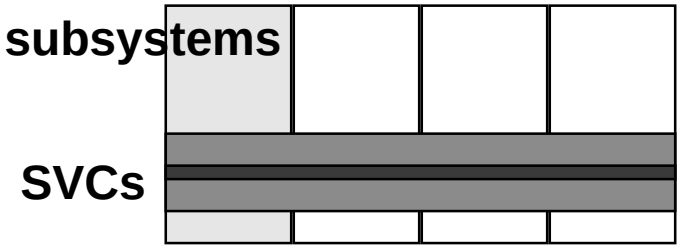
- Parmlib tops the checklist: APF libs, SVCs, LPA libs, exits, other modules
- APF libraries provide subsystems
- LPA holds SVCs, exits
- APF authorized programs rewrite the rules & insert themselves
- OS/390 produces own audit trail showing progress of system build.



APF Libraries



LPA Libraries



Auditing System Services and Facilities

Audit Tools

- DSMON - IBM supplied
- SYSLOG / LOGR - IBM supplied

- SMF
  - IRRADU00 - SMF unload and RACFICE
  - RYO reporting

- 3rd party products

System Services

- Focus on services
  - Review *parmlib* and other controls on services

- Services are generally combination of multiple *parmlib* and other controls

- *parmlib* members can affect multiple services

- What services will be addressed?
  - Authorized Program Facility (APF)
  - Sensitive and Critical Data sets
  - System Exits
  - Supervisor Calls (SVCs)
  - Subsystems
  - Other *parmlib* definitions

IBM's DSMON

Must have AUDITOR or PADS access to DSMON steplib

```
//xx JOB
//step     EXEC  PGM=ICHDSM00
//SYSPRINT DD  SYSOUT=x
//SYSUT2   DD  SYSOUT=x
```

DSMON reports

Reports:

- System identification
- PPT analysis
- Group Tree
- Authorized Callers Table
- Class Descriptor Table
- RACF Exits report
- Global Access report

DSMON - System ID

CPU-ID	117036
CPU MODEL	3090
OPERATING SYSTEM/LEVEL HBB5510 DRIVER19	MVS/ 3.8 SP5.1.0 HBB5510
SYSTEM RESIDENCE VOLUME	SYS021
SMF-ID	ISM3
RACF VERSION 2 RELEASE 1.0 IS ACTIVE	

- CPU Serial and model
- Operating system levels
- Sysres volser, SMF identifier in all records, RACF vesion

Define APF services

- What is APF authorization?
  - To be considered part of MVS
  - Permission to invoke **ANY** system service
  - Permission to use restricted SVCs (IBM and IEASVCxx)
  - Flag placed in user's address space for life of program (step)
  - The ability become system state or system key at will
  - The ability to change anyone's system storage, including security control blocks

- Two step control:
  - Program must execute from APF library (IEAAPFxx or PROGxx)
  - Program must be link edited with AC(1)

- Program Properties Table (PPT) entries must be executed from an APF library, but need not be AC(1)

APF Libraries

- **IEAAPFxx** - Static APF Library list
  - Cannot exceed 253 entries
  - Static table (by MVS definition)
  - Data set need not exist
  - Volume must be volser, blank (for SMS) or '******' for active sysres
  - non-keyword syntax (positional)

- **PROGxx** - Dynamic APF [and exit] list
  - No defined size restriction
  - Dynamic table (via SET PROG command)
  - Data set need not exist
  - Volume must be volser, SMS, *MCAT* or '******' for active sysres
  - Keyword syntax
  - controlled via FACILITY class profiles: CSVAPF.xxx

Program Properties Table

- Program Properties Table
  - Part IBM load module (IEFSDPPT)
  - Part installation defined text member of *parmlib*(SCHEDxx)

- Properties
  - Most related to tuning and performance
  - Some security and audit related
    - **NOPASS** - bypass password and RACF checking
    - **KEY(n)** - assign protect key (0-7 are system keys)

- Report using DSMON or other utility

- Dynamic control: SET SCH=(xx,L) command replaces table

- SMF90 record written to journal change of table

DSMON - PPT Report

P R O G R A M P R O P E R T I E S T A B L E R E P O R T

PROGRAM NAME	BYPASS PASSWORD PROTECTION	SYSTEM KEY
IEDQTCAM	NO	YES
ISTINM01	YES	YES
IKTCAS00	NO	YES
AHLGTF	NO	YES
HHLGTF	NO	YES
IHLGTF	NO	YES
IEFIIC	NO	YES
IEEMB860	YES	YES
IEEVMNT2	NO	YES
IASXWR00	NO	YES
CSVVFCRE	NO	YES
HASJES20	YES	YES
DFSMVRC0	NO	YES
IATINTK	YES	YES

APF Further Defined

- How else can programs become APF-Authorized?
  - System utility programs
  - Unscrupulous SVCs

- How to protect against unauthorized programs?
  - Universal access <= READ
    [consider UACC(NONE) and ID(*) ACCESS(<=READ)]
    - who must use?
  - More with system data sets

- Any product requiring APF authorization must be reviewed - WHY?
  What does it do? What are internal / external controls?

APF members of parmlib

```
SYS1.LOADLIB    WYWPK1,  /* MSH SYSTEM REQUIRED  */
SMS.LOADLIB           ,  /* DFB SMS-MGD LOADLIB */
SYSRES.LOADLIB  ******,  /* DCC MUST BE ON SYSRES */
DSN.NOT.THERE   WORK01
```

IEAAPFxx

```
FORMAT=[STATIC / DYNAMIC]
APF ADD DSNAME(SYS1.LOADLIB) VOL(SYSPK1) /*
MSH etc */
APF ADD DSNAME(PDQ.LOADLIB) SMS  /* DFB PDQ
LOAD  */
APF ADD DSNAME(SYSRES.LOADLIB) VOL(******)
```

PROGxx

OS:
D PROG,APF

```
AUTHCMD  NAMES(     /* AUTHORIZED COMMANDS
          +
          AD        /* RACF ADD DATASET
          +
          XMIT)     /* TSO/E 5665-285 */
AUTHPGM  NAMES(     /* AUTH0RIZED PROGRAMS
          +
          IEBCOPY   /* STANDARD IBM UTILITY
*/ +
          ICHUT100) /* RACF UTILITY     */
```

IKJTSOxx

TSO:
PARMLIB LIST

```
PPT PGMNAME(IEEMB860)         /* MASTER SCHED
*/
          NOCANCEL  KEY(0)  NOPASS  SYST
PPT PGMNAME(IEFIIC) /* INITIATOR */
          NOCANCEL KEY(0) PRIV SYST [PASS]
```

SCHEDxx

T SCH=(xx,L)
DSMON report

APF Summary

- Defines programs to be considered part of MVS, **regardless of who executes them**

- **Require use of comments** to identify / document entries. Include who, when, why the change was made.

- **IEASYSxx** - LNKAUTH= operand
  - APFTAB is most restrictive
  - LNKLST is more open - allows non-STEPLIB APF authorization

- **IKJTSOxx** - TSO/E controls
  - 00 is default, closely examine ANY other suffixes
  - alphabetize entries for easier finding
  - AUTHCMD, AUTHPGM and AUTHTSF should have all entries validated
  - Review CLASS(TSOAUTH) RESOURCE(PARMLIB) for who can issue PARMLIB LIST or PARMLIB UPDATE(xx) commands. Consider AUDIT(SUCCESS(READ)) on the command profile to know whenever the it is used

Checklist: APF

- APF Authorization
  - ***IEAAPFxx*** and **PROGxx** members of *parmlib*
  - **IKJTSOxx** members of *parmlib*
    - Check for PARMLIB command CLASS(TSOAUTH) to see who can change active IKJTSOxx member
  - **SCHEDxx** members of parmlib
    - closely check NOPASS and KEY(0-7) entries
  - LNKAUTH=[LNKLST / APFTAB] in **IEASYSxx**
  - Review APF data sets [DSMON list to start]
    - correct NF, NM entries
    - review UACC, protecting profiles, GAC, PERMIT lists

HIGH: System Data Sets

- Review System Data Sets Definition manual: GC28-1432
- If RACF: use DSMON SYSDSN report
- Usually HLQ of **SYS1.**
  - if it's SYS1.anything, LOOK AT IT!
  - Some must be universal read
  - Others must NOT be universal read
- May be defined in *parmlib* members
- NONE should be universal update
  - Except SYS1.BRODCAST (required)

HIGH: System Data Sets

- System required
  - logical *parmlib*
  - SYS1.NUCLEUS
  - SYS1.SVCLIB
  - SYS1.MIGLIB <PROGxx override>
  - SYS1.CSSLIB (SP 5.1+)
  - logical LINKLIB (LNKLSTxx or PROGxx)
  - logical LPALIB (LPALSTxx or PROGxx)
  - page data sets
  - catalog data sets
  - dump data sets
- Security data set(s)
- Change control, audit updates, signoff & review

DSMON - Sensitive Data Sets

S E L E C T E D D A T A S E T S R E P O R T

DATA SET NAME	VOLUME SERIAL	SELECTION CRITERION	RACF INDICATED	RACF PROTECTED	UACC
PAGE08.CATALOG	PAGE08	MASTER CATALOG	NO	YES	UPDATE
RACFBLD.DR15XA.LINKLIB	LIBPK6	APF	N.M.	YES	READ
RACFBLD.LINKLIB	RESPKC	LNKLST - APF	NO	YES	READ
SYS1.CMDLIB	RESPK1	LNKLST - APF	NO	YES	READ
		SYSTEM			
SYS1.IMAGELIB	RESPK1	SYSTEM	NO	YES	READ
SYS1.LINKLIB	RESPK1	APF	NO	YES	READ
		LNKLST - APF			
		SYSTEM			
SYS1.LPALIB	RESPK1	SYSTEM	NO	YES	READ
SYS1.MIGLIB	RESPK1	LNKLST- APF	NO	YES	READ
SYS1.NUCLEUS	RESPK1	SYSTEM	NO	YES	READ
SYS1.PARMLIB	RESPKC	SYSTEM	NO	YES	READ
SYS1.PROCLIB	RESPKC	SYSTEM	NO	YES	READ
SYS1.RACFB1	RDBBAR	RACF PRIMARY	NO	YES	UPDATE
SYS1.SVCLIB	RESPK1	APF	NO	YES	READ
		SYSTEM			
SYS1.UADS	RESPKC	SYSTEM	NO	YES	READ
SYS1.VTAMLIB	RESPK1	APF	NO	YES	READ
		LNKLST - APF			
	LIBPK6	APF	N.M.	YES	READ
	LIBPK5	APF	N.M.	YES	READ
SYS1.VTAMOBJ	RESPK1	APF	N.F.	YES	READ
	LIBPK6	APF	N.M.	YES	READ
SYS2.LINKLIB	RESPK1	LNKLST - APF	NO	YES	READ
SYS2.MIGLIB	RESPK1	LNKLST - APF	NO	YES	READ
SIVLE.AUTH.LOAD	RESPKC	APF	NO	NO	
TKCHIN.LOAD	D94RF2	APF	N.F.		

System Data Set Summary

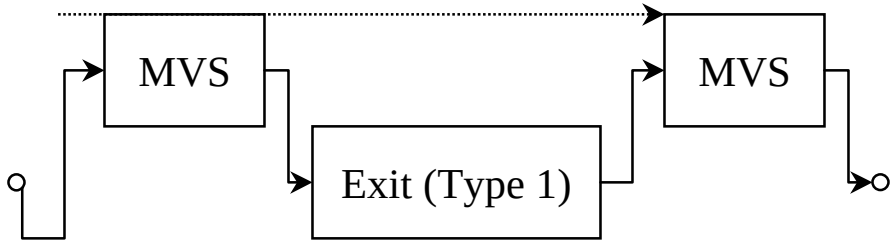
- Carefully review profiles
  - Be wary of GAC with **SYS1.**/READ**
    and depending upon follow-on SYS1.xx/NONE - new data sets added "without warning": SYS1.MIGLIB and SYS1.CSSLIB, for example.
    - Remember GAC cannot prevent access, only allow without further checking.
  - Review UACC, protecting profiles, GAC, PERMIT lists
  - Even a SYS1.** profile may not be appropriate with UACC above NONE. Protect these by default.
  - Use DSMON or other utility to seek out most of the system data sets
  - Review *parmlib* members known to house data set parms:
    - APPCPxx, COUPLExx, CSVLLAxx, IEAAPFxx, IEASYSxx, LOADxx, LNKLSTxx, LPALSTxx, MMSLSTxx, PROGxx, SMFPRMxx
    - be sure to watch for symbolic names (&SYSNAME as of SP 5.1)
    - Much more widely used
- Appendix C of RACF: Security Administrator's Guide (SC23-3726)

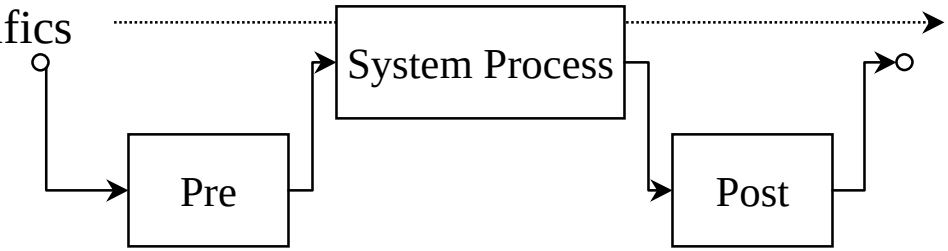
Dynamic Data set lists

- R1.2 - Dynamic parmlib
- R1.3 - Dynamic Linklist
  Reorder System Libraries
- R2.4 - Dynamic LPALIST
- Re-sequencing SYS1.LINKLIB, SYS1.LPALIB and SYS1.MIGLIB

HIGH: System Exits

- Function: provide customized (exception) processing to standard system service.
- Two types of exits:
  1. Self contained / decision making: IEFACTRT, IEFUJI
     may need chaining, based on products
  2. Modify input / decision: pre- and post-processing exit: ICHRIX01/02
- Future system trend:
  - exits are on the rise
  - better documented than covert
- Review manual:
- Benefits:
  - automated enforcement of standards
  - tailor installation / system specifics
- Risks
  - back door
  - compromise controls





- Generally reside in logical LPALIB - may be dynamically installed into (E)CSA

HIGH: System Exits

- "Standard" exits
  - IGGPRE00 - DASD pre-allocation
- *parmlib* specified
  - **CONSOLxx** - universal WTO exit (IEAVMXIT) (in)active
  - **EXITxx** - Allocation Recovery exits (SP 4.1- 5.1)
  - **MPFLSTxx** - Message Processing Facility
    - message processing
    - command intercept (MVS 4.2)
  - **PROGxx** - dynamic exits (SP 5.1)
    - SAF calls for FACILITY resource calls: **CSVDYNEX.****
  - **SMFPRMxx** - SMF exits
  - ... and there's more

HIGH: System Exits

- External Security system
  - RACF / Top Secret - active if present in LPA / name dependent
  - ACF2 - specified in C-GSO-EXITS record
- Product exits
  - review manuals: security / exits / customization
  - JES2 / JES3: many installation exits
    - some can compromise security processing
  - VTAM: more system exits

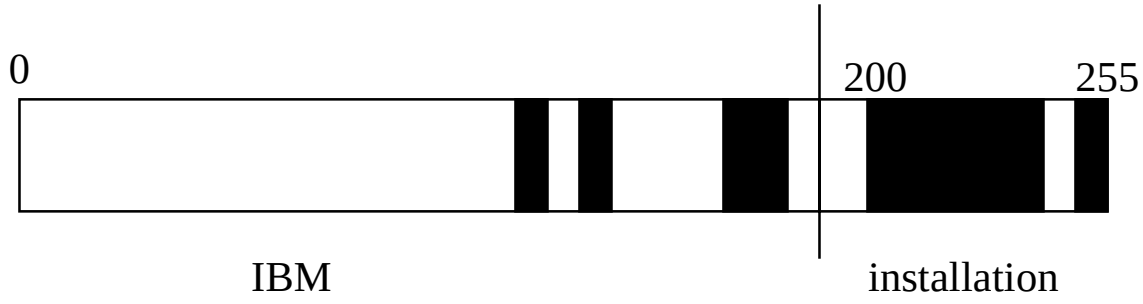
HIGH: SVCs

- Supervisor Calls: request operating system services on user level
- IBM sanctioned method to gain supervisor state
- IBM reserves 0-199
  - some are ESRs
  - Not all are in use
- 200-255 for installation use
  - possible to hijack unused SVC, based upon MVS release
  - ESR can be hijacked, too
- *parmlib*(IEASVCxx) documents some
- Some APF-authorized products install SVCs themselves!
- System utility to map SVCs in use

HIGH: SVCs

- Installation-written SVCs must be reviewed for integrity
- Vendor SVCs should be reviewed -- or a certificate of integrity issued
- Any product requiring an SVC number as installation parm NEEDS review

```
SVCPARM 215,REPLACE,TYPE(6)    /* CICS 1.7 HPSVC */
SVCPARM 216,REPLACE,TYPE(3),EPNAME(DFHCSVC)  /* CICS 3.1 SVC */
SVCPARM 226,REPLACE,TYPE(4),APF(YES)  /* ONLY APF CALLERS */
```



HIGH: Subsystems

- Subsystem: perform functions on behalf of many users.
  - DB2, JESx, IMS, automated operations, etc..
- Integral part of operating system
  - notified of select system events
  - can receive EVERY operator command issued
- Most are defined in *parmlib*(IEFSSNxx)
  - Is someone assigned responsibility?
  - Is IEFSSNxx commented?
- Some APF-authorized products install themselves
- Review for exits, security interfaces, etc..
  - JESx
  - DB2
- Use storage scan facility to review

IEFSSNxx
Subsystem Name Table

- Comments follow specification -- encourage their use
- Specifies multi-user service system components
- SMS should be first (if used)
  - PROMPT=DISPLAY echoes parms at startup
- JESx should be first unless SMS is used
- Look for RACF subsystem
- Not all subsystems are defined here, some dynamically install themselves (APF authorized), some may announce their presence, others not.

IEFSSNxx
Subsystem Name Table

```
SMS,IGDSSIIN,'ID=00,PROMPT=DISPLAY'
JES2,,,PRIMARY,NOSTART
CICS
D22E,DSN3INI,'DSN3EP,:'    DB2 TEST SYSTEM
============================
/* SUBSYSTEM1 BY MSH ON 6/13/96 */
SUBSYS SUBNAME(SSN1) INITRTN(SS1INIT) INITPARM(SSI,'+')
```

```
$HASP100 ADDxxxx  ON STCINRDR
$HASP ADDxxxx STARTED
xxxnnnI svc ### AND SSCT (xxxx) HAVE BEEN INSTALLED
$HASP395 ADDxxxx ENDED
```

Summary

•Primary native audit tool in Security Server shop is DSMON

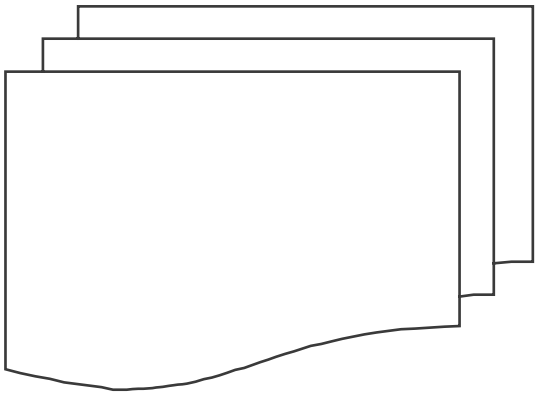
•Provides several reports: system id, formats Program Properties Table, reviews sensitive data set protection

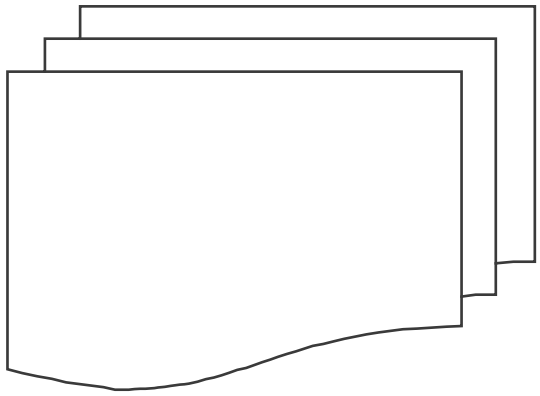
•Not all system data sets are named SYS1.**

•Parmlib members define (installation or system-image specific) system data sets

•Other system facilities require RYO reporting and/or console log review.

Build a *Self-Documenting* MVS System





Introduction

Why?

Enhanced audit & security

Historical Perspective

Four levels of documentation (author defined)

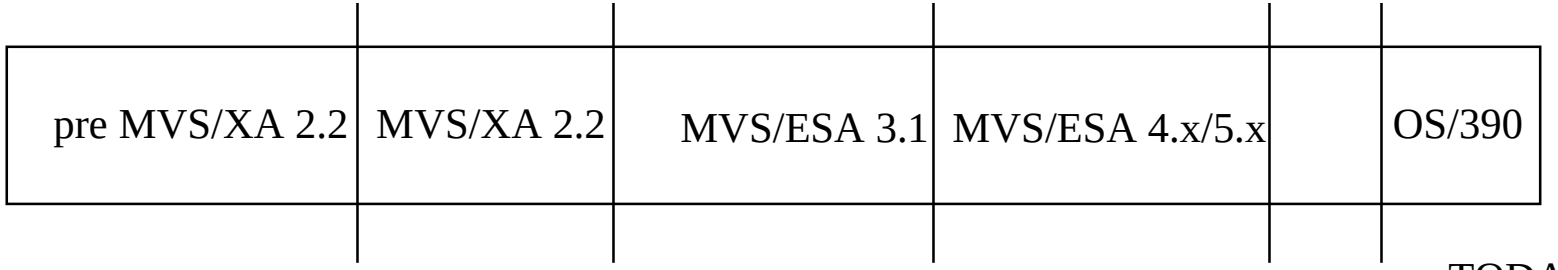
- MVS Automatic
- MVS Supported
- Operator Requested
- Ambitious Self-Documentation

“Static Audit” cannot fully detail environment

parmlib is just the BEGINNING

- Over 25 commands to modify operating environment
- Multiple *parmlib* data sets possible
- Some parms are sysplex in scope: CONSOLExx, ADYSETxx

Historical Perspective



Pre-MVS/XA 2.x

- Mostly static definitions

MVS/XA 2.x

- Many SYSGEN parms became English text in *PARMLIB*
- Still static
- System global (SET)

MVS/ESA 3.1

- System address space (MODIFY)

MVS/ESA 4.x

- Additive parameters (not only replacement)
- More SET commands (and SMF90 records)
- *IPLPARM* introduced

MVS/ESA 5.1

- Dynamic exits and system-image specific data set names

More History

- OS/390
  - Release 1.1 - no major changes
  - Release 1.2 - MAJOR changes
    - Enable multiple parmlib concatenations
    - Redfine initial LNKLSTxx and LPALSTxx libraries
  - Release 2.3
    - Define Linklist within PROGxx
  - Release 2.4
    - Dynamic LPA
  - Release 2.5
    - not much in security / security server arena
    - new parmlib member: IVTPRM00 (Communications Storage Manager (CSM)

SYSLOG or SMF

Every MVS system has SYSLOG and SMF

SYSLOG

- Free form messages, VB format
- Provides feel for entire system or sysplex
- Readily readable if access to SYSLOG (SDSF, IOF, FLASHER, etc..)
- Easy extensions: MPF, automated operations

SMF

- “Fixed” format VB records
- Discrete event documentation
- Requires report generation program
- SMF exit (IEFU8x); non-trivial
- HIGH risk for crash

Level I - MVS Automatic

Based on MVS release level, messages vary

Already wealth of status messages -- are you using?

Why take action?

- Pre-MVS SP4: *PARMLIB* supplies all
- IPL is two-phase in MVS SP4
  - 1. *IPLPARM* supplies nucleus construction
  - 2. *PARMLIB* supplies the rest (cataloged)

PARMLIB & IPLPARM

IPLPARM as of MVS/SP 4.2

- SYSn.IPLPARM on OPRCTL “Load Parm Address”
- SYS1.PARMLIB on OPRCTL “Load Parm Address”
- SYS1.PARMLIB on OPRCTL “Load Unit Address” (sysres)
- supplies LOADxx and NUCLSTxx

PARMLIB

- SYS1.PARMLIB on OPRCTL “Load Unit Address” (sysres)
- SYS1.PARMLIB based upon catalog address (SP 4.2)
- supplies all remaining parameters

multiple ***parmlibs*** (as of OS/390 R2)

- defined in LOADxx
- dynamic changes through SETLOAD command

Importance of IPLPARM

Recommended by IBM

- "It is not recommended that you use SYS1.PARMLIB to contain the LOADxx IPL parameters, but rather to use a SYSn.IPLPARM data set"

LOADxx specifies

- IEASYSxx
- Master catalog location => *parmlib* & system data sets
- Nucleus construction member (4.2)

NUCLSTxx (4.2)

- Specifies module add / delete / replacement in the nucleus AT IPL
- Simplifies testing nucleus resident module installs

SYS1.PARMLIB is first sysplex data set

Level I Log Messages

```
IEA371I SYS1.IPLPARM ON DEVICE 0143 SELECTED FOR IPL PARAMETERS
IEA246I LOAD ID 00 SELECTED
IEA519I IODF DSN = SYS1.IODF00
IEA520I CONFIGURATION ID = SYS9     . IODF  DEFICE NUMBER = 0123
IEA091I NUCLEUS 1 SELECTED
IEA370I MASTER CATALOG SELECTED IS ICFCAT.VMVS422
IEA372 DUMBR14  EXCLUDED FROM NUCLEUS
*IEA247I USING IEASYS00 FOR RELEASE 03.8  , VERSION SP4.2.2
JBB4422
```

```
IEA826 IEASVC00: SVC234: TYPE 4 ROUTINE IGC0023D NOT FOUND.
   <====
IEA168I VATLST09: VATLST DEFAULT USE ATTRIBUTE OF PRIVATE USED.
   <==
IEA191I CONSOLE E50 DEFINED AS MASTER CONSOLE
IEE536I MPF     VALUE ES NOW IN EFFECT
ICH508I ACTIVE RACF EXITS: ICHDEX01
ICH524I INSTALLATION CLASS DESCRIPTOR TABLE PROCESSES
ICH520I RACF 1.8.1 IS ACTIVE.
DSN3100I ! DSN3UR00 - SUBSYSTEM DBV2 READY FOR !START COMMAND
COF025I VLF IS NOW ACTIVE
```

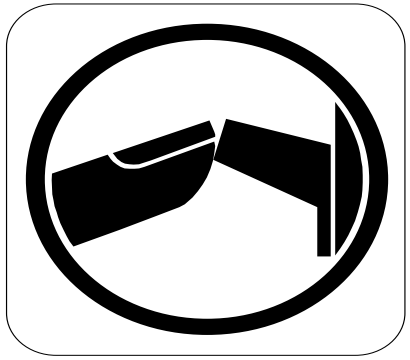
Level II - MVS Supported

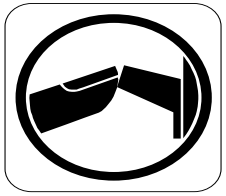
One-time updates to SYS1.PARMLIB members provides lasting documentation

- several IEASYSxx ‘director’ parms allow LIST option
- some stored START commands support LIST option
- select members have listing parameters

SYSLOG traffic increase at IPL

No impact on SMF volume





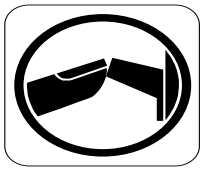
Parameter Updates

LOADxx allows SYSPARM (xx,L) to list IEASYSxx on syslog.

IEASYSxx directors:

- CLOCK=(xx,L) Timezone adjustment, ETR definition
- CON=(xx,L) Console definitions*
- FIX=(xx,L) Fixed Link Pack definition
- ICS=(xx,L) Installation Control Specifications (tuning)
- IPS=(xx,L) Installation Performance Specifications (tuning)
- LNK=(xx,L) Linklist (logical SYS1.LINKLIB) definition
- LPA=(xx,L) Link Pack Area definition
- MLPA=(xx,L) Modified Link Pack Area (system test) definition
- OPT=(xx,L) Tuning Options specification
- PAK=(xx,L) LPA packing definition
- SCH=(xx,L) PPT & other parameters
- SVC=(xx,L) Installation SVCs definition

*** Console definitions impacted by sysplex environment**



Parameter Updates

parmlib COMMNDxx member

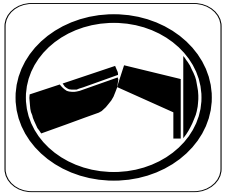
- START APPC,SUB=MSTR[,APPC=(nn,..)][,L] SP4.1
- START ASCH,SUB=MSTR[,ASCH=(nn,..)][,L] SP4.1
  parmlib IEFSSNxx m

ember

- PROMPT=DISPLAY on IGDSMS definition

parmlib SMFPRMxx member

- LISTDSN parameter



Level II Log Messages

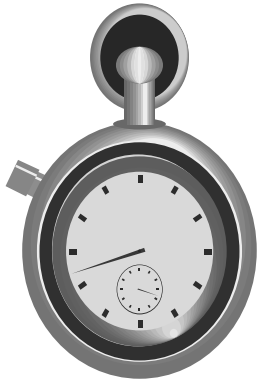
```
IEA713I LPALST LIBRARY CONCATENATION
LPA=(xx,L)
SYS1‘.LPALIB
SYS1.ISPF.V3R2.LPALIB
SYS1.CICS.LPALIB

IEA331I LINK LIBRARY CONCATENATION
LNK=(xx,L)
SYS1.LINKLIB
SYS1.MIGLIB
SYS1.CMDLIB
SYS1.CAILIB

IGD301I SMS PARAMETERS                              PROMPT=DISPLAY in IEFSSNxx
ACDS     = SYS1.SMSACDS

IEE949I hh.mm.ss SMF DATA SETS                                 LISTDSN in SMFPRMxx
          NAME      VOLSER SIZE(BLKS) %FULL  STATUS
        P-SYS1.MAN1 MVSPK1       900     13  ACTIVE
        S-SYS1.MAN2 MVSPK2       900      0 ALTERNATE
        S-SYS1.MAN3 MVSPK5      1500    100 DUMP REQUIRED
```

Level III - Operator Requested

Operator commands journal requested information to SYSLOG

- DISPLAY (generally system-wide processes)
- MODIFY (specific address spaces)
- SET (generally system-wide, with some specifics)

Can use JES timed commands or automated operations timed commands



Operator Commands

SP	System Service	DISPLAY	MODIFY	SET
4.3	APF Library list	D PROG,APF	---	T PROG=(xx)
4.1	APPC (Advanced Pgm to Pgm Commun)	D APPC	--	T APPC=(xx,L)
4.1	ASCH (APPC scheduling)	D ASCH	---	T ASCH=(xx,L)
2.x	Consoles	D CONSOLES	---	---
4.1	Console Groups (alternate console defn)	D CNGRP	---	T CNGRP=(xx,L)
3.1	Data Lookaside Facility (HIPERBATCH)	---	F DLF,STATUS	---
4.1	I/O Subsystem configuration	D IOS,CONFIG	---	---
4.1	MVS Messaging Service	D MMS	---	---
2.x	Message Processing Facility	D MPF	---	---
4.2	Program Properties Table	---	---	T SCH=(xx,L)
2.2	System Management Facility	D SMF[,S][,O]	---	---
4.1	Cross System Coupling Facility	D XCF,SYSPLEX	*D XCF,COUPLE*	*D XCF,PRSMPOLICY*
3.1	Storage Management Subsystem	D SMS[,A],[OPTIONS]	---	---

OS/390 Operator Commands

OS	System Service	DISPLAY	SET
1.1	Product Registration	D PROD	SET PROD=
1.2	Dynamic Parmlib	D PARMLIB	SETLOAD
1.2	IPL information	D IPLINFO	
1.3	LNKLST info	D PROG,LNKLST	SET PROG= SETPROG LNKLST
2.4	LPALST info	D PROG, LPA	SET PROG= SETPROG LPA

Level III & IV Log Messages

```
D MPF
NESSAGE-ID SUP RETAIN USEREXIT  MESSAGE-ID SUP RETAIN USEREXIT
MPF=xx
IEA994E     N  Y       UXDUMPIT  IEFTMS0    N   Y      UXREPLYU
GENERAL WTO USER EXIT (IEAVMXIT) INACTIVE
.  .  .
COMMAND USER EXITS NOT FOUND

SET SMF=FF

IEE949I  09.06.58 SMF DATA SETS ####       <LISTDSN IN SMFPRMFF>
      NAME          VOLSER SIZE(BLKS)  %FULL  STATUS
     P-SYS1.MANx    volum1       900      24   ACTIVE
     S-SYS1.MANy    volum2       900       9   DUMP REQUIRED
     S-SYS1.MANz    VOLUM3       900       0   ALTERNATE
```

Level IV intercepted message

```
COFxxxI  MODIFY VLF COMPLETE
F VLF,STATUS     <= issued by automated operations / MPF
VLF status display including exits
```

Level IV - Ambitious Self-Documentation

Tools

- SMFWTM if APF authorized
- WTO regardless of APF status
- Change monitor or system audit package
- Real time system modifiers MUST report their changes

Reporting

- SMF90 records
- Other critical SMF records
  - 00 - IPL
  - 07 - SMF lost

MPF or automated operations

- force “(xx,L) on appropriate commands
- write journal records based on console messages
  - e.g. F DLF,STATUS when F DLF,NN=nn completes
- track critical applications

Level IV Report Sample

SMF Record x90 Analysis

SYSID: SYS9 January 6, 1997

Date and Time	Revu	Action	New Val	Old Val
06JAN @ 09:02		IPL SMF		
06JAN @ 09:02		IPL Tuning	IEAICS00	IEAIPS09
06JAN @ 09:03		SET MPF	MPFLST09	
06JAN @ 09:03		SET DAE	ADYSET00	
06JAN @ 10:02		SET APPC	APPCPM09	
06JAN @ 10:02		SET ASCH	ASCHPMMH	
06JAN @ 10:50		MODIFY DLF	COFDLF17	(instln)
06JAN @ 11:15		SWITCH SMF		
06JAN @ 13:02	>>	SET SCH	SCHED99	
06JAN @ 14:30	>>	SWITCH SMF		SYS1.MAN3

Reviewing the Environment
SMF Record 90

SMF
SET SMF (5)
SWITCH SMF (6)
HALT EOD (7)
IPL SMF (9)
SETSMF (13)
SET SMF (15) restart

TOD
SET TIME (1)
SET DATE (2)

APPC/MVS
SET APPC (19)
SET ASCH (20)

Misc
SET MPF (14)
SET DAE (16)
SET PFK (17)
SET GRSRNL (18)

Performance Controls
IPL SRM (10)
SET IPS (4)
SET OPT (11)
SET ICS (12)

Security Impact
SET SCH (21)
SET PROG(NO)
Activate LNKLST (29)
Dynamic LPA (31)

***BY THE WAY*:** How many of your shops issue a HALT/Z EOD before QUIESCING the system? **YOU SHOULD!!**

Summary

MVS is already writing SYSLOG and SMF -- are you reading it?

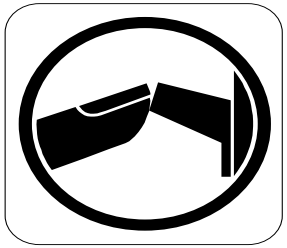
The more environmental data the system collects, the less you have to

Determine the level of data collection you desire -- then get it

SYSLOG is easily readable, hard to trend analyze; SMF is good for trending, but hard to read

MVS construction is increasing in complexity, ensure data collection is dynamic

The STATIC AUDIT is DEAD!



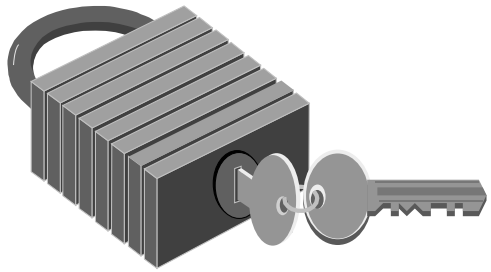
Specific parmlib Member Review

Reviewing Members

- Focus on important (system critical) members
- WHY are they critical
  - APF
  - Data set definition
  - Exit definition
  - other
- HOW can actual in-use values be verified?
- Wrap up on remaining member functions

Parmlib Controls

•While these are “ideal” from a security and audit perspective, not all may be attainable



•Use ISPF option 3.5 to set base level for change tracking.

•Require STATISTICS ON for all updates.

- No statistics is a red flag
- SORT CHANGED and see what changed recently

•Strongly encourage comment lines showing who, did what and why.

•Assign primary contact for each member of parmlib.

•Work towards old/new pairs: for easy rollback.

•Ensure operations is aware of fallback procedure.

•Define parmlib data set(s) as corporate resource - not systems resource

- Updates require levels of approval, based upon impact of member
- Work towards change control - with emergency procedure

•Delete unused and unnecessary members; also only IBM system parameters **REQUIRED** to be housed.

•Create a “base copy” and periodically review.

•Work towards a “self documenting” IPL

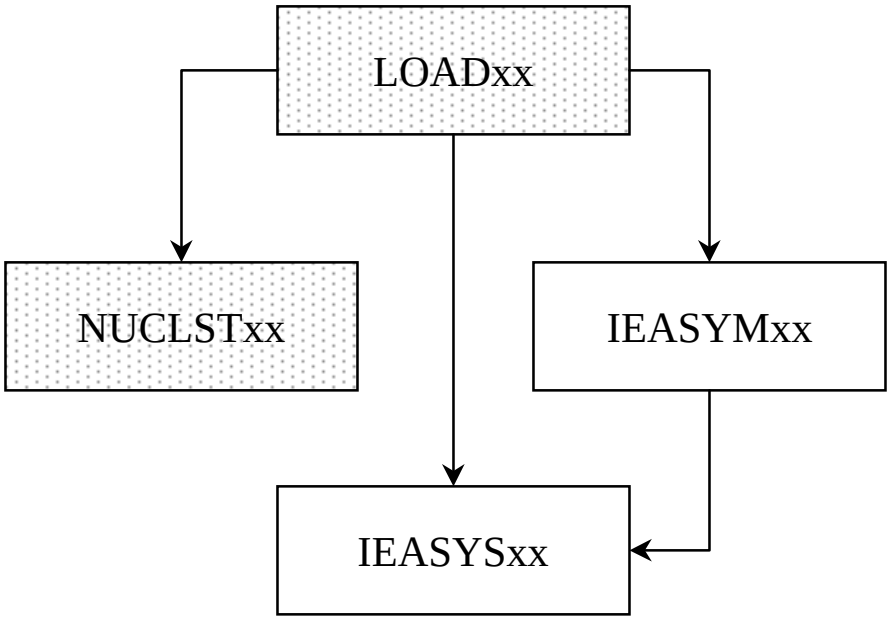
System initialization

- These members define the building blocks of the system and determine which sets of parameters to use

  - **LOADxx** - Build the system parameters
    - **NUCLSTxx** - Modify the nucleus (if needed)
    - **IEASYMxx** - System symbolics and
      - **IEASYSxx** - Specify most of the system parameters

LOADxx - System Configuration Data Sets

- Specifies system image specific parameters.
- Root of all parameters
- Names the sysplex (optional, recommended) and the system image.



- Defines the master catalog, nucleus, IEASYSxx member, IEASYMxx member (which can also specify (IEASYSxx), and *parmlib* concatenation.
- Has filters, enabling multiple system images to use same
- Should be in SYSn.IPLPARM
- Consider ",L" on SYSPARM statement
- Column dependent member
- Most values written to SYSLOG

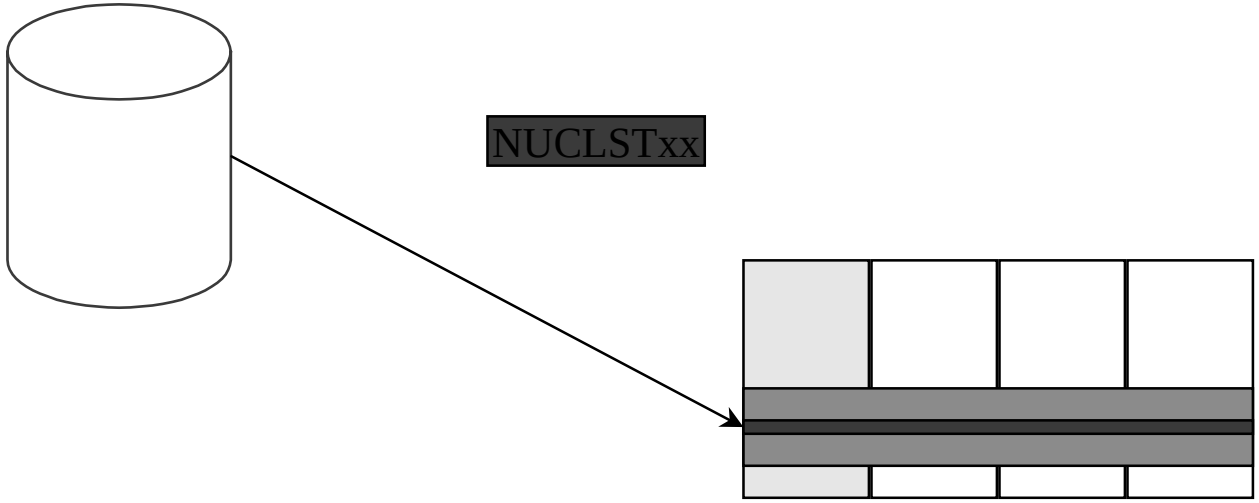
LOADx
System Loading Parameters

```
*** Altered by SSP06 at 7:18am on 06/13/95 for HBB4422
*---+----1----+----2----+----3
NUCLEUS  1
SYSPARM  00
IODF     01 SYS1      SYSV    00
SYSCAT   MVSV21113CICFCAT.VMVSV21

NUCLST   xx
```

```
IEA317I SYS0.IPLPARM IB DEVUCE 9395 SEKECTED FIR UOK OARANETERS
IEA246I LOAD ID 00 SELECTED
IEA519I IODF DSN = SYS1.IODF00
IEA520I CONFIGURATION ID = SYSV   . IODF DEVICE NUMBER = 0305
IEA097I I/O CONFIGURATION 00 SELECTED
IEA091I NUCLEUS 1 SELECTED

IEA247I USING IEASYS00 FOR RELEASE 03.8  , VERSION SP4.1.0 HBB4410

IEA370I MASTER CATALOG SELECTED IS ICFCAT.VMVSV21
```

NUCLCTxx - Nucleus Customization

- Must be in same data set as LOADxx - therefore SYSn.IPLPARM
- Likely not present
- Written justification if so.



NUCLSTxx
Nucleus Building Parameters

- Pointed to by LOADxx <4.2.2>
- Must be in same data set as LOADxx
  IBM recommends SYSn.IPLPARM
- Use comments (* in col 1)
- Column sensitive entries - not keyword
- Add, delete or replace modules IN THE NUCLEUS

```
INCLUDE  PDQBR14
*  Remove the failing module from nucleus
EXCLUDE  BADMOD
```

IEASYSxx
System IPL Parameters

Pointed to by LOADxx <4.1> or IEASYMxx <5.2>
IEASYS00 by default

Contains keyword values and directors

- Alphabetize the entries: ease of finding
- Where possible, consider using ",L" for SYSLOG hardcopy of parms

LNKAUTH= define APF-status for LNKLST data sets

- LNKLST - APF if *not* STEPLIB target
- APFTAB - APF only if in IEAAPFxx or PROGxx <4.3>

Sets the system id:for sysplex, &SYSTEM variable
until 5.2 - then use LOADxx or IEASYMxx.

Specifies some system data sets (may use variable)

IEASYSxx - Checklist

- This member should be closely audited
- Alphabetize the parameter entries
- Review Director entries and use (xx,L) whenever appropriate
- Backout member controls are appropriate
- Consider a minimal parameters for emergency use
- Consider use of localized OPI=NO
- Require comments documenting changes, changer and reason

```
LNKAUTH=APFTAB,
LOGREC=&SYSNAME..SYS1.LOGREC,
PROG=(01,02,OPI=NO),
SMF=00,
```

System Routines

- These specify system routines (not exits)
- Specified by **IEASYSxx**
  - **IEALPAxx** - Modified Link Pack Area (MLPA) test exits
  - **IEASVCxx** - Installation Supervisor Call definitions
  - **IEFSSNxx** - Subsystems
  - **SCHEDxx** - Program Properties Table (PPT)
- Not specified by IEASYSxx
  - **IKJTSOxx** - TSO APF services
    - defaults to IKJTSO00
    - changes by TSO *PARMLIB UPDATE(xx)*

Data Sets Specification

- Also define system data sets as a parameter
  - **APPCPMxx** - Advanced Program to Program Communication - *DISPLAY APPC* - Sideinfo data set
  - **CSVRTLxx** - Run Time Library Services (2.4)
  - **COUPLExx** - *DISPLAY XCF* - Communication data sets
  - **EPHWP00** - UNIX System Services MAN files (2.7)
  - **IEAAPFxx** - Static APF list
  - **IEASYSxx** - Includes system data sets; may use &variables
  - **LPALSTxx** - Link Pack Area library list
  - **LNKLSTxx** - Link Library list
  - **LOADxx** - mastercat, multiple parmlibs
  - **MMSLSTxx** - MVS Message Services (translation) - *DISPLAY MMS* - Message data sets
  - **PROGxx** - Dynamic APF (*D PROG,APF)*, LNKLST (*D PROG,LNKLST)* or LPA *(D PROG,LPA)* - override LNKLSTxx and LPALSTxx default; define Linklist, add modules to LPA after IPL.
  - **SMFPRMxx** - SMF parameters including SMF data sets - *D SMF,O* and *D SMF,S*

Exit Specifications

- Also define exit routines as parameters
  - **COFDLFxx** - Hiperbatch controls
    - which data sets to retain; superceded by RACF 1.9 controls
  - **COFVLFxx** - Virtual Lookaside Facility
    - Tuning and statistics exits
  - **EXITxx** - Dynamic allocation exits
    - Exits for allocation recovery situations; no real security or audit concerns
  - **IEALPAxx** - Modified Link Pack Area (MLPA) test exits
  - **IEAPP00** - I/O appendages
    - Exits for unusual I/O conditions; routines in SYS1.SVCLIB
  - **MPFLSTxx** - exits for specific messages and MVS command input manipulation
  - **SMFPRMxx** - SMF exits

MVS Tuning Members

- These members specify **tuning and performance** controls
  - **IEAICSxx** - Installation Control Specifications
  - **IEAIPSxx** - Installation Performance Specifications
  - **IEAOPTxx** - Performance Options

- These members specify **storage performance** parameters
  - **CSVLLAxx** - Library Lookaside
  - **IEABLDxx** - (superceded by CSVLLAxx)
  - **IEAFIXxx** - page fix these modules
  - **IEALOD00** - MVS/370 fixed storage for modules
  - **IEAPAKxx** - “packing list” for LPA modules

- These members relate to **TSO and Telecommuncations**
  - **IKJPRM00** - TCAM parameters (if still used?)
  - **TSOKEY00** - TSO/VTAM parameters

MVS Debugging

By far, the most entries are for **IPCS and Component Tracing**

- **BLSCxxxx** (read by BLJSPRMI -
- **xxxIPCSP** at IPL)
- **CTIxxxxx** (specified by system component)

Dump Services - read at IPL - totally static

- **IEAABD00** - SYSABEND dump parms
- **IEACMP00** - SYSUDUMP dump parms
- **IEADMR00** - SYSMDUMP dump parms
- **IEADMCxx** - Dump command parameters (2.6)

Dump Suppression

- **ADYSETxx** - Suppress duplicated dumps
- **IEASLPxx** - Suppress dumps based on ABEND code

DIAGxx - Common Storage Area tracking
IVTPRM00 - Communications Storage Manager (CSM)

Misc Members

- **IPL command** members
  - **COMMNDxx** - Installation commands for IPL
  - **IEACMD00** - IBM standard commands for IPL

- **Multiple system support**
  - **GRSCNFxx** - Global Resource Serialization configuration
  - **GRSRNLxx** - Global Resource Serialization resource names
  - **XCFPOLxx** - Sysplex policy member

Misc Members

- These members are hard to group
  - **ALLOCxx** - New DASD data set allocation defaults
  - **ASCHPM** - APPC scheduling
  - **CLOCKxx** - Time zone offset and ETR controls
  - **CNGRPxx** - Alternate console groups
  - **CONFIGxx** - Configuration comparison
  - **CNLcccxx** - Time and date translation formats (MMS)
  - **DEVSUPxx** - Data compaction on newer tape drives
  - **EXSPATxx** - Excessive Spin loop response
  - **GTFPARM** - actually ANY name desired:GTF parameters
  - **IECIOSxx** - Missing interrupt thresholds for incomplete I/O operations
  - **IGDSMSxx** - Storage Management Subsystem parameters
  - **IKJTSO00** - Misc TSO/E parameters
  - *MVIKEY00 - Mass Storage System Controls(MSSC) was obsoleted by 4.1*
  - *PARMTZ - Replaced by CLOCKxx in XA 2.2*
  - **PFKTABxx** - Console Program Function Keys (PFK) definitions
  - **VATLSTxx** - DASD Volume Attribute List

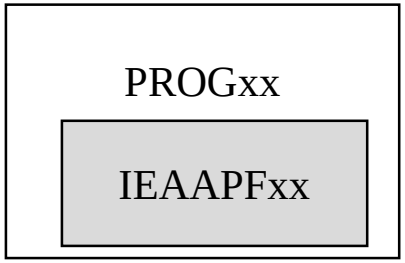
IEAAPP00 and IEALPAxx

- **IEAAPP00** - I/O appendages
  - Mostly a thing of the past; largely resolved by APF
  - Which special routines to gain control during I/O processing
  - Must reside in SYS1.SVCLIB; named IGG019WA - IGG019ZZ

- **IEALPAxx** - Modified Link Pack Area (MLPA) Modules
  - HIGH risk system exits
  - Allows testing with easy re-IPL fall back
  - Review ALL such members in parmlib
  - Encourage the ",L" operand to document what is in MLPA
  - DO NOT expect to MLPA=(xx) in IEASYSxx; if so encourage MLPA=(xx,L)

PROGxx Dynamic APF

- Dynamic APF introduced in MVS SP4.3



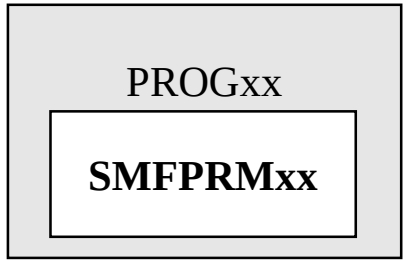
- Supercedes IEAAPFxx, but MVS merges
- Recommendations
  - Separate EXIT controls from APF list
  - Alphabetize data set entries -- easier to find
- FORMAT=STATIC
  - 253 library limit enforced
  - Similar to IEAAPFxx, keywords changed
- FORMAT=DYNAMIC
  - Beats the 253 library limit
  - Support operator changes to APF list - but NO SMF90 written
    - SET PROG=(xx[,L])
- DISPLAY PROG,APF operator command to review

PROGxx Dynamic Exit List

- Dynamic exits added in MVS SP5.1

- Currently SMF exits only and works with SMFPRMxx for exit definition

- Enables multiple users of standard exit without competition or system programmer intervention



- Recommendations
  - Alphabetize the list of data set names (ease of find)
  - Separate EXIT controls from APF list
  - Require use of comments: who, when, why

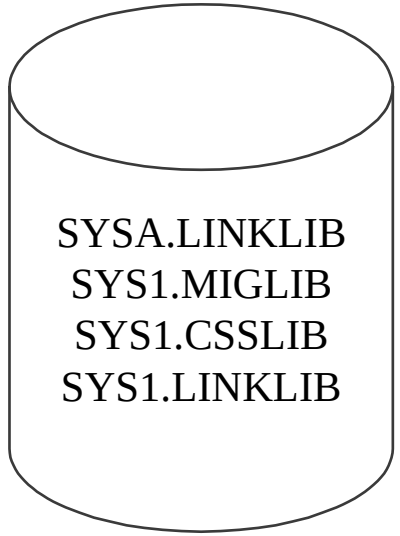
  - Review using DISPLAY PROG,EXIT

```
EXIT ADD EXITNAME(SYS.IEFUJI) MODNAME(R1)
EXIT MODIFY EXITNAME(SYS.IEFUSI) MODNAME(R2) STATE(INACTIVE)
EXIT DELETE EXITNAME(SYS.IEFACTRT) MODNAME(R3)
```

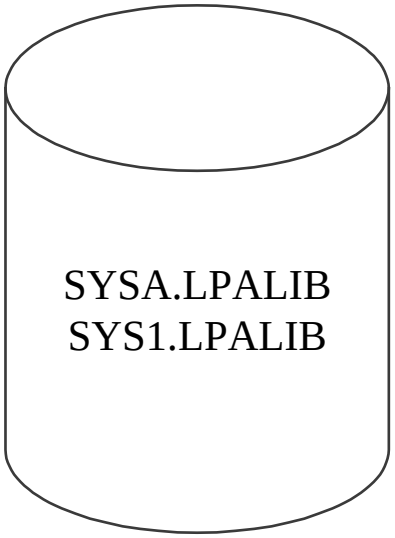
LNKLSTxx and LPALSTxx

- Specify system libraries for building / running the system
  both accept the ",L" specification requesting paramter logging
  - **LNKLSTxx -** Link Library list
    - like a PC PATH= statement
    - only specify data set names
      - Should exist, error message if not
      - Must be in Master Catalog, not user catalogs
      - Need to order by frequency of use replaced by CSVLLAxx
      - APF authorized if no STEPLIB and LNKAUTH=LNKLST
  - **LPALSTxx -** Link Pack Area list
    - only specify data set names
      - Must be in Master Catalog, not user catalogs
    - Usual source for system exits and routines shared by ALL users

PROGxx LNKLST and LPALST



- New with OS/390 R2
  - LNKLST statement
    - Defines Linklist (override LNKLSTxx member)
  - SYSLIB statement
    - overrides normal LNKLSTxx default sequence:
      - –LINKLIB(dsn)
      - –MIGLIB(dsn)
      - –CSSLIB(dsn)
    - overrides normal LPALSTxx sequence
      - –LPALIB(dsn)



CONSOLxx Console Definitions

- Added in MVS/XA 2.2
- Pointed to by IEASYSxx, CON=(xx[,L]) parameter
- /* Comments should be used freely */
- points to PF Key table (PFKTABxx)
- Message processing options:
  - Message Processing Facility (MPFLSTxx)
  - MVS Message Services (MMSLSTxx) for translation <4.1>
- Recommend console naming (TAPECONS, etc) <4.1>
- Display current system controls
  - D CONSOLES
  - D MPF

CONSOLxx Console Definitions

- Alternate consoles - properly secured?
  - address
  - name (CNGRPxx) <4.1>
- WTO exit enable (IEAVMXIT)

Console authority: level of commands

  - multiple MASTER authority <4.2>

- Determines if LOGON(REQUIRED, OPTIONAL)
- Determines if years are formatted as yy or yyyy

```
INIT AMRC(NO) MPF(01) PFK(01) UEXIT(Y) MMS(NO) CNGRP(xx)
        CTRACE(CTIOPS00)
CONSOLE DEVNUM(0A0) AUTH(MASTER) NAME(MASTCON)
HARDCOPY DEVNUM(SYSLOG,OPERLOG) HCFORMAT(YEAR/CENTURY)
```

MPFLSTxx
Message Processing Facility

- Pointed to by CONSOLxx

- Dynamic update possible: SET MPF=(xx) / Audit with DISPLAY MPF
  - up to 39 members as of MVS 4.3
  - SMF90(14) written for changes

- Comments supported
  - following specification
  - use /* delimiters */

- Define attributes for messages by type (immediate action, etc)

- SUP(YES) Allows for suppression of WTOs from console displays, but not from SYSLOG

- USEREXIT(modname) specifies exit to execute when message written

- .CMD USEREXIT(exit1, ... exit6) to modify command input <4.1>

MPFLSTxx
Message Processing Facility

```
$HASP100,SUP(YES),RETAIN(YES)
IEA994A,SUP(NO),USEREXIT(U7XDUMPIT)  All DUMP datasets full
.CMD USEREXIT(exit1,...exit6)
```

```
D MPF
IEE189I 13.18.21 MPF DISPLAY nnn
MESSAGE-ID SUP RETAIN USEREXIT MESSAGE-ID SUP RETAIN USEREXIT
MPF=xx
IEA994E     N     Y   UXDUMPIT IEFTMS0     N     Y   UREPLYU
. . .
GENERAL WTO USER EXIT (IEAVMXIT) INACTIVE
COMMAND USER EXITS NOT FOUND
```

SCHEDxx
Program Properties Table

- Specifies 4 types of entry,
  - Mater Trace Table size
  - Eligible Device Table (obsolete)
  - Restart ABEND codes
  - Program Properties Table (PPT) is most important
- PPT properties
  - Must execute from APF library, need not AC(1)
  - **NOCANCEL** - ignore operator CANCEL commands
  - **KEY(n)** - 0-7 are system keys
  - **NOPASS** - DFP ignores SAF calls for data set opens
  - remainder are primarily performance related

```
PPT PGMNAME(IEFIIC) NOCANCEL KEY(0) PRIV SYST SAFF(NONE)
PPT PGMNAME(IEEMB860) NOCANCEL KEY(0) NOPASS PRIV SYST
```

SMFPRMxx SMF Parameters

- System Management Facility data is critical!
- NO member should have INACTIVE!
- Parms specified and security considerations

  - SID(xxxx) 4-character system id for SMF records and (optional) data set name
  - DSNAME(dsn1,dsn2)
    - &SID uses SID(xxxx) value[SP 5.1]
    - &SYSNAME uses SYSNAME= value from IEASYSxx [SP 5.1]
  - **LISTDSN** requests listing data sets and status message at IPL or SET command <These are system critical data sets>
  - Exit specifications work in concert with PROGxx [SP5.1]. Must specify in SMFPRMxx and (optionally PROGxx).
    <These are system critical exits>
  - JWT specifies how long a job may voluntarily wait
  - (NO)PROMPT allows the operator to modify parms when set
  - Note: IBM recommends use of IEFUAV for APPC transactions instead of IEFUJV (Init and Tuning Reference SP 5.1)

SMFPRMxx SMF Parameters

- Parms specified and security considerations (*cont*)
  - NOTYPE(record number & ranges) - SUPPRESS the writing of SMF records. Ensure that 00, 07, 90, security types(e.g. 8x, 230) and others are written
  - NOPROMPT - disallows operator intervention on the parameters (unless an error is detected).

- DoD B1 parameters (or VERY secure)
  Designed to prevent SMF data loss
  - NOBUFFS(HALT) - slow down the system when no more buffers
  - NODSN(HALT) - slow down the system when no more data sets

- Audit using *DISPLAY SMF*,*O* and *DISPLAY SMF*,*S*

- SMF90 records also written:
  - (05) SET SMF; (06) SMF Switch; (09) IPL SMF; (13) SETSMF; (15) SET (restart) SMF

SMFPRMxx - SMF Parameters

```
ACTIVE           /* REMEMBER COMMENTS */
DSNAME(SYS1.MANX,SYS1.MANY)/* or
DSNAME(SYS1.&SYSNAME..SMF) */
NOPROMPT
JWT(hhmm)
SID(xxxx)
LISTDSN  /* CONSOLE LOG DATA SETS AT IPL AND SET */
SYS(NOTYPE(34))
/* keep 00, 07, 90, 8x/230 out of here */
```

```
*IEE362A  SMF ENTER DUMP FOR SYS1.MANA ON MVSA21
 IEE366I NO SMF DUATA SETS AVAILABLE -- DATA BEING BUFFERED

 IEE949I 06.13.53 SMF DATA SETS nnn
          NAME     VOLSER SIZE(BLKS)  %FULL  STATUS
        P-SYS1.MAN1 MVSA21        900    100  DUMP REQUIRED
        P-SYS1.MAN2 MVSA21        900    100  DUMP REQUIRED
        P-SYS1.MAN3 MVSA21        900    100  DUMP REQUIRED
```

Final Summary

- MVS is much more than a collection of *parmlib* members
- Most system facilities interact more than act in isolated fashion
- *parmlib* can be [should be] more than 1 data set
- *parmlib* is a corporate resource and should be protected as such
  - There are several steps available
  - Not all are attainable
- Self-documenting systems make it easier to know what is happening
- While it is possible to categorize *parmlib* members, it isn't easy