SYSTEM INITIALIZATION

SYSTEM INITIALIZATION

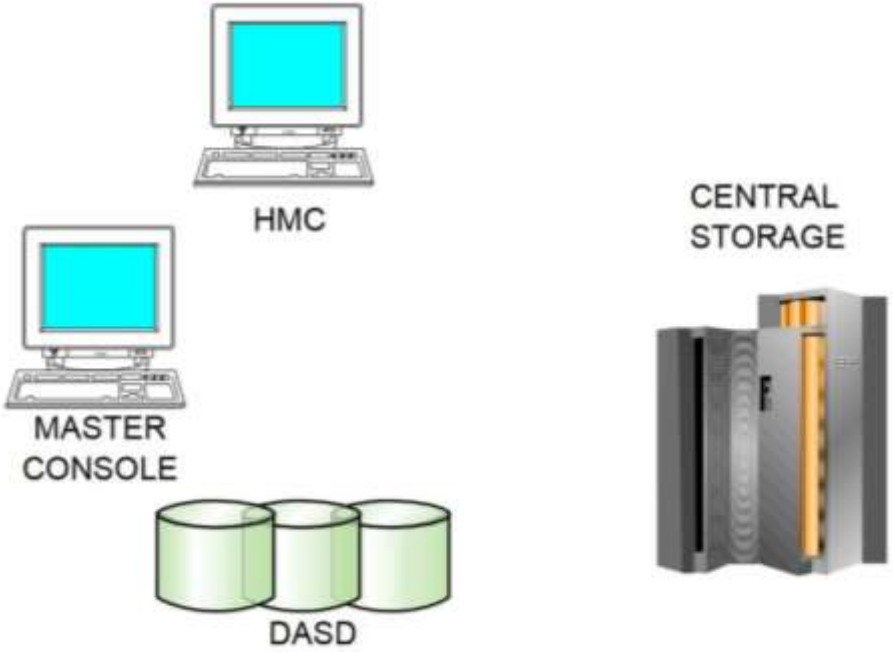
System Initialization

Initialization Process

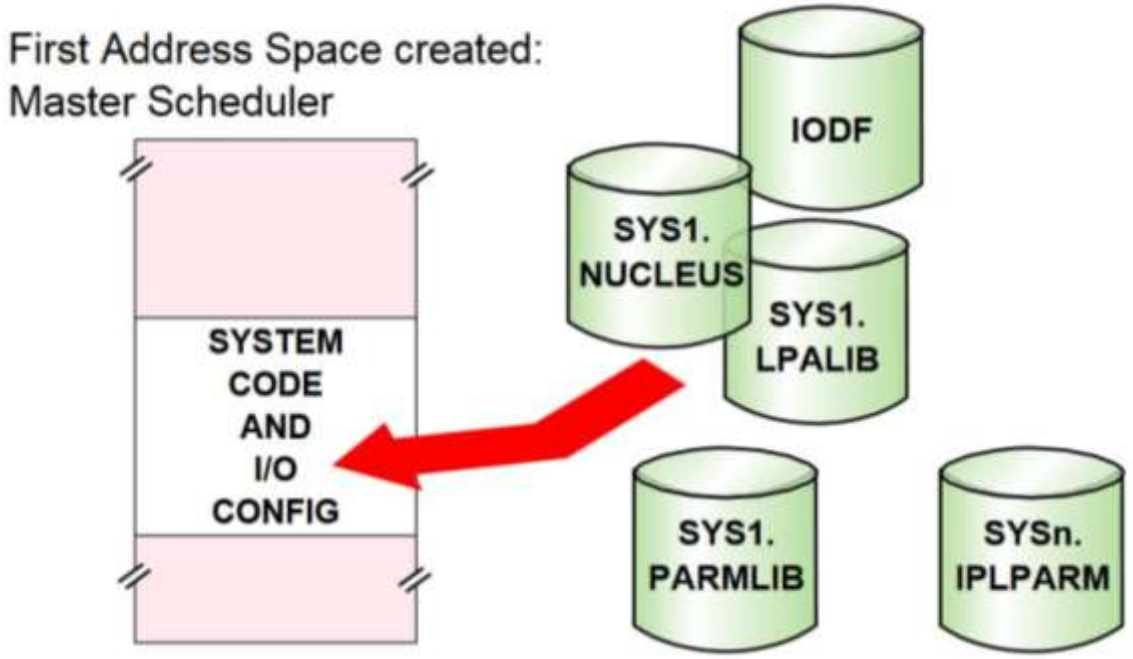
System Address Spaces

Other Address Spaces

HARDWARE REQUIREMENTS



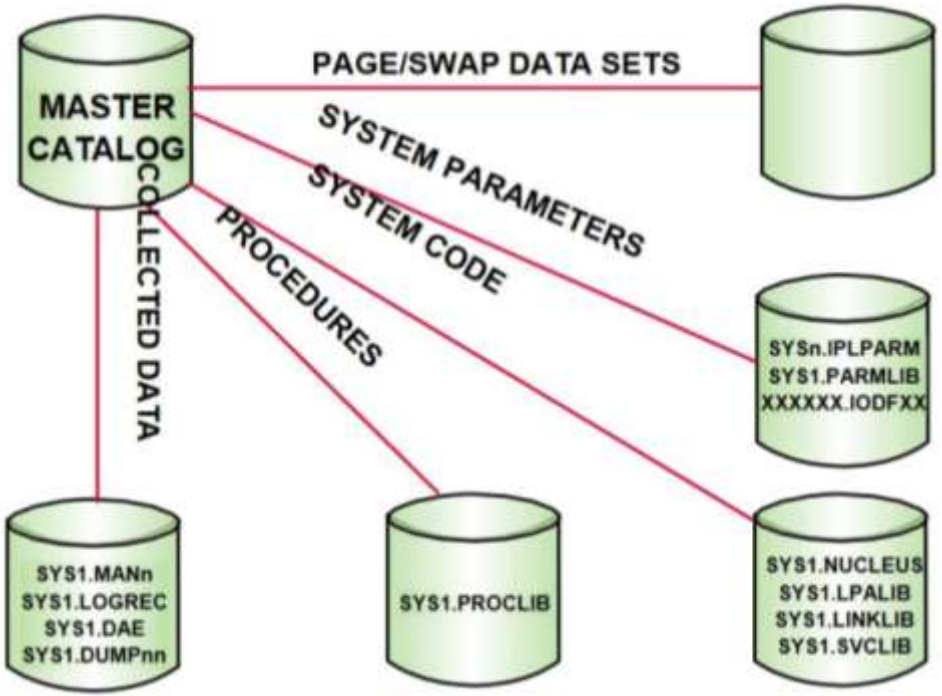
INITIALIZATION



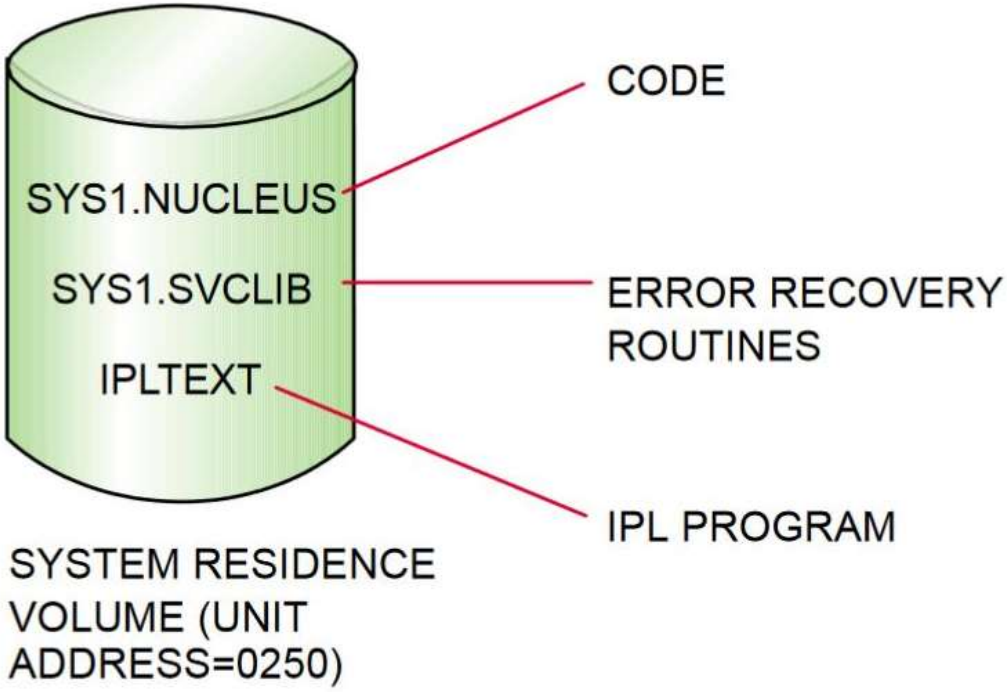
INITIALIZATION STEPS

- Setting the stage
  - System library requirements.
  - Hardware requirements.
- Starting z/OS
  - The initialization process.
  - Starting system address spaces.
  - Starting other address spaces.

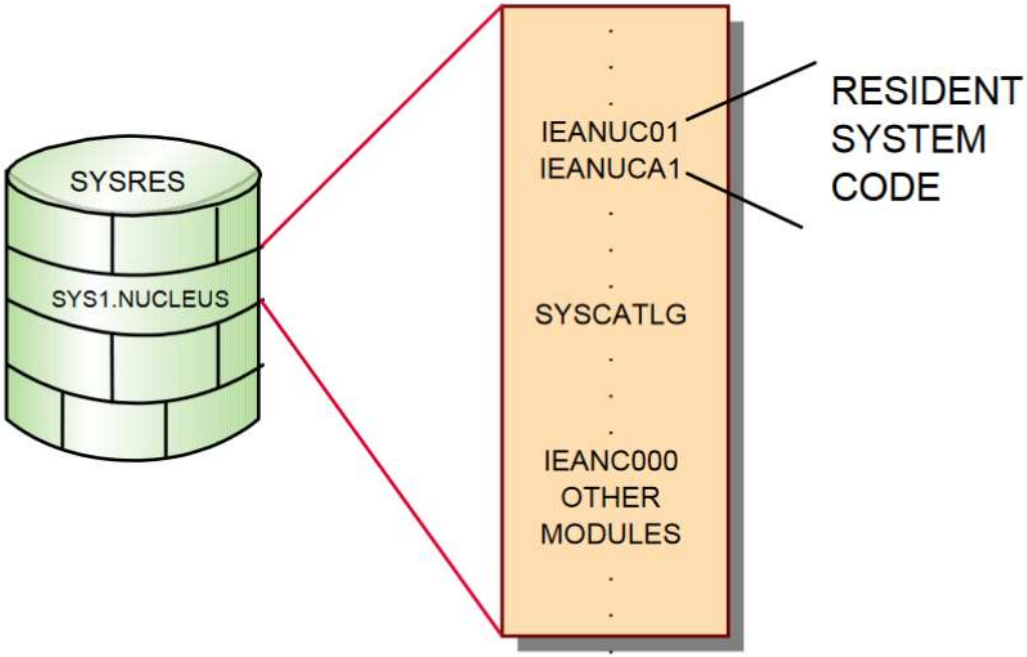
SYSTEM DATA SETS



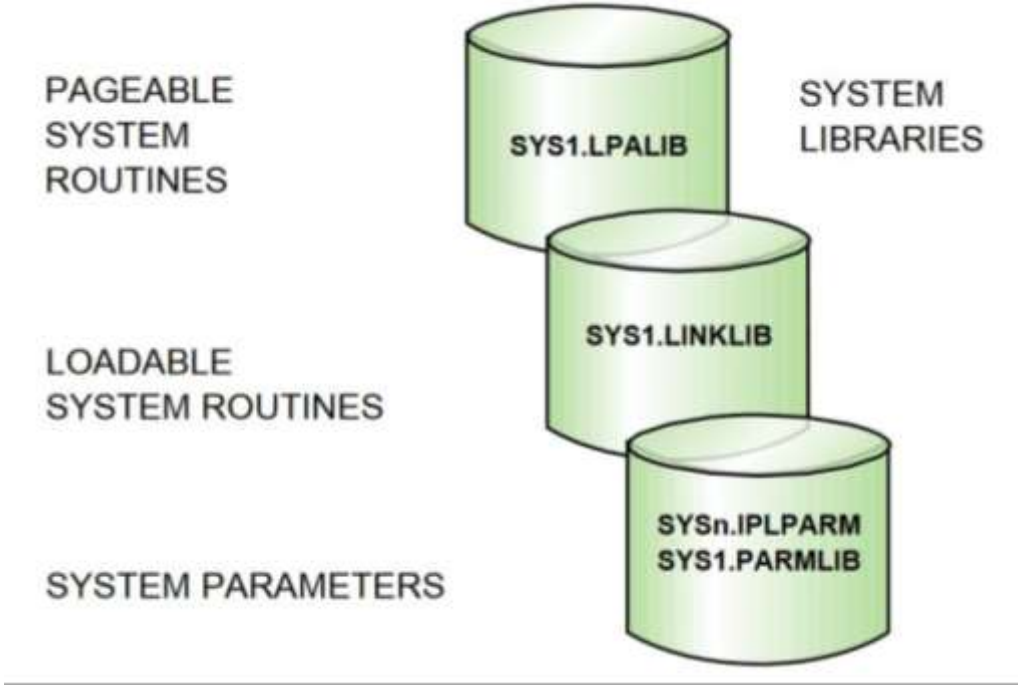
SYSRES VOLUME



SYS1.NUCLEUS

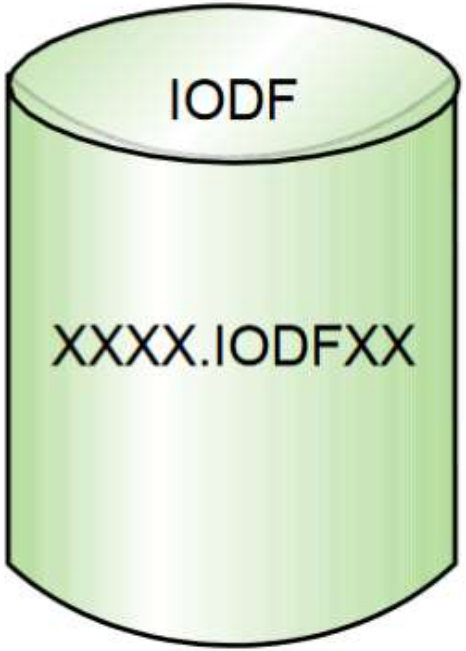


ADDITIONAL SYSTEM LIBRARIES

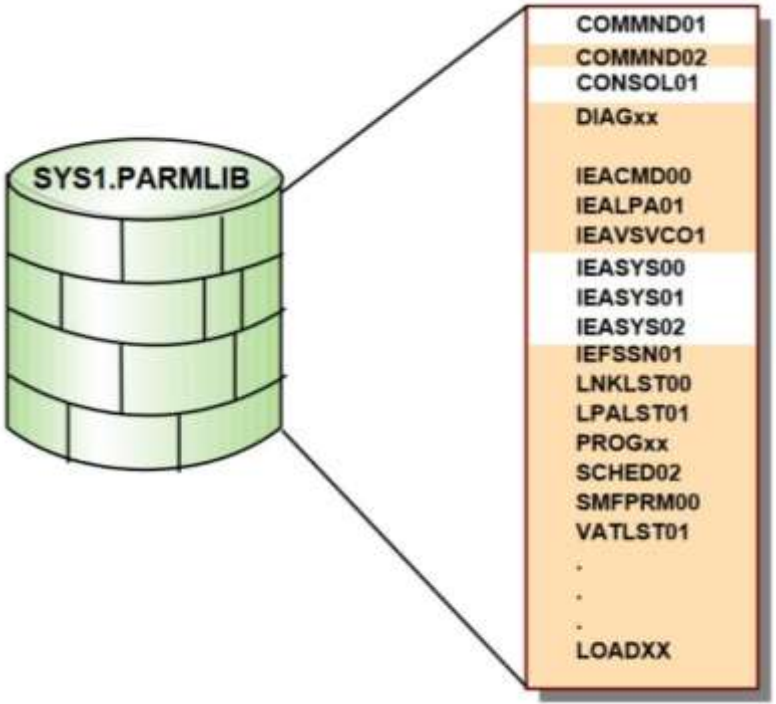


INPUT/OUTPUT DEFINITION FILE DATA SET (IODF)

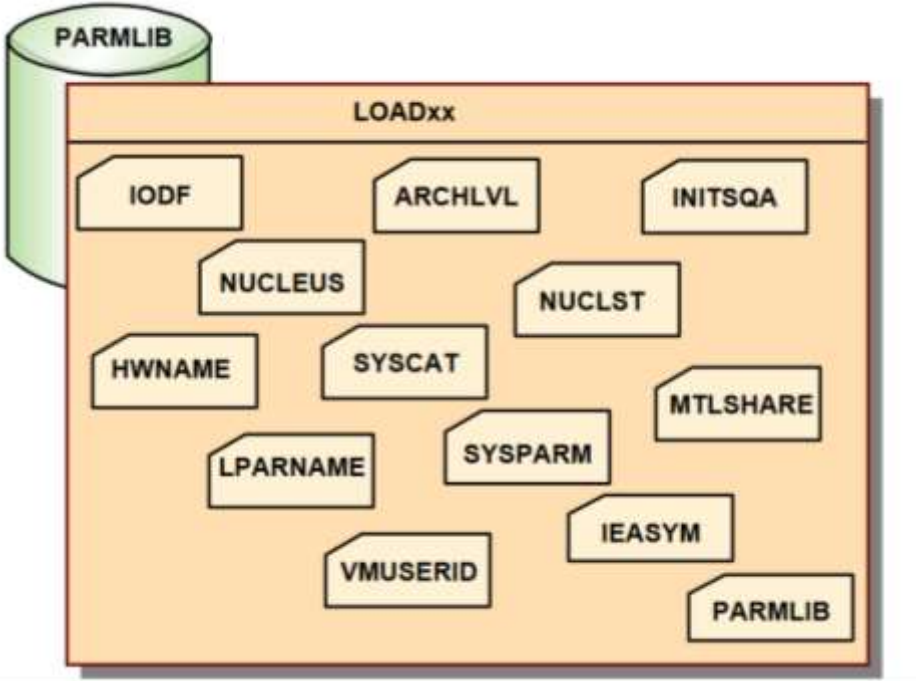
I/O
CONFIGURATION



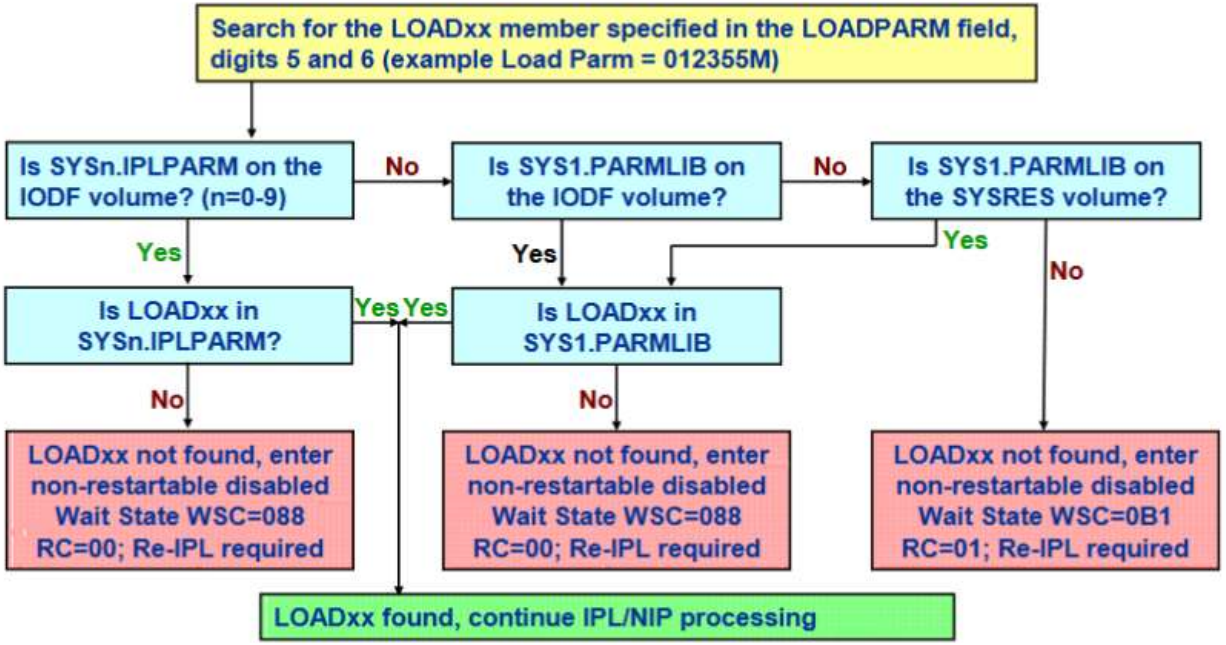
SYS1.PARMLIB



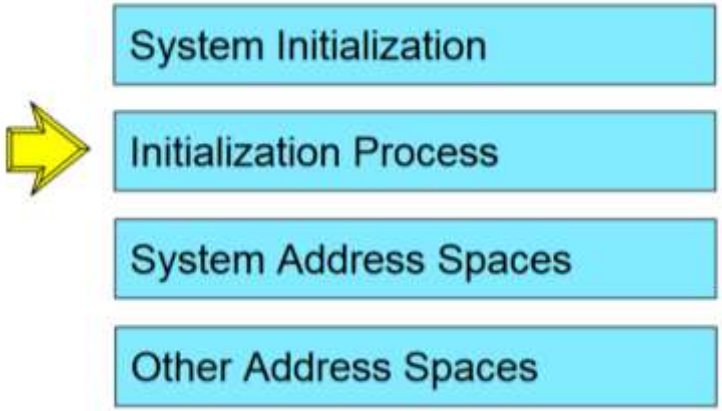
SYSn.IPLPARM (LOADxx)



LOADxx SEARCH SEQUENCE



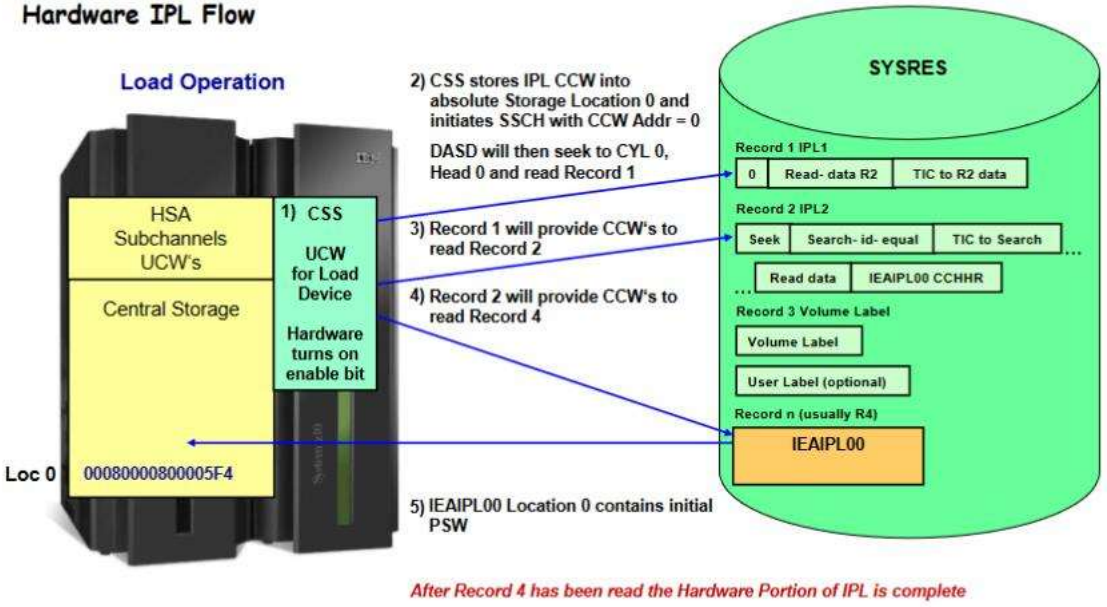
INITIALIZATION PROCESS



THE INITIAL PROGRAM LOAD

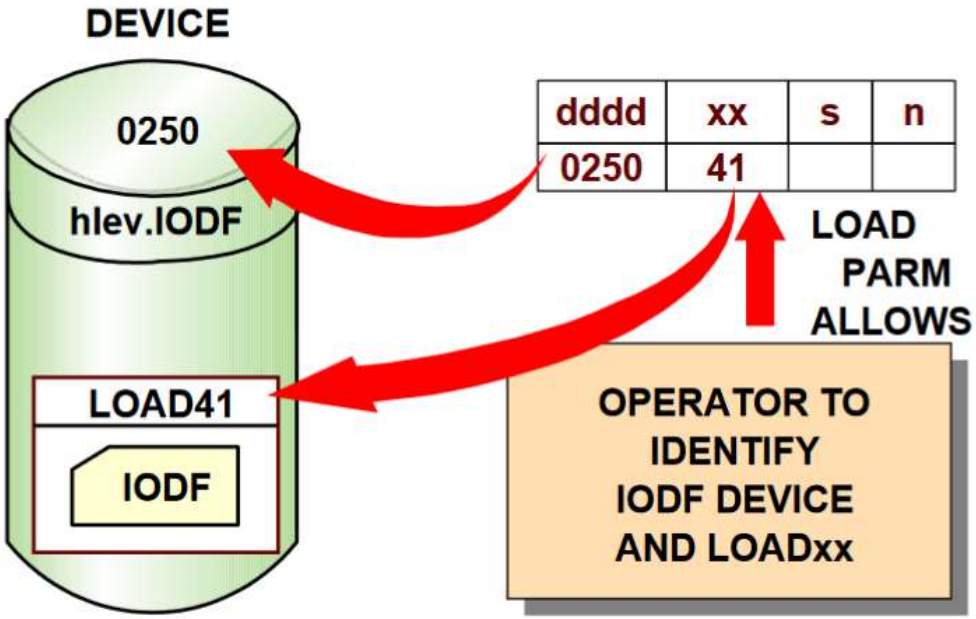
- Load nucleus.
- Locate Master Catalog and Other System Data Sets.
- Initialize z/OS Using SYS1.PARMLIB/SYSn.IPLPARM.
- Build First Address Space.
- Create Other System Address Spaces.

HARDWARE IPL FLOW

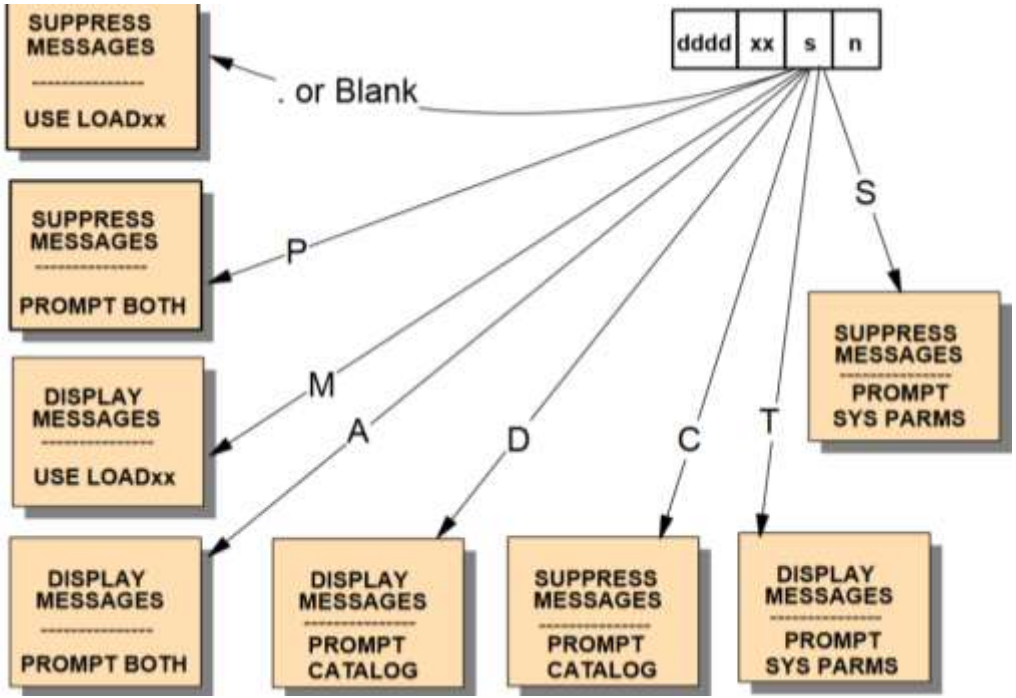


(CSS: Channel Subsystem).

LOADxx AND IODF LOCATION



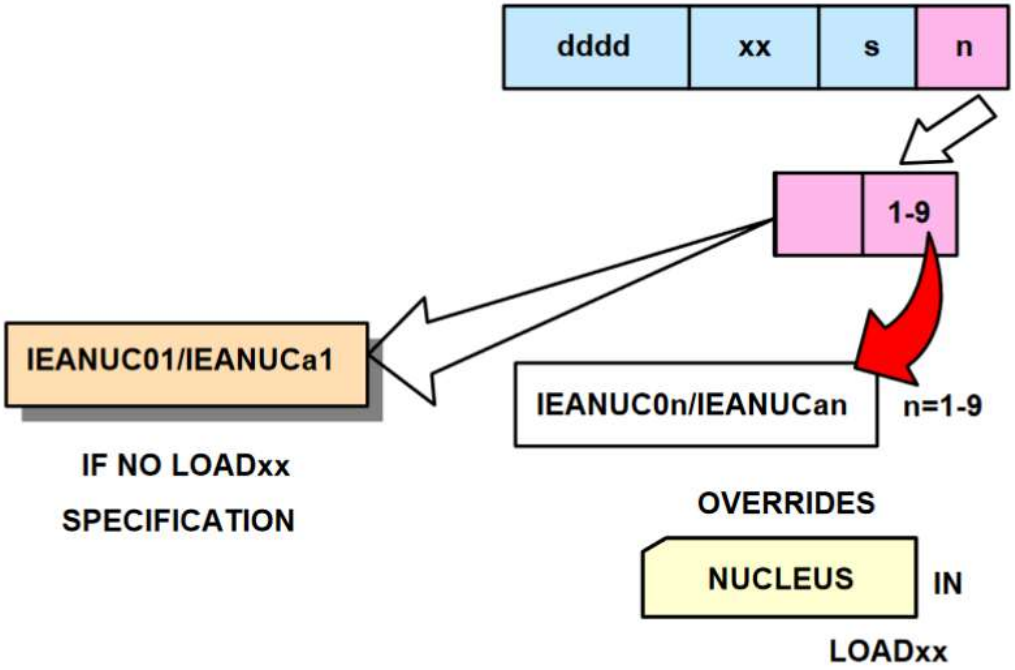
MESSAGE SUPPRESSION



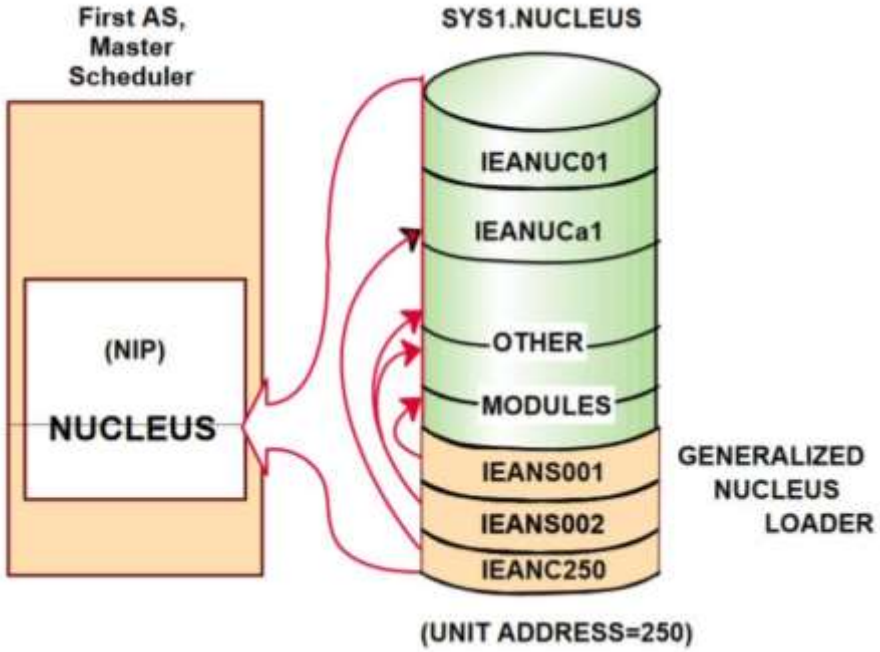
IMSI: Initialization Message Suppression Indicator

IMSI character	Display informational messages	Prompt for master catalog response	Prompt for system parameters Response
period (.) or *blank*	No	No	No
A	Yes	Yes	Yes
C	No	Yes	No
D	Yes	Yes	No
M	Yes	No	No
P	No	Yes	Yes
S	No	No	Yes
T	Yes	No	Yes

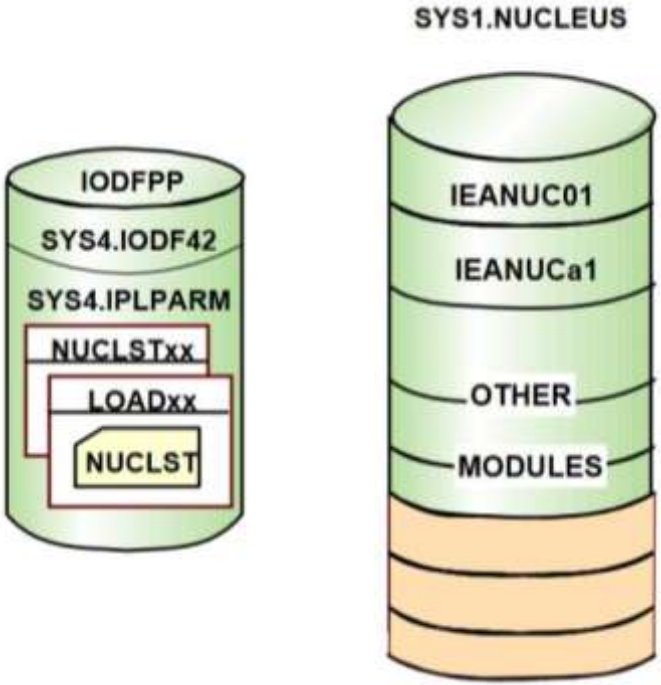
NUCLEUS IDENTIFIER



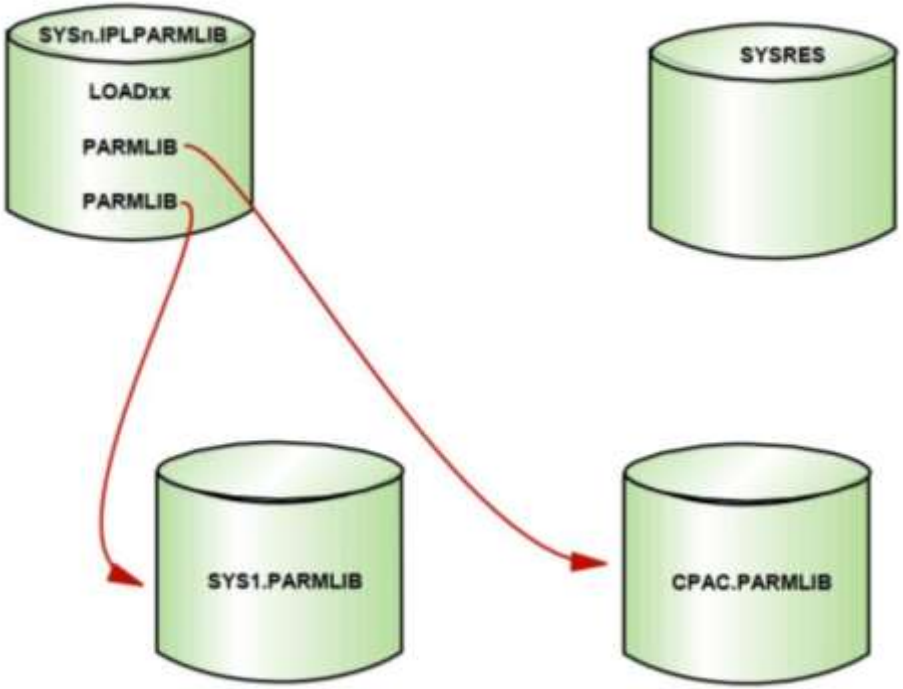
LOADING THE NUCLEUS



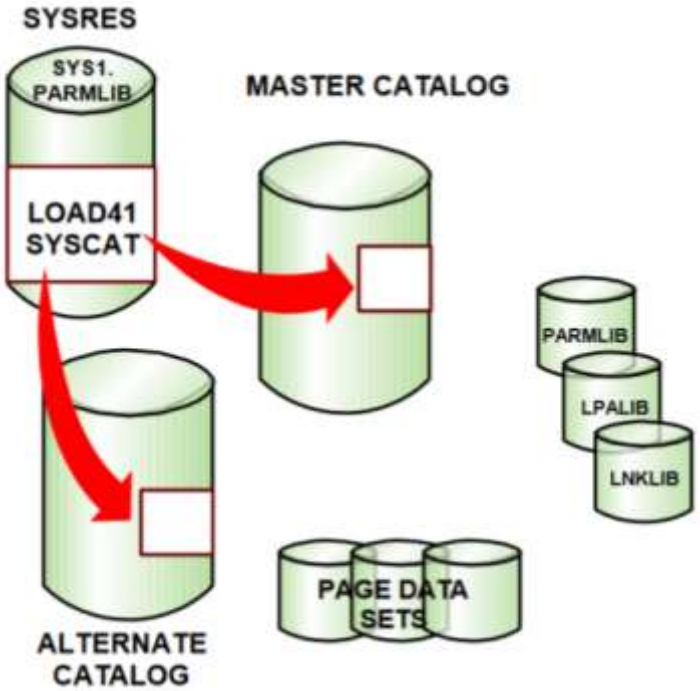
NUCLSTxx


IODFPP
SYS4.IODF42
SYS4.IPLPARM
NUCLSTxx
LOADxx
NUCLST
SYS1.NUCLEUS
IEANUC01
IEANUCa1
OTHER
MODULES

PARMLIB DATA SET SELECTIONS

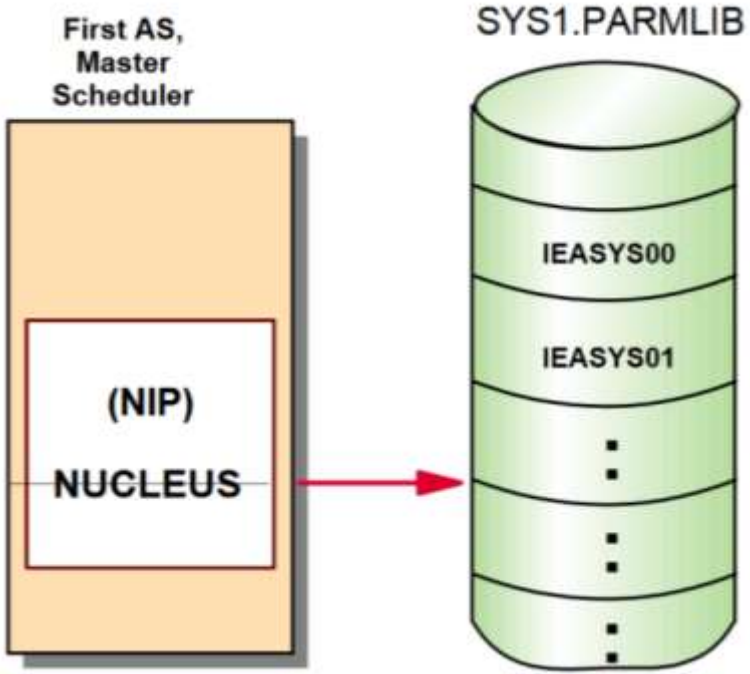


MASTER CATALOG LOCATION

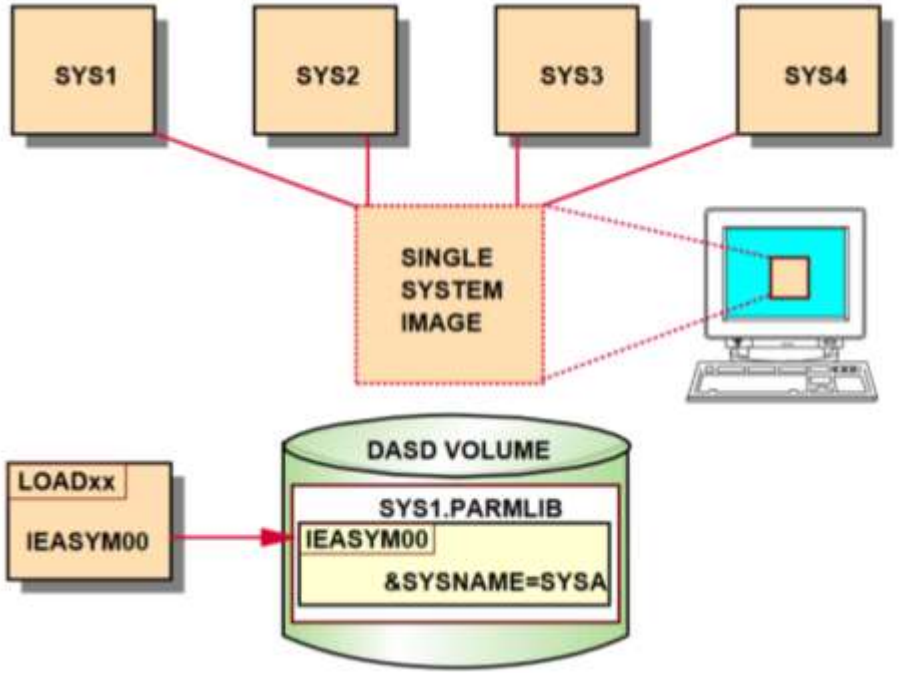


OPERATOR PROMPT (Master Catalog)

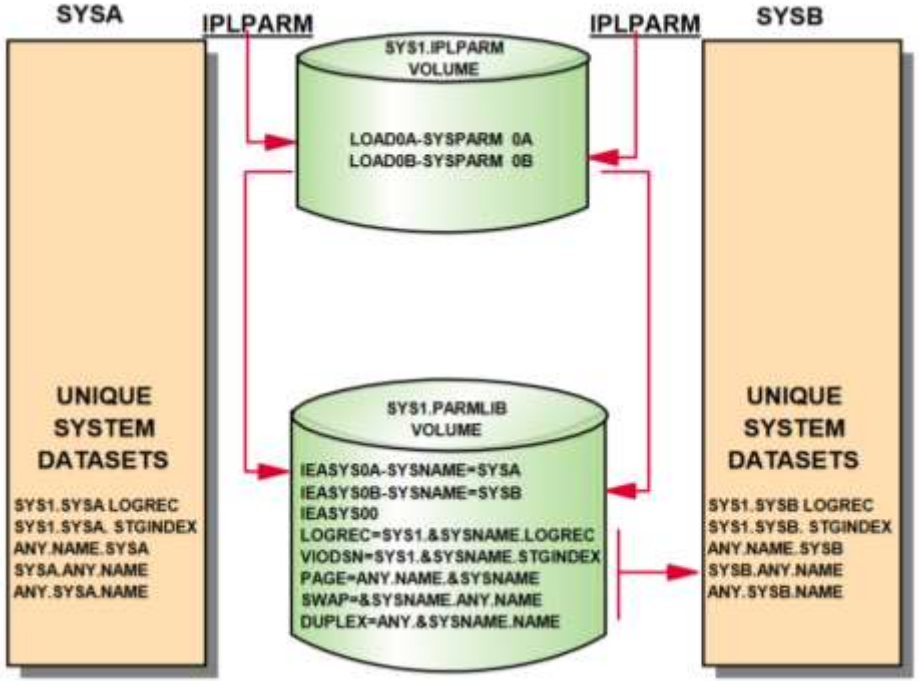
SYS1.PARMLIB



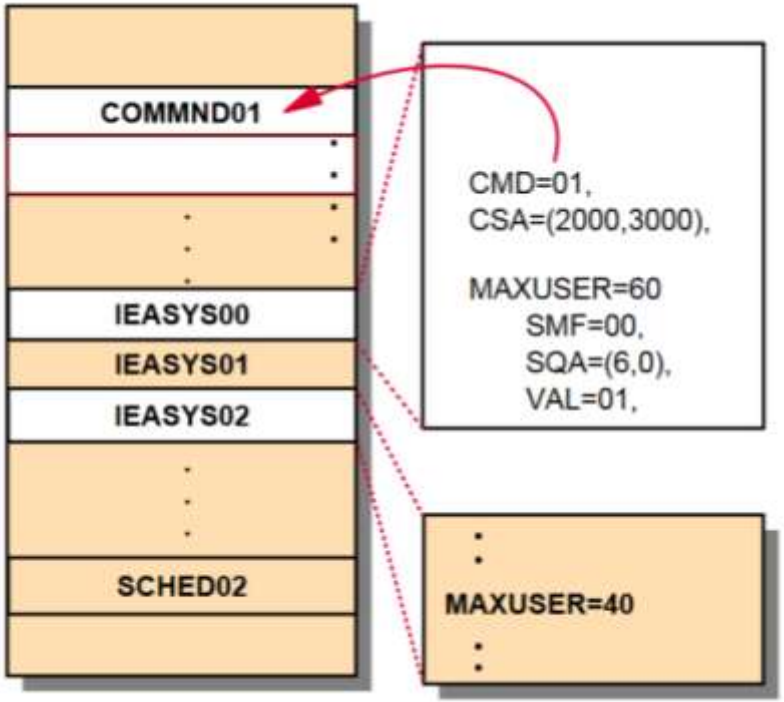
SHARING SYS1.PARMLIB



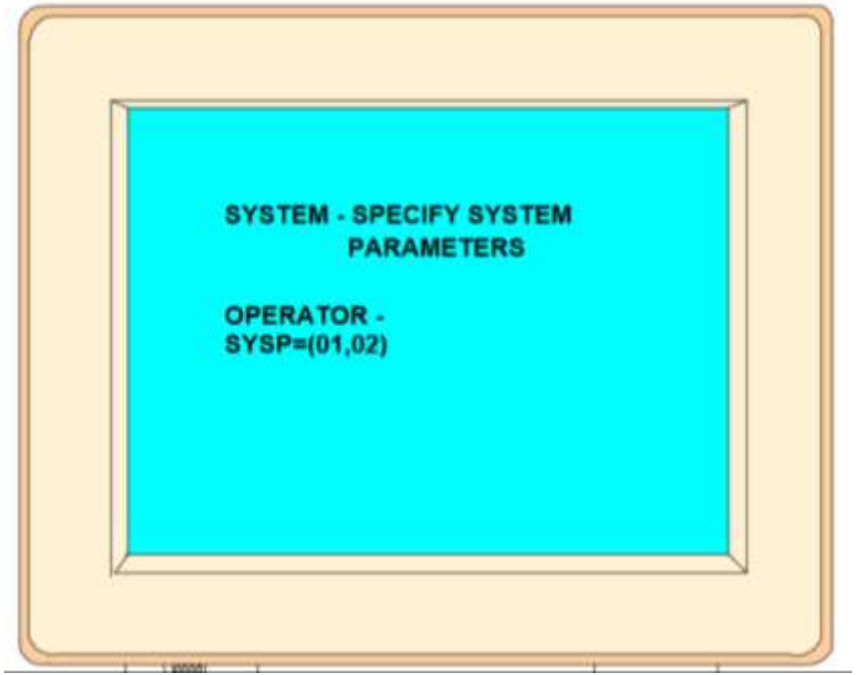
&SYSNAME SYMBOLIC



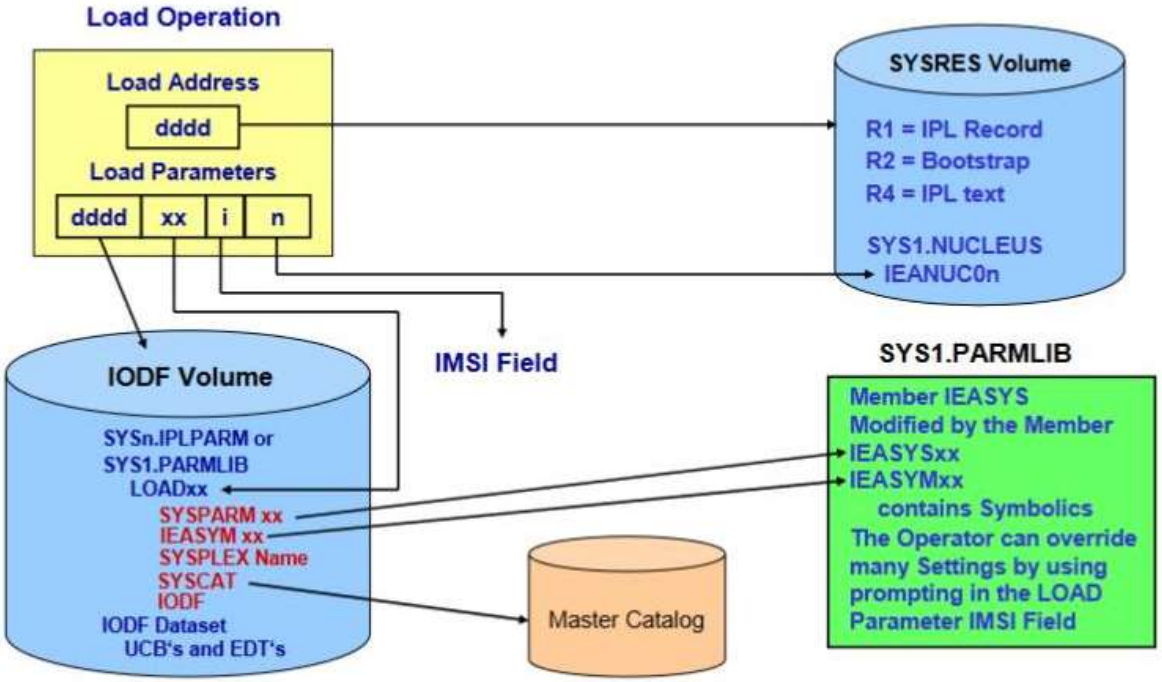
IEASYSnn


COMMND01
IEASYS00
IEASYS01
IEASYS02
SCHED02
CMD=01,
CSA=(2000,3000),
MAXUSER=60
SMF=00,
SQA=(6,0),
VAL=01,
MAXUSER=40

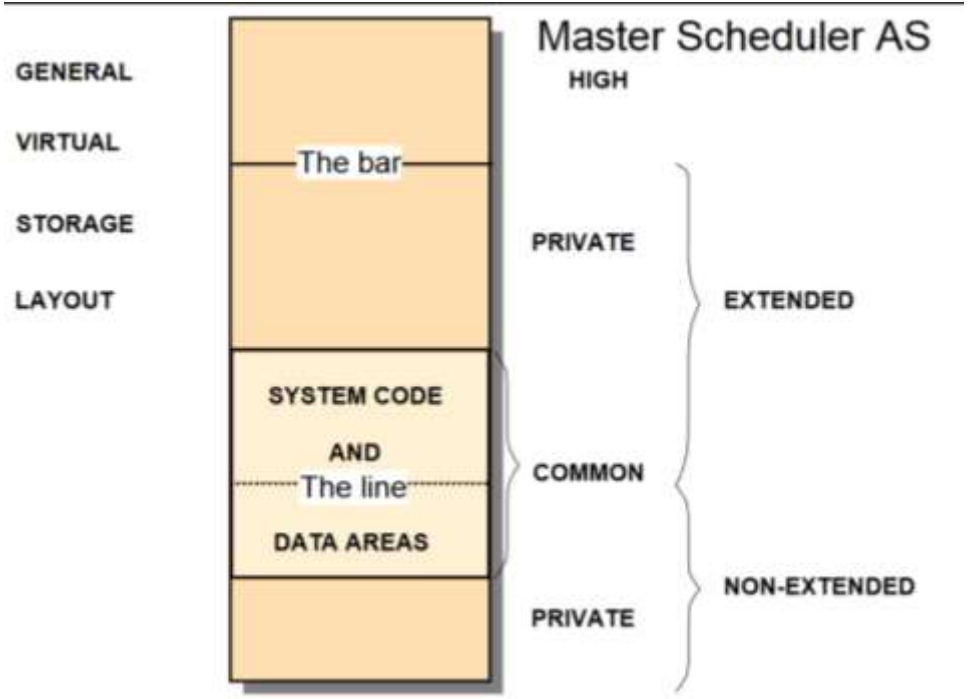
OPERATOR PROMPT (System Parameters)



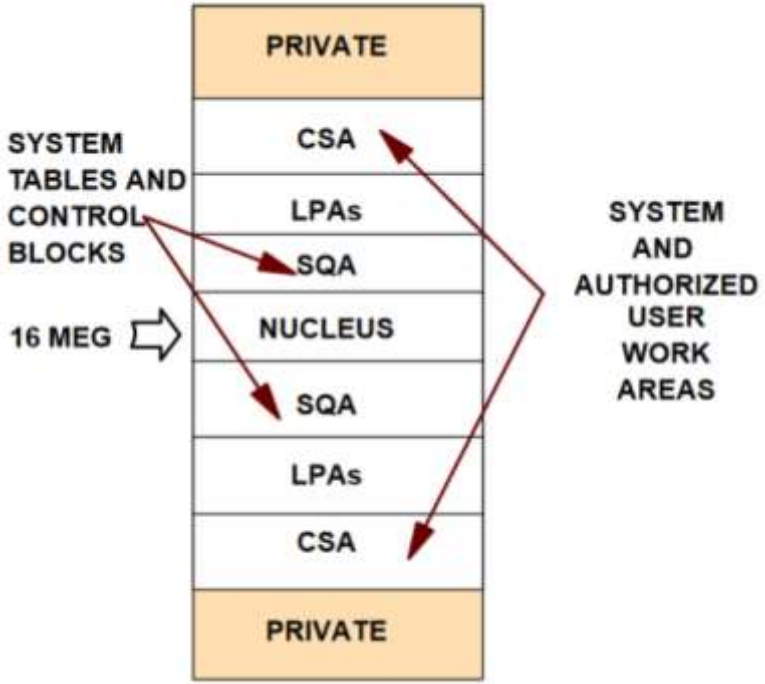
DATA SET CONSIDERATIONS – THE BIG PICTURE



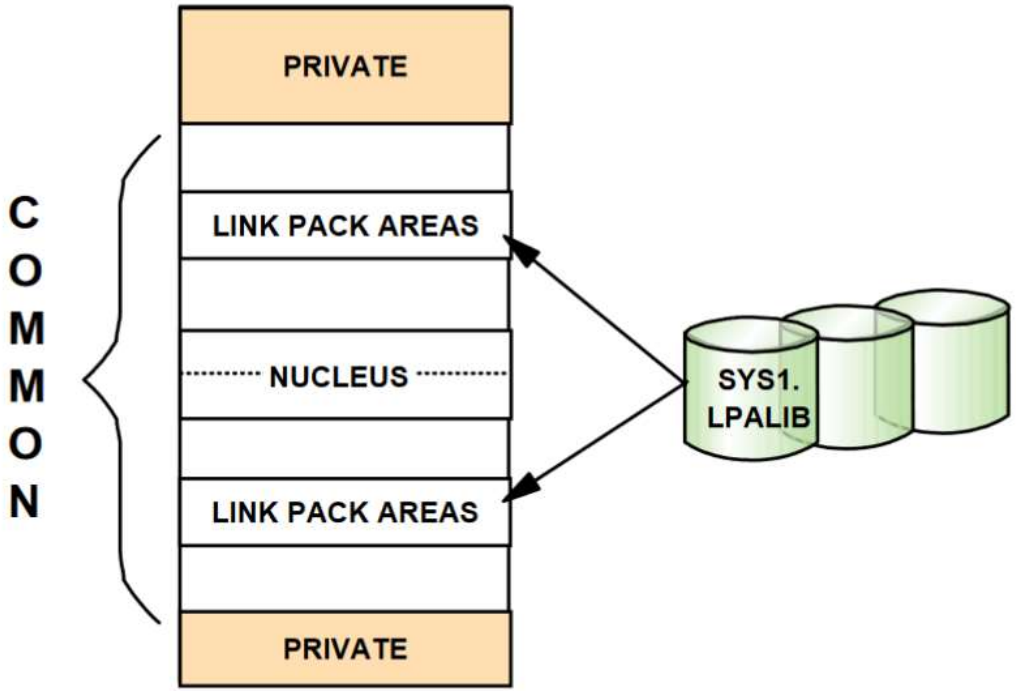
THE FIRST ADDRESS SPACE



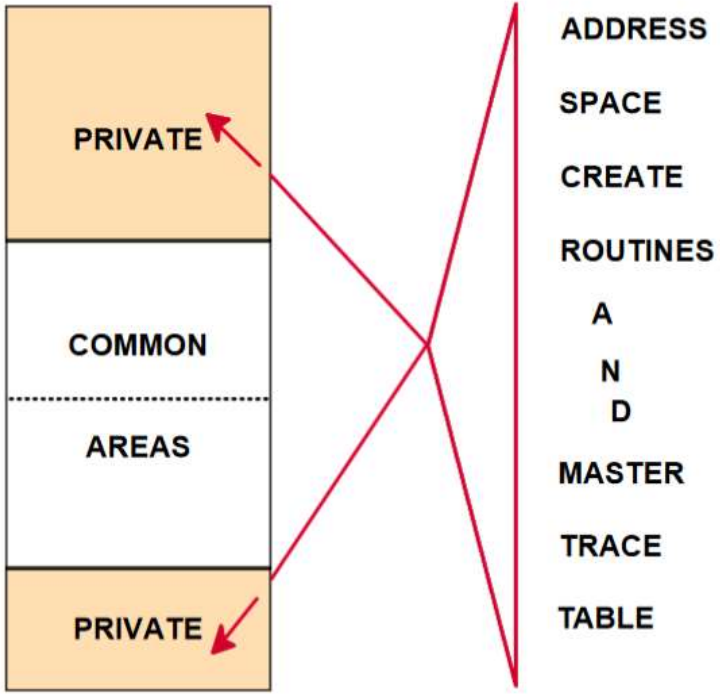
CSA AND SQA BOUNDARIES



THE LINK PACK AREA



MASTER SCHEDULER



MASTER SCHEDULER INITIALIZATION (MSI)

MSI PROCESSING DETAILS

1. Initialize MIH services.
2. Complete ASM initialization.
3. Initialize IOS dynamic pathing, create IOSAS.
4. Initialize Master's security environment.
5. Initialize Console ottributes, DEL=RD etc.
6. Initialize APPC services.
7. Initialize TSO services.
8. Initialize LOGREC Logstream recording.
9. Enable ENF services.
10. Initialize System Logger services, creates IXGLOG address space.
11. Vary all available CPs online.
    - we are now multiprocessing.
12. Initialize SMF services, creates SMF address space.

MIH: Missing Interrupt Handler (D IOS,MIH).
ASM: Auxiliary Storage Manager.
IOS: Input/Output Supervisor.
ENF: Event Notification Facility.
RTM: Recovery Termination Manager.

G&L GROUP

MSI PROCESSING DETAILS...

13. Issue commands in IEACMDOO and COMMNDxx parmlib members.
    - only commands processed by CONSOLE will execute now.
14. Initialize RTM services.
    - LOGREC recording.
    - Address space termination.
    - SVC dump processing.
15. Initialize System security processing.
16. Build defined subsystems.
    - Invoke initialization routine.
    - Issue START for primary JES subsystem, if requested.
17. Hold primary JES STC and TSO processing.
18. Indicate MSI is complete.
19. Initialize Master command processing.
    - Any pending commands that execute in Master will now be executed.
    - Start commands are executed by Master.

MSI PROCESSING DETAILS...

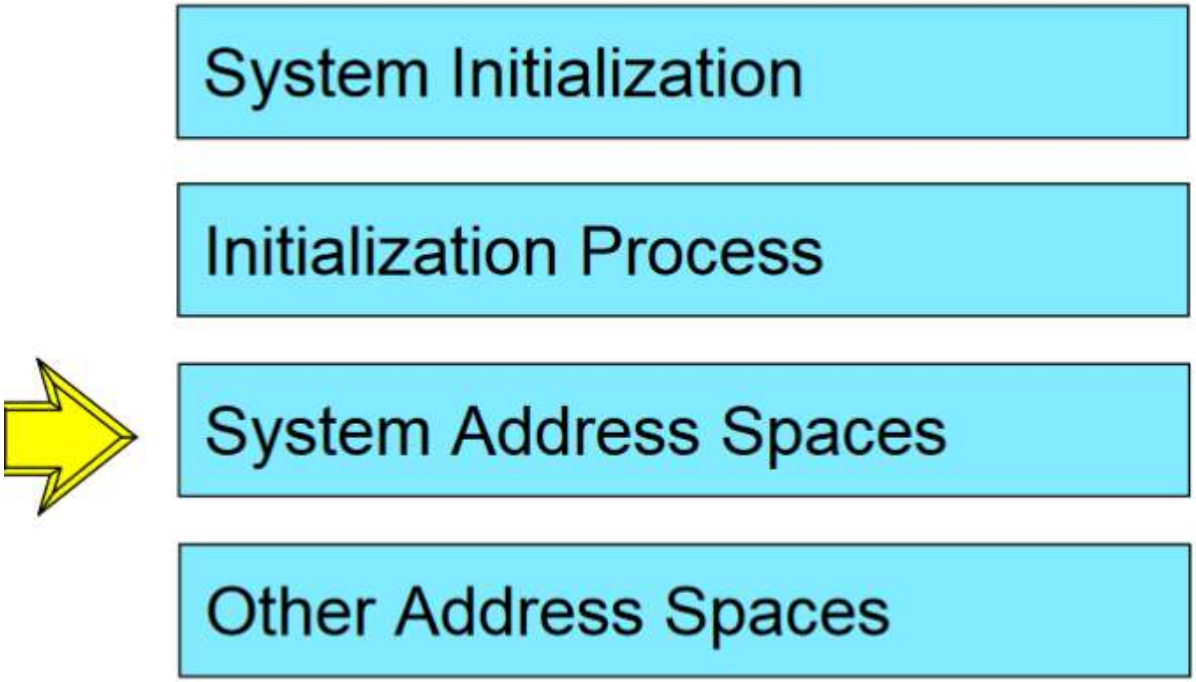
Full function address spaces can be created – JES and other tasks started under MSTR will now start.

20. Issue command processing available message.
21. Allow pending address space creates (not done by Master) to complete.
    - ° Create full function CATALOG.
    - ° Original CATALOG terminates.
    - ° Address spaces may switchover from limited to full function.
22. Wait for JES to indicate primary services are available.
    - ° Release primary JES STC and TSO processing.
    - ° Start the System Log Syslog / OPERLOG.

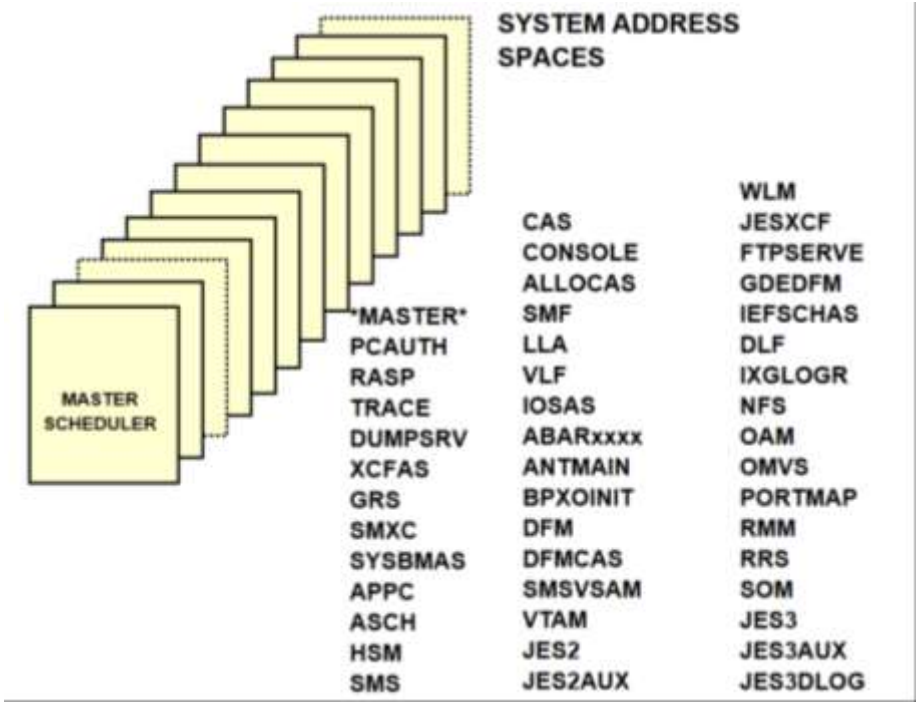
All IPL processing is now complete.

The next and final step is to bring up and initialize the job entry subsystem (JES2 or JES3).

SYSTEM ADDRESS SPACES



SYSTEM ADDRESS SPACES



SYSTEM ADDRESS SPACES

ALLOCAS: Allocation Address Space.
ANTAS000: XRC eXtended Remote Copy.
ANTMAIN: DFSMS support for concurrent/extended remote copy.
ASCH: APPC/MVS scheduling.
AXR: System-REXX.
BPXOINIT: z/OS UNIX System Services.
CATALOG: Access to catalogs.
CEA: Common Event Agent.
CONSOLE: Handling of operator command entry and systems response.
DEVMAN: Device Manager.
DLF: Data Lookaside Facility.
DUMPSRV: Dump Services für SVC dumps.

SYSTEM ADDRESS SPACES

EPWFFST: First Failure Support Technology (network).

GRS: Global Resource Serialization (data set level).

IEFSCHAS: Scheduler address space.

IOSAS: Input/Output Supervisor: System Interface to I/O devices.

IXGLOGR: System Logger.

JESXCF: Communication between JES2 systems in a MAS.

JES2: Job entry subsystem.

JES2AUX: JES2 additional support.

JES2MON: JES2 Monitor.

LLA: Library Lookaside.

OAM: Tape Library Data Servers.

SYSTEM ADDRESS SPACES

OMVS: z/OS UNIX System Services.
OTIS: OAM thread isolation support.
PCAUTH: Program Call Authorization.
RACF: Security Server.
RASP: Real Storage Manager System Adress Space.
RRS: Resource Recovery Services.
SMF: System Management Facility.
SMS: Storage Management SubSystem.
SMSPDSE: Handling of PDSE data set.
SMXC: DFSMS, control member access in PDSE.
SYSBMAS: Owner of dataspaces and hiperspaces for PDSE dir/mem.
TNF: Termination Notification Facility.

SYSTEM ADDRESS SPACES

TRACE: Detailed information on processors, SSCH, SVC, Program Check.

VLF: Virtual Lookaside Facility.

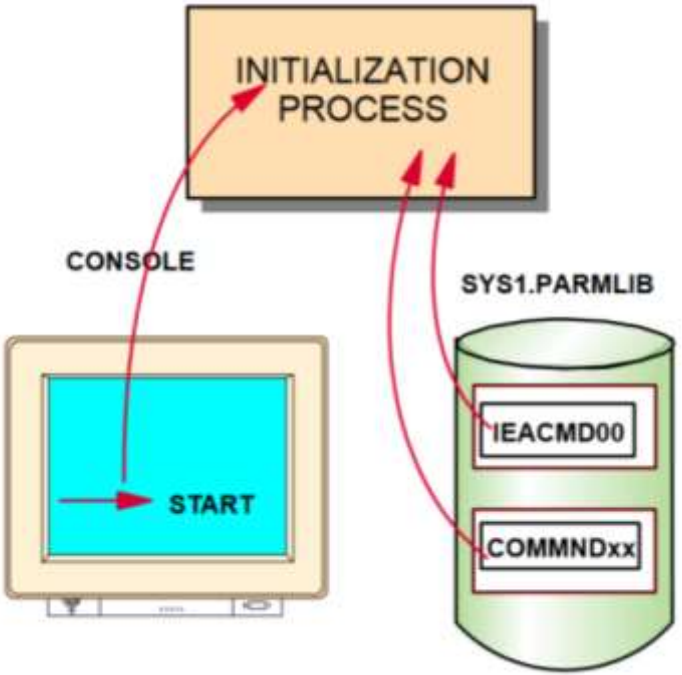
VMCF: Virtual Machine Communication Facility.

WLM: Workload Manager.

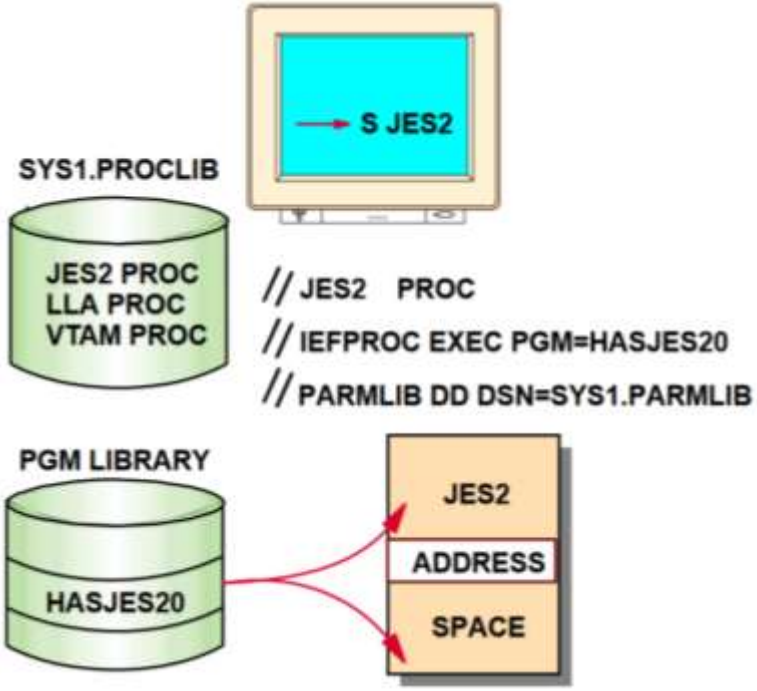
XCFAS: Cross System Coupling Facility Adress Space.

ZFS: z/OS FileSystem.

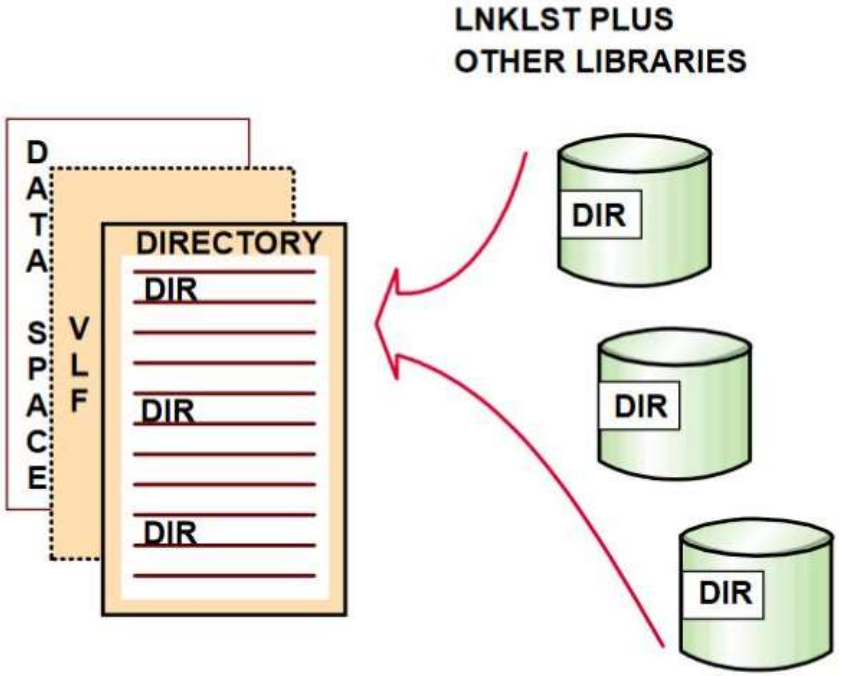
START COMMANDS



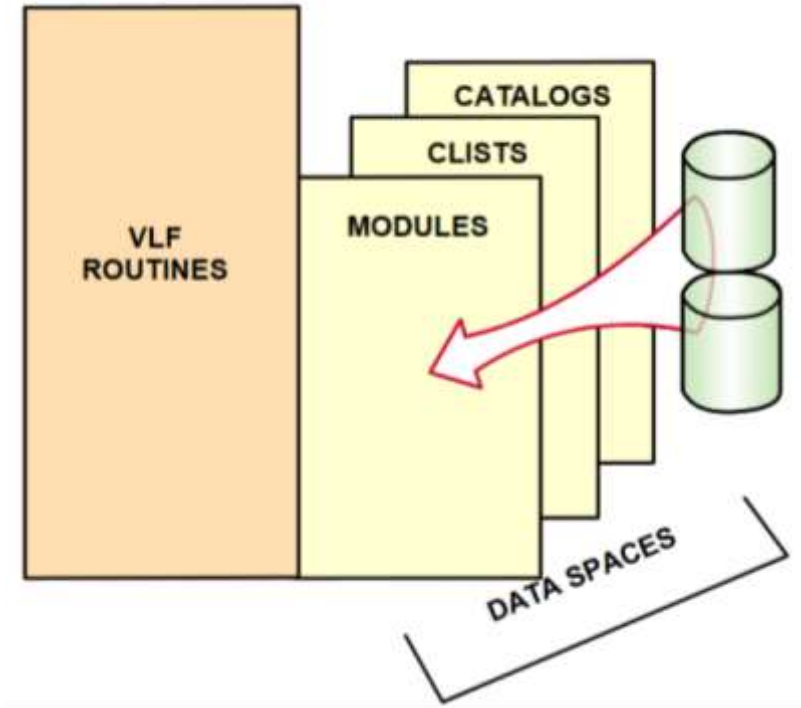
SYS1.PROCLIB



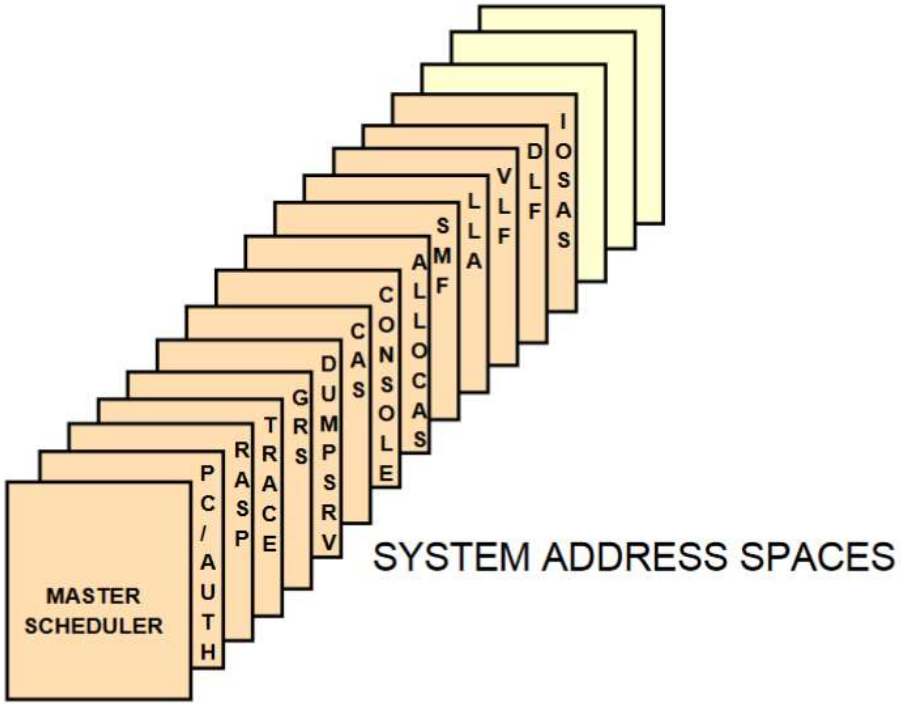
LIBRARY LOOKASIDE (LLA)



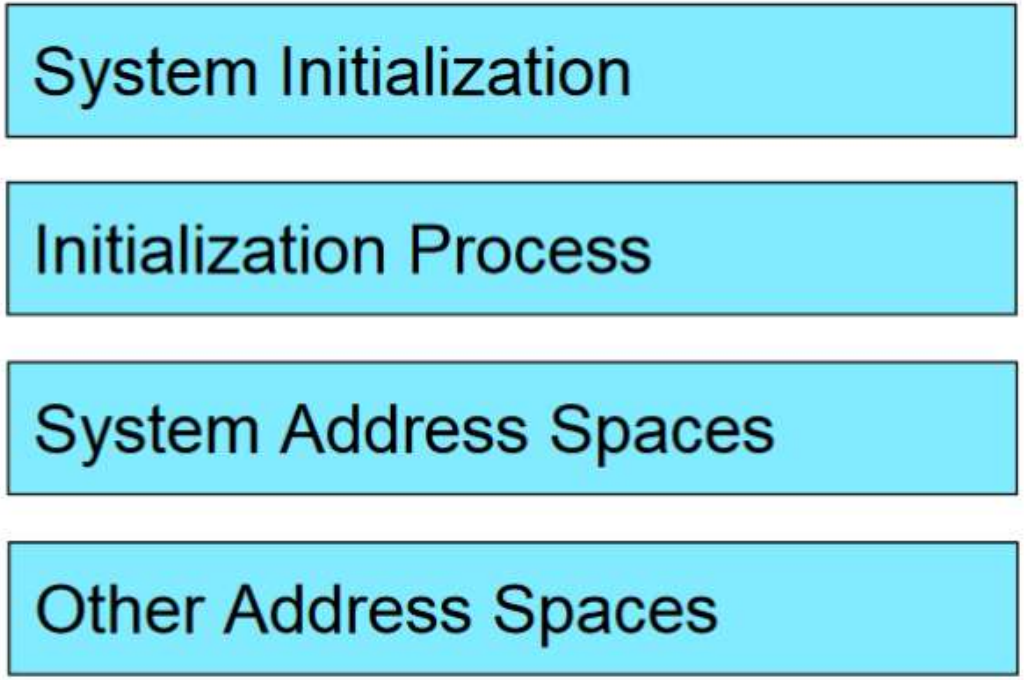
VIRTUAL LOOKASIDE FACILITY (VLF)



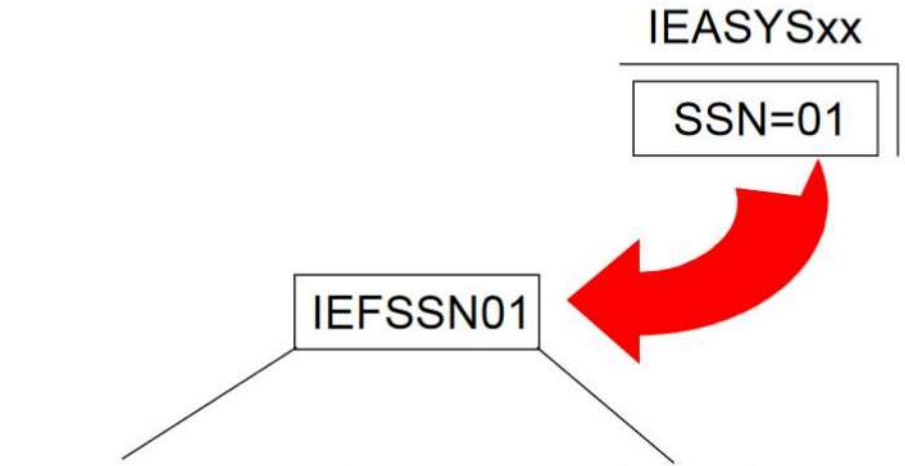
ADDITIONAL SYSTEM ADDRESS SPACES



OTHER ADDRESS SPACES

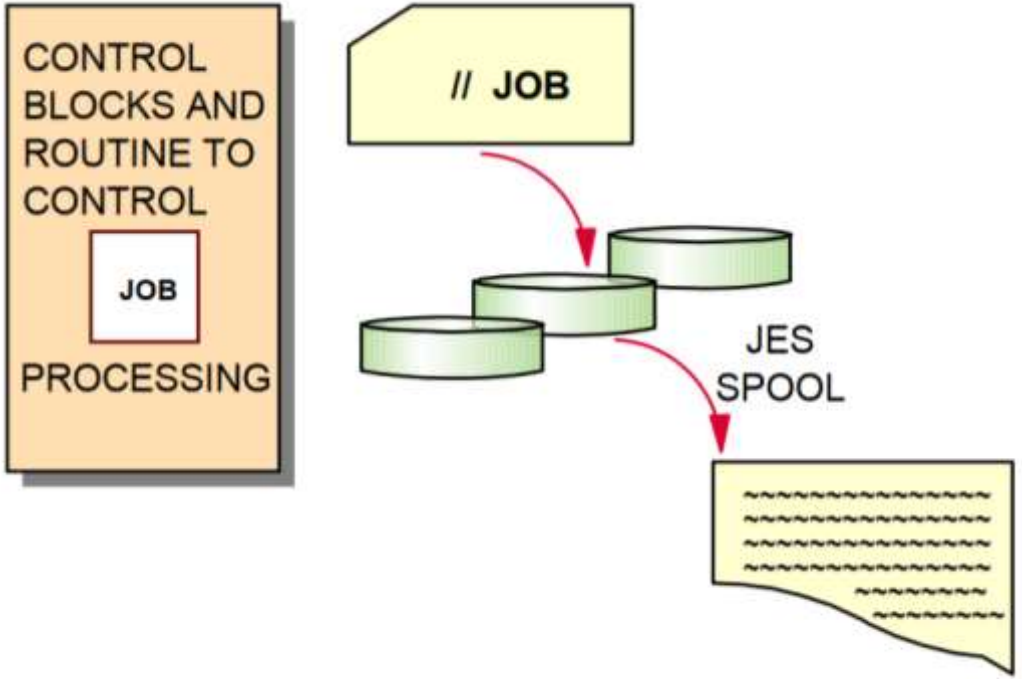


DEFINING SUBSYSTEMS

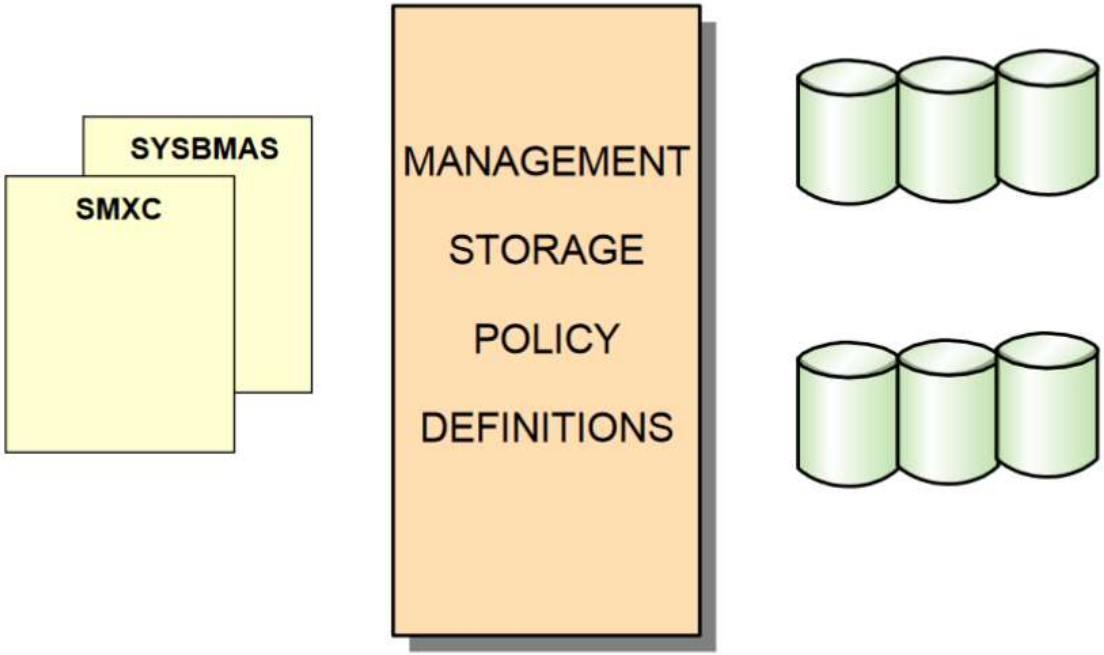



```
SUBSYS SUBNAME(SMS INITRTN(IGDSSIIN) . . .
SUBSYS SUBNAME(JES2) PRIMARY(YES)
START(NO)
SUBSYS SUBNAME(JESA)
SUBSYS SUBNAME(CICS)
```

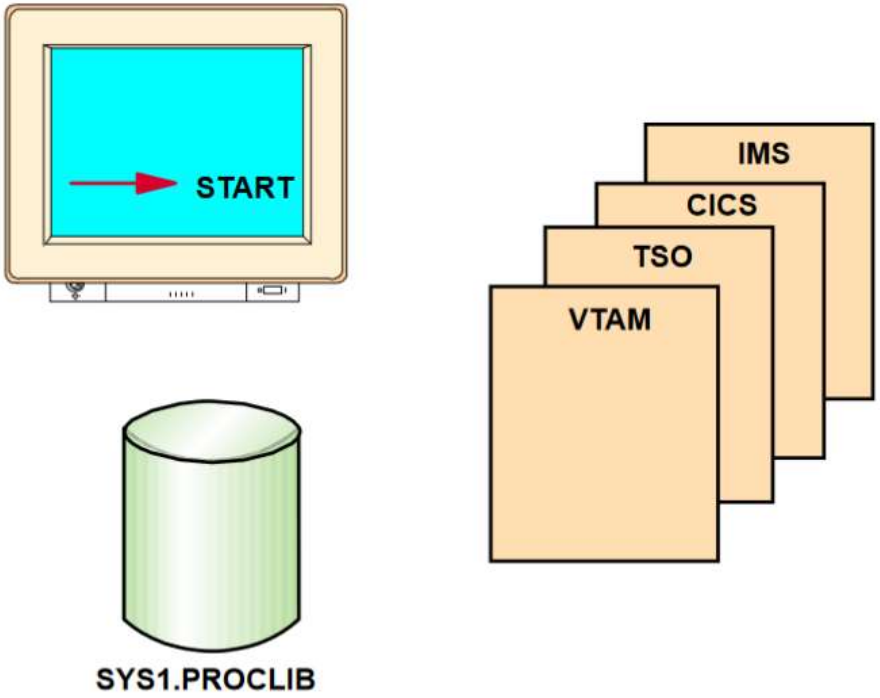
JOB ENTRY SUBSYSTEM (JES)



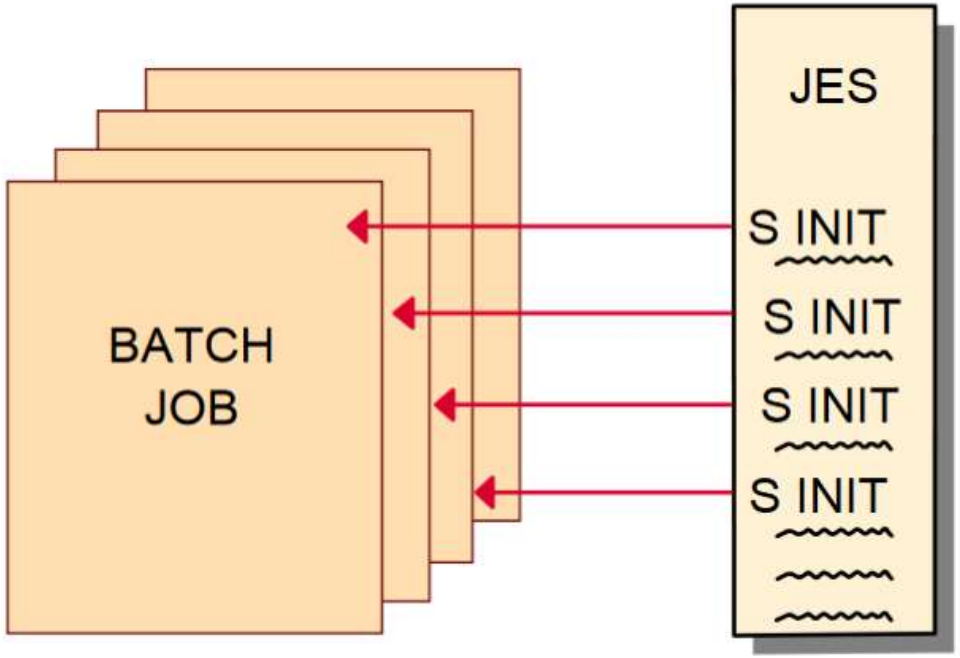
STORAGE MANAGEMENT SUBSYSTEM



STARTED TASK ADDRESS SPACES

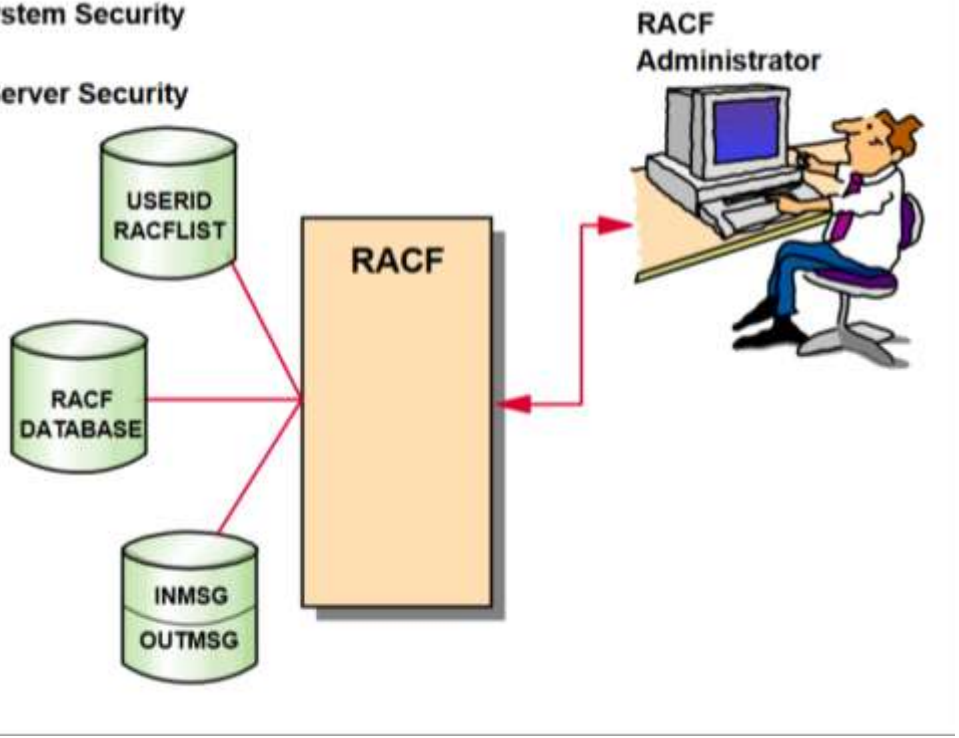


BATCH INITIATORS

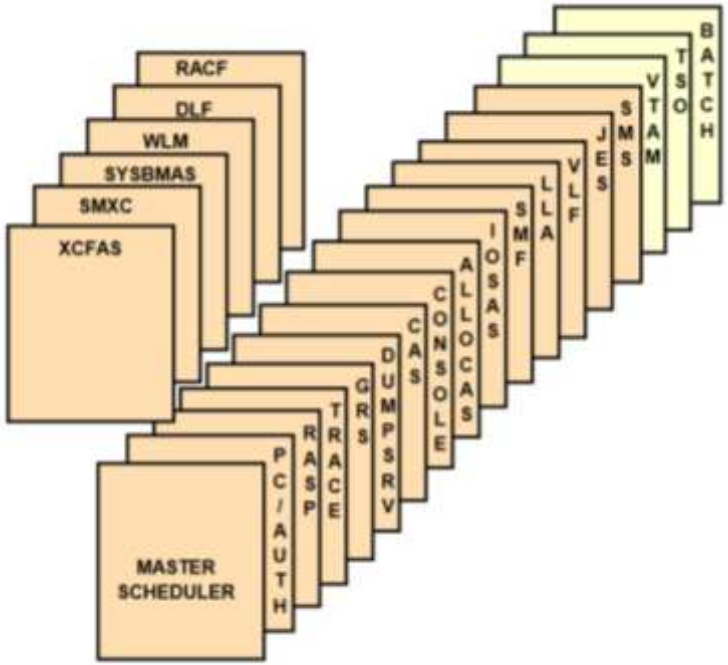


SECURITY SERVER RACF

- Host System Security
- Client/Server Security



OPERATIONAL z/OS



MIDDLEWARE FOR z/OS

° Middleware se llama al conjunto de programas que opera entre el sistema operativo y las aplicaciones.

° Generan plataformas que permiten desarrollar aplicaciones.

° z/OS middlewares típicos son:

- Database systems – DB2, IMS.
- Web servers – HTTP Web Server.
- Message queuing and routing functions – MQ Series.
- Transaction managers – CICS, IMS.
- Java virtual machines.
- XML processing functions.

MIDDLEWARE

El término generalmente se aplica a los principales productos de software, como bases de datos. administradores, server de transacciones, servidores web, etc.

El middleware típico de z/OS incluye:

- Sistemas de bases de datos.
- Servidores web.
- Funciones de enrutamiento y cola de mensajes.
- Gestores de transacciones (Transaction Server).
- Máquinas virtuales java.
- Funciones de procesamiento XML.

MIDDLEWARE

Un producto de middleware suele incluir una interfaz de programación de aplicaciones (API).

En algunos casos, las aplicaciones se escriben completamente en esta API de middleware, mientras que en otros se utiliza únicamente para fines únicos.

Algunos ejemplos de API de middleware de mainframe incluir:

El conjunto de productos WebSphere, que proporciona una API completa que es portable en múltiples sistemas operativos.

Entre ellos, WebSphere MQ proporciona una API que es multiplataforma.

El producto de gestión de bases de datos DB2, que proporciona una API (expresada en formato Lenguaje SQL) que se utiliza con muchos lenguajes y aplicaciones diferentes.

FIN.