z/OS MVS
System Initialization Logic
Initial Program Load (IPL)

Silvio Sasso
IBM Switzerland, Global Services ITS
sisa@ch.ibm.com
The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

<table>
<thead>
<tr>
<th>CICS*</th>
<th>IBM*</th>
<th>RACF*</th>
<th>z/OS*</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2*</td>
<td>IBM eServer</td>
<td>S/390*</td>
<td>z/VM*</td>
</tr>
<tr>
<td>DB2 Universal Database</td>
<td>IBM logo*</td>
<td>System z9</td>
<td>z/VSE</td>
</tr>
<tr>
<td>DirMaint</td>
<td>IMS</td>
<td>Tivoli*</td>
<td>zSeries*</td>
</tr>
<tr>
<td>ESCON*</td>
<td>NetView*</td>
<td>TotalStorage*</td>
<td>zSeries Entry License Charge</td>
</tr>
<tr>
<td>FICON*</td>
<td>OMEGAMON*</td>
<td>VSE/ESA</td>
<td></td>
</tr>
<tr>
<td>GDPS*</td>
<td>On Demand Business logo</td>
<td>VTAM*</td>
<td></td>
</tr>
<tr>
<td>HiperSockets</td>
<td>Parallel Sysplex*</td>
<td>WebSphere*</td>
<td></td>
</tr>
<tr>
<td>HyperSwap</td>
<td></td>
<td>z/Architecture</td>
<td></td>
</tr>
</tbody>
</table>

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft is a registered trademark of Microsoft Corporation in the United States and other countries.

* 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.
Abstract

- This presentation will describe the general processing which is involved in initializing a z/OS system, from the IPL process until the system is ready to start either JES2 or JES3

- The major steps described are:
  - The hardware process of loading z/OS
  - The loading and initialization of the nucleus
  - The initialization of general system resources
  - Master Scheduler Initialization

- IBM may change the implementation of internal processing at any time with no prior notice
z/OS Initialization Overview

- Processed in different phases
- Each phase builds on the next
- Within each phase, steps build on each other
**UCW and UCB Generation**

- **Activate using a Reset Profile (POR)**
  - **Central Storage**
    - **HSA**
      - UCWs (1 per Device)
    - **Main Storage**
      - UCBs (1 per Device)
  - **Dynamic Changes**
    - **HCD WORK IODF**
      - Allows adding, deleting or modifying the Hardware Definitions (CHPID, CU, Devices)
      - UCBs and EDTs
      - (requires a Production IODF to activate)
    - **HCD DYNAMIC**
      - Production IODF
      - Contains the LPAR Names, Channel Info, Control Unit, I/O Device Info for both UCWs and UCB’s, EDT and NIP consoles
    - **UCBs built during IPL/NIP**
      - IODF pointed to by the LOADPARMs and LOADxx

- **SE**
  - Selected IOCDS
  - UCW Images (subchannels)

- **IOCP Program**
  - Writes IOCDS

- **HCD**
  - Work IODF used to define Configuration and to make Changes
Dataset Considerations – the big Picture

Load Operation

Load Address

Load Parameters

ddddd

XX

I

NN

IODF Volume

SYS1.IPLPARM or
SYS1.PARMLIB

LOADxx

SYSPARM xx

IEASYM xx

SYSPLEX Name

SYSCAT

IODF

IODF Dataset

UCB’s and EDT’s

SYSRES Volume

R1 = IPL Record

R2 = Bootstrap

R4 = IPL text

SYS1.NUCLEUS

IEANUC0n

SYS1.PARMLIB

Member IEASYS

Modified by the Member

IEASYSxx

IEASYMxx

contains Symbolics

The Operator can override

many Settings by using

prompting in the LOAD

Parameter IMSI Field

IMSI Field

Master Catalog
Load Parameters

<table>
<thead>
<tr>
<th>IODF</th>
<th>LOADxx</th>
<th>IMSI</th>
<th>NUCx</th>
</tr>
</thead>
<tbody>
<tr>
<td>dddd</td>
<td>xx</td>
<td>i</td>
<td>nn</td>
</tr>
</tbody>
</table>

**DDDXXINN Load Parameter Values**

- **DDDD:** Device number of the volume containing the IODF dataset (Default is SYSRES)
- **XX:** ID of the LOADxx member to be used (the default is LOAD00)
- **I:** Initial Message Suppression Indicator (IMSI)
  - The default suppresses most informational messages and does not prompt for system parameters; will use the LOADxx values
- **NN:** Nucleus ID to be used (default is 1: IEANUC01)
## IMSI Character

<table>
<thead>
<tr>
<th>IMSI Character</th>
<th>Display informational Messages</th>
<th>Prompt for Master Catalog Response</th>
<th>Prompt for System Parameter Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period (.) or blank</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>A</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>C</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>D</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>M</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>P</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>S</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>T</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
And all begins with a Mouse Click...
Hardware IPL Overview

- Process is defined by the z/Architecture
- Controlled by hardware
- A single CPU is used for IPL - all other CPUs are placed into a manual (i.e. stopped) state
- A hardware system reset occurs before the process begins
- IPL records are written with ICKDSF
  - Cyl 0, Trk 0, R1, R2, IEAIPLO0
Hardware IPL Flow

Load Operation

1) CSS
   UCW for Load Device
   Hardware turns on enable bit

2) CSS sends IPL CCW to SYSRES
   DASD will seek to CYL 0, Head 0 and read Record 1

3) Record 1 will provide CCW's to read Record 2

4) Record 2 will provide CCW's to read Record 4

5) IEAIPL00 Location 0 contains initial PSW

After Record 4 has been read the Hardware Portion of IPL is complete
Hardware IPL Summary

- Hardware generates an IPL read of 24 bytes into location 0
  - For DASD, this always reads cylinder 0, track 0, record 1

- Location 8 treated as a command chained CCW
  - Read record 2 into storage, command chain to next CCW
  - Transfer CCW execution to record 2 location
  - Seek and search for IEAIPLOO record
  - Read IEAIPLOO into location 0

- CCW chain completion, PSW is loaded from absolute 0 and execution begun
  - IEAIPLOO location 0 contains initial PSW
IPL Resource Initialization

- **Overview**
  - Originally just loaded the Nucleus and set up the Master address space environment
    - Processing has gotten more complex with the XA architecture and Dynamic I/O support
  - Processing is single threaded
  - The IPL vector table (IVT) contains global information during this phase
- **IEAIPLOO**
  - A mini operating system - non relocatable
  - Builds an initial virtual environment
    - IPL workspace located at X'20000000' virtual
  - Provides services to
    - Back virtual storage with real frames
    - Do I/O
  - Controls the IPL initialization process
    - Loads IPL Resource Initialization Modules (RIMs) into workspace
    - Gives them control
IPL RIM Processing

1. **Test Block Instruction (clear Storage)**
2. **Read SCPINFO**
   - Get loadparm
   - Set autostore status on
3. **Locate usable real storage at top of memory**
4. **Get IPL load parameters, and set any defaults**
5. **Search LOADxx, process the information in LOADxx**
   
   IEA371I SYS0.IPLPARM ON DEVICE 5411 SELECTED FOR IPL PARAMETERS  first Message displayed on NIP Console
   IEA246I LOAD  ID 00 SELECTED

6. **Search IODF, process the information in the IODF**
   
   IEA246I NUCLST ID 00 SELECTED
   IEA519I IODF DSN = SYSIOD.IODF24
   IEA520I CONFIGURATION ID = SM15DPRI. IODF DEVICE NUMBER = 5411

   - **Build a table of NIP consoles**
     
     max. number of NIP consoles supported by IPL RIM is 64 (HCD supports 128)
     
     - see APAR OA12877 for additional information
IPL RIM Processing...

6. process the information in the IODF (cont.)
   - Invoke the device UIMs to
     - Identify device specific nucleus and LPA modules
     - Calculate required SQA and ESQA
     - Build device control blocks in the workspace
     - Build the Allocation EDT in the workspace

7. Create a map of the DAT-on nucleus CSECTs
   IEA091I NUCLEUS 1 SELECTED
   IEA093I MODULE IENUC01 CONTAINS UNRESOLVED WEAK EXTERNAL REFERENCE
   IFFIO
   IEA093I MODULE IENUC01 CONTAINS UNRESOLVED WEAK EXTERNAL REFERENCE
   IEDQATTN
   IEA093I MODULE IENUC01 CONTAINS UNRESOLVED WEAK EXTERNAL REFERENCE
   IECTATEN
   - Includes modules identified by NMLs, NUCLSTxx, and UIMs
   - CSECTs are grouped/positioned by attributes, RMODE and read-only

8. Load modules, dynamically resolving external references
IPL RIM Processing...

9. Create the initial SQA/ESQA areas
   - Sum of IBM supplied value, LOADxx INITSQA, UIM determined value

10. Create Master's VSM control blocks and LSQA

11. Create Master's permanent page and segment tables

12. Move from the workspace into SQA/ESQA
   - Device control blocks
   - Allocation EDT
   - IPL Messages
   - LPA device support module list

13. Validate real storage, build available frame queue
   - IPL workspace is destroyed

14. Load Prefix Register

15. Switch to nucleus version of the PSA
## Virtual Storage Layout

<table>
<thead>
<tr>
<th>Area</th>
<th>Description</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Private</td>
<td>High User Region</td>
<td>16Eb</td>
</tr>
<tr>
<td></td>
<td>Default shared Memory Addressing</td>
<td>512Tb</td>
</tr>
<tr>
<td>Low User Private</td>
<td>Low User Region</td>
<td>2Tb</td>
</tr>
<tr>
<td></td>
<td>Reserved</td>
<td>4Gb</td>
</tr>
<tr>
<td>Extended Private</td>
<td>Extended LSQA/SWA/229/230</td>
<td>2Gb</td>
</tr>
<tr>
<td></td>
<td>Extended User Region</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Extended CSA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Extended FLPA/MLPA/PLPA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Extended SQA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Extended Nucleus</td>
<td></td>
</tr>
<tr>
<td>Extended Common</td>
<td>Nucleus</td>
<td>16Mb</td>
</tr>
<tr>
<td></td>
<td>SQA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FLPA/MLPA/PLPA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LSQA/SWA/229/230</td>
<td></td>
</tr>
<tr>
<td>Private</td>
<td>User Region</td>
<td>8K</td>
</tr>
<tr>
<td></td>
<td>System Region</td>
<td>0</td>
</tr>
<tr>
<td>Common</td>
<td>PSA</td>
<td>24K</td>
</tr>
<tr>
<td></td>
<td>System Region</td>
<td>8K</td>
</tr>
<tr>
<td></td>
<td>PSA</td>
<td>0</td>
</tr>
</tbody>
</table>
**LOADxx Search Sequence**

1. **Search for the LOADxx member specified in the LOADPARM field, digits 5 and 6 (example Load Parm = 012355M)**

2. Is **SYSn.IPLPARM on the IODF volume?** (n=0-9)
   - **No**
   - **Yes**

3. Is **LOADxx in SYSn.IPLPARM?**
   - **No**
   - **Yes**

4. **LOADxx not found, enter non-restartable disabled Wait State WSC=088 RC=00; Re-IPL required**

5. **Is SYS1.PARMLIB on the IODF volume?**
   - **No**
   - **Yes**

6. Is **LOADxx in SYS1.PARMLIB**
   - **No**
   - **Yes**

7. **LOADxx not found, enter non-restartable disabled Wait State WSC=088 RC=00; Re-IPL required**

8. **Is SYS1.PARMLIB on the SYSRES volume?**
   - **No**
   - **Yes**

9. **LOADxx not found, enter non-restartable disabled Wait State WSC=0B1 RC=01; Re-IPL required**

10. **LOADxx found, continue IPL/NIP processing**
Overview

- Initializes basic system resources
- Processing is multithreaded - normal dispatching of work is done
- Basic system service (SRBs, WAIT, POST, EXCP, ATTACH, etc.) are initially available
- Additional services enabled as NIP RIMs run
- The NIP vector table (NVT) contains global information during this phase

Control routine

- Sets traps for unexpected errors (no RTM support is available yet)
- Verifies the hardware environment
- Creates IPL processor control blocks
- Creates global VSM control blocks
- Creates I/O control block pools
- Creates the initial system trace table
- Opens SYS1.NUCLEUS as the LNKLST
- Loads and invokes NIP RIM routines
In order for MVS to use a device:
- a UCW for the device must exist
- a UCB for the device must exist

During device mapping:
- each matching UCW is enabled
- each matching UCB is connected

During the mapping process, the I/O configuration (UCWs) loaded into the HSA with a POR (or updated via dynamic I/O) is matched with the operating system configuration (UCBs) defined in the IODF.

The UCWs are placed in the disabled state after POR or system reset.

Initial UCB state:
- the UCBs are built with the “not connected“ state bit = 1 (UCB byte 7, bit 2)
- at the completion of this mapping process all devices defined to both the channel subsystem (UCWs) and MVS (UCBs) will be enabled and connected
  - any UCWs without corresponding UCBs will be left disabled
  - any UCBs without corresponding UCWs will be left not connected

Devices in either one of these states cannot be used by the system.
Non-DASD Pathing

- The process of determining path availability is referred to as Pathing
  - during this process MVS will check all paths for devices generated to come up online by attempting to complete an I/O operation down each path defined to a device
  - if at least one path is operational the device will be online
  - Tapes are an exception: pathing is performed to offline tape devices

_MVS does not report any paths or devices that are found to be not operational during pathing_
DASD Pathing

- A NIP console is required before DASD pathing takes place to allow the operator to respond to out-of-line conditions encountered during the DASD pathing
  - Issue SSCH to multiple devices (test multiple devices in parallel)
  - After each successful I/O another device is tested
  - Redrive another device if an I/O is complete for a device
    - If an I/O fails to complete within 15 seconds, the I/O operation is purged
  - Perform path testing on each path
    - no 1.5 sec. timeout (no IOS120A message during path testing)
    - create PIM (Path Installed Mask), represents CHPID's defined in IOCDS
    - create LPM (Logical Path Mask), used by UCB to control paths to be used for an I/O operation
  - Get device characteristics - one path
  - Self description - each path (msg IOS291I)
  - VOLSER checking - one path for SDP products (all paths for other DASD)
    - duplicate VOLSER message (IEA213A - not SYSRES, IEA214A SYSRES)
  - at end of pathing wait 15 seconds for any outstanding I/O to complete
    - mark any UCB with outstanding I/O to test later again
    - purge all outstanding requests
### DASD Pathing...

<table>
<thead>
<tr>
<th>1 (P)</th>
<th>2 (D)</th>
<th>3 (S)</th>
<th>4 (V)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path Testing</td>
<td>UCB Device Characteristics Initialization</td>
<td>SDP</td>
<td>VOLSER</td>
</tr>
<tr>
<td>Each Path</td>
<td>One Path</td>
<td>Each Path</td>
<td>One Path SDP Device Each Paths non-SDP device</td>
</tr>
<tr>
<td>CCW: 94 Release</td>
<td>CCWs: E4 Sense Id; 64 RDC; 54 Subsystem; FA RCD</td>
<td>CCWs: E4 Sense Id (one Path) FA RCD (each Path)</td>
<td></td>
</tr>
<tr>
<td>Messages: IGGN504A; IGGN505A; Required Dataset missing; IOS120A moved to MSI</td>
<td>Message: IEC334I (duplicate SSID)</td>
<td>Message: IOS291I (Configuration Mismatch)</td>
<td>Messages: IEA213A; IEA214A (Duplicate VOLSER)</td>
</tr>
</tbody>
</table>

DASD pathing consists of 4 different phases: path testing on each path (P), read device characteristics (D), self-describing product (S) and VOLSER processing (V).

Any error conditions detected during the DASD pathing steps are reported to the NIP console via messages IGGN504A, IGGN505A, IEC334I, IOS291I, IEA213A or IEA214A (*any A or action messages requires operator response*).

- **CCW** = Channel Command Word
- **SDP** = Self-describing Product
- **RCD** = Read Configuration Data
- **SSID** = Subsystem ID (DASD CUs)
- **RDC** = Read Device Characteristics
DASD Pathing...

- Dynamic Pathing Error Messages

  IOS291 CONFIGURATION DATA COULD NOT BE READ ON PATH (24C0,49) RC=21

  - IOS291I messages with a RC of 21, 27 or 29 indicate a possible configuration error and should be investigated

  IEC334I DUPLICATE SUBSYSTEM X`0001`,CCA X`00`, DEVIVE 24C0 NOT BROUGHT ONLINE

  - In addition the the IOS291I messages, a misconfiguration problem to a DASD CU may also show up as a duplicate SSID condition

  IEA213A DUPLICATE VOLUME `SPOOL1` FOUND ON DEVICES 2465 AND 28A0
  IEA213A REPLY DEVICE NUMBER WHICH IS TO REMAIN OFFLINE

  IEA214A DUPLICATE SYSRES `S15R21` FOUND ON DEVICE 22C4
  IEA214A VERIFY THAT THE CORRECT DEVICE WAS USED FOR IPL
  IEA214A DUPLICATE DEVICE WILL REMAIN OFFLINE
  IEA214A REPLY `CONT` TO CONTINUE IPL

  - The last step of dynamic pathing is Direct Access Volume Verification (DAVV)

  - DAVV processing reads the volume label of each online DASD device and updates the UCB with the VOLSER

  - If a duplicate VOLSER exists, either message IEA213A or IEA214A will be issued
DASD Pathing...

- Dynamic Pathing Error Messages...

  IGGN505A SPECIFY UNIT FOR SYS1.PRODXY.LINKLIB ON DCSYS2 OR CANCEL
  
  R 00,1A60
  IEE600I REPLY TO 00 IS;1A60

  · If the busy condition still exists IOS120A will be issued

  *IOS120A DEVICE 1A60 SHARED, REPLY 'CONT' OR 'WAIT'
  IOS600I REPLY TO 00 IS 'WAIT'
  *IOS124A STILL WAITING FOR RESPONSE FROM DEVICE 1A60, TOTAL WAIT TIME
   IS 46 SECONDS, REPLY 'CONT' OR 'WAIT'

  *IOS120A DEVICE 1A60 SHARED, REPLY 'CONT' OR 'WAIT'
  IOS600I REPLY TO 00 IS 'WAIT'
  IGGN306I 1A60,UNIT UNACCEPTABLE, 00000004
  IGGN505A SPECIFY UNIT FOR SYS1.PRODXY.LINKLIB ON DCSYS2 OR CANCEL

  · IGGN504A or IGGN505A message issued if required dataset is on a volume that was busy during DASD
   pathing and the dataset is required for the IPL to complete

  · Issue D U,VOL=vvvvv on an active system that shares the DASD device to obtain the device number
   associated with the VOLSER
NIP RIM Processing

1. Create RTM recovery and control blocks
2. Create WTO control blocks and pools
   - WTOs issued now will be logged in SYSLOG
3. Initialize Machine Check handling (MCH)
4. Device mapping (UCWs to UCBs), test availability, and initialize non-DASD devices
5. Select and initialize NIP
   - WTOs will now be displayed on the NIP console
6. Test availability, and initialize DASD devices (DASD Pathing)
   - Operator can be prompted during validation
7. Open the master catalog
8. Create the system symbolics from IEASYMxx
9. Open SVCLIB, PARMLIB, and LOGREC
10. If required, prompt for system parameters (message IEA101A)
11. Merge and analyze the system parameters
NIP RIM Processing...

12. Initialize ASM, opening page and swap datasets
13. Process SQA= parameter
   ▪ On a quickstart (CLPA not specified), PLPA boundaries control SQA/ESQA boundaries
   ▪ On a coldstart, expand initial SQA/ESQA
14. Create user SVC table entries from IEASVCxx
15. Create the PLPA if CLPA specified
   ▪ LPALSTxx datasets
   ▪ UIM specified device support from SYS1.NUCLEUS
16. Create FLPA and MLPA, fix FLPA area and protect both areas as requested
17. Complete type 3 and 4 SVC table entries
18. Process CSA= parameter
19. Initialize system resource manager (SRM)
20. Enable RTM for task termination / SRB purge
   ▪ Limited Function Address spaces can now be created by master scheduler
21. Initialize Cross-memory services, creates PCAUTH address space
NIP RIM Processing...

22. Initialize RSM Dataspaces services, creates RASP
23. Initialize System Trace services, creates TRACE
24. Initialize Timing services, sets TOD if needed
25. Initialize SVC dump services, creates DUMPSRV address space
26. Initialize XCF/XES services, creates XCFAS address space
27. Initialize GRS services, creates GRS address space
28. Initialize SMS and PDSE services, creates SMXC and SYSBMAS address spaces
29. Open LNKLST -- drops SYS1.NUCLEUS
30. Initialize Console services, creates CONSOLE address space
   • Full function console is still unavailable
31. Initialize WLM services, creates WLM address space
32. Initialize data management
33. Initialize Concurrent-copy, creates ANTMAIN and ANTAS000 address spaces
34. Initialize UNIX System Services, creates OMVS address space
NIP RIM Processing...

35. Close master catalog

36. Initialize Catalog services, creates CATALOG address space
   - Limited function, for use until MSI completes

37. Exit NIP processing
   - Create the IPL parameter area (IPA)
   - Free control blocks no longer needed by NIP
   - Reset traps for unexpected errors, enables full RTM recovery/retry
   - LINK to Master Scheduler processing
- **Master Scheduler Initialization (MSI) Overview**
  - Completes initialization of system functions
  - Coordinates final completion with primary subsystem (JES2/JES3)

- **Basic Processing**
  - Initialize Master Trace processing
  - Enable full function Console processing
    - All MCS consoles are now available
  - Initialize Sysplex-wide ENF services, creates IEFSCHAS address space
  - Initialize MSTR subsystem
  - Initialize Common JES services, creates JESXCF address space
  - Initialize Allocation services, creates ALLOCAS address space
  - Attach Initiator to start Master JCL
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 attributes, 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
MSI Processing Details...

13. Issue commands in IEACMD00 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)
IPCS Display IPL Statistic

VERBX BLSAIPST

*** IPL Statistics ***

| IEAIPL10  | 00:00:00.000 | ISNIRIM - Read SCPINFO |
| IEAIPL20  | 00:00:12.57 | Test Block storage to 2G |
| IEAIPL11  | 00:00:00.009 | Fast FIND service |
| IEAIPL31  | 00:00:00.001 | LOAD service |
| IEAIPL30  | 00:00:00.007 | IPLWTO service |
| IEAIPL46  | 00:00:00.166 | Read SCHIBs into IPL workspace |
| IEAIPL49  | 00:00:00.000 | Process Load and Default parameters |
| IEAIPL50  | 00:00:01.784 | IPL parmlib - process LOADxx and NUCLSTxx |
| IEAIPL51  | 00:00:00.12 | System architecture |
| IEAIPL43  | 00:00:00.010 | Find and Open IODF data set |
| IEAIPL60  | 00:00:00.000 | Read NCRs from IODF |
| IEAIPL70  | 00:00:01.58 | UIM environment - load CBD and IOS services |
| IEAIPL71  | 00:00:01.99 | Build DFT for each device |
| IEAIPL08  | 00:00:00.29 | Read EDT information from IODF |
| IEAIPL40  | 00:00:00.107 | Read MLTs from nucleus |
| IEAIPL42  | 00:00:00.006 | Read NMLs from nucleus (IEANynnn modules) |
| IEAIPL41  | 00:00:01.662 | Read PDS directory entries and CESD records |
| IEAIPL05  | 00:00:00.595 | Build and sort NUCMAP |
| IEAIPL02  | 00:00:06.357 | Load nucleus modules |
| IEAIPL04  | 00:00:00.14 | Allocate PFT and SQA/ESQA |
| IEAIPL14  | 00:00:00.000 | Build LSQA/ELSQA for Master |
| IEAIPL06  | 00:00:00.000 | IARMI - RSM blocks, master SGT |
| IEAIPL09  | 00:00:00.47 | IAXMI - PFT, master RAB, etc. |
| IEAIPL07  | 00:00:00.19 | Update AMODE for nucleus resident SVCs |
| IEAIPL03  | 00:00:00.52 | Build UCBs, ULUT, etc. |
| IEAIPL18  | 00:00:00.093 | Copy and relocate EDT to ESQA |
| IEAIPL99  | 00:00:00.317 | Page frame table and cleanup |

Total IPL Time: 00:00:12.914
### IPCS Display IPL Statistic...

#### *** NIP Statistics ***

<table>
<thead>
<tr>
<th>Code</th>
<th>Time (sec)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEAVNIP0</td>
<td>00:00:00.071</td>
<td>NIP Base</td>
</tr>
<tr>
<td>IEAVNIPM</td>
<td>00:00:00.114</td>
<td>Invoke NIP RIMs</td>
</tr>
<tr>
<td>IEAVNPE6</td>
<td>00:00:00.099</td>
<td>Service Processor Interface</td>
</tr>
<tr>
<td>IEAVNPFF</td>
<td>00:00:00.056</td>
<td>Loadwait/Restart</td>
</tr>
<tr>
<td>IEAVNPA6</td>
<td>00:00:00.032</td>
<td>RTM - RTCT and recording buffer</td>
</tr>
<tr>
<td>IEAVNPC6</td>
<td>00:00:00.018</td>
<td>WTO</td>
</tr>
<tr>
<td>IEAVNPC3</td>
<td>00:00:00.025</td>
<td>Issue messages from IPL message queue</td>
</tr>
<tr>
<td>IEAVNP06</td>
<td>00:00:00.034</td>
<td>SMS Open/Mount</td>
</tr>
<tr>
<td>IEAVNP27</td>
<td>00:00:00.029</td>
<td>Reconfiguration</td>
</tr>
<tr>
<td>IEAVNPA2</td>
<td>00:01:25.428</td>
<td>IOS - Non-DASD UCBs</td>
</tr>
<tr>
<td>IEAVNPCA</td>
<td>00:00:00.012</td>
<td>NIP Console</td>
</tr>
<tr>
<td>IEAVNPB2</td>
<td>00:00:08.569</td>
<td>IOS - DASD UCBs</td>
</tr>
<tr>
<td>IEAVNP11</td>
<td>00:00:00.032</td>
<td>Locate and Open master catalog</td>
</tr>
<tr>
<td>IEAVNPC7</td>
<td>00:00:00.008</td>
<td>Open SYS1.SVCLIB</td>
</tr>
<tr>
<td>IEAVNPOP</td>
<td>00:00:00.054</td>
<td>Open PARMLIB</td>
</tr>
<tr>
<td>IEAVNPL</td>
<td>00:00:00.179</td>
<td>Process IEALSTxx</td>
</tr>
<tr>
<td>IEAVNPC4</td>
<td>00:00:00.015</td>
<td>Prompt for System Parameters</td>
</tr>
<tr>
<td>IEAVNPO3</td>
<td>00:00:00.009</td>
<td>Merge and analyze system parameters</td>
</tr>
<tr>
<td>IEAVNPCF</td>
<td>00:00:04.189</td>
<td>Process system name and system variables</td>
</tr>
<tr>
<td>IEAVNP76</td>
<td>00:00:00.022</td>
<td>Open LOGREC</td>
</tr>
<tr>
<td>IEAVNPE8</td>
<td>00:00:00.039</td>
<td>RSM - Process REAL=</td>
</tr>
<tr>
<td>IEAVNP23</td>
<td>00:00:00.037</td>
<td>Build GRS blocks in SQA</td>
</tr>
<tr>
<td>IEAVNP04</td>
<td>00:00:00.102</td>
<td>ASM - Open page and swap data sets</td>
</tr>
<tr>
<td>IEAVNPA8</td>
<td>00:00:00.012</td>
<td>VSM - Expand SQA</td>
</tr>
<tr>
<td>IEAVNPC2</td>
<td>00:00:00.057</td>
<td>IOS - Move CDT to SQA</td>
</tr>
<tr>
<td>IEAVNP14</td>
<td>00:00:02.276</td>
<td>ASM part 2 - Build SQA control blocks</td>
</tr>
<tr>
<td>IEAVNPGD</td>
<td>00:00:00.004</td>
<td>Move console data to ESQA</td>
</tr>
<tr>
<td>IEAVNP25</td>
<td>00:00:00.033</td>
<td>Process SVC=</td>
</tr>
<tr>
<td>IEAVNP05</td>
<td>00:00:16.493</td>
<td>LPA, APF</td>
</tr>
<tr>
<td>IEAVNP44</td>
<td>00:00:00.003</td>
<td>ASA Reuse stuff</td>
</tr>
<tr>
<td>IEAVNPB1</td>
<td>00:00:00.002</td>
<td>Process CSCBLOC=</td>
</tr>
</tbody>
</table>
### IPCS Display IPL Statistic...

<table>
<thead>
<tr>
<th>Job</th>
<th>Time</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEAVNPBE</td>
<td>00:00:00.005</td>
<td>RACF SAF</td>
</tr>
<tr>
<td>IEAVNPB8</td>
<td>00:00:00.019</td>
<td>Create CSA</td>
</tr>
<tr>
<td>IEAVNPP47</td>
<td>00:00:00.003</td>
<td>ENF</td>
</tr>
<tr>
<td>IEAVNPD6</td>
<td>00:00:00.002</td>
<td>RTM - SDUMP, ABDUMP, ESTAE</td>
</tr>
<tr>
<td>IEAVNPD8</td>
<td>00:00:04.891</td>
<td>RSM - Frame queues, VRREGN= and RSU=</td>
</tr>
<tr>
<td>IEAVNP010</td>
<td>00:00:00.015</td>
<td>SRM - OPT=, IPS=, etc.</td>
</tr>
<tr>
<td>IEAVNP1</td>
<td>00:00:00.023</td>
<td>ABDUMP</td>
</tr>
<tr>
<td>IEAVNPD2</td>
<td>00:00:00.030</td>
<td>SDUMP</td>
</tr>
<tr>
<td>IEAVNPCX</td>
<td>00:00:00.003</td>
<td>Context services, registration services</td>
</tr>
<tr>
<td>IEAVNPX1</td>
<td>00:00:00.021</td>
<td>NIP cleanup</td>
</tr>
<tr>
<td>IEAVNPB5</td>
<td>00:00:00.156</td>
<td>PCAUTH</td>
</tr>
<tr>
<td>IEAVNPB8</td>
<td>00:00:00.167</td>
<td>RASP</td>
</tr>
<tr>
<td>IEAVNP1F</td>
<td>00:00:00.232</td>
<td>SRM - I/O measurement blocks</td>
</tr>
<tr>
<td>IEAVNP51</td>
<td>00:00:00.262</td>
<td>TRACE</td>
</tr>
<tr>
<td>IEAVNP20</td>
<td>00:00:00.029</td>
<td>Process CLOCK=</td>
</tr>
<tr>
<td>IEAVNP21</td>
<td>00:00:13.894</td>
<td>TOD clock</td>
</tr>
<tr>
<td>IEAVNP57</td>
<td>00:00:00.026</td>
<td>SDUMP</td>
</tr>
<tr>
<td>IEAVNP9</td>
<td>00:00:17.681</td>
<td>XCF</td>
</tr>
<tr>
<td>IEAVNP33</td>
<td>00:00:31.126</td>
<td>GRS</td>
</tr>
<tr>
<td>IEAVNP1M</td>
<td>00:00:00.017</td>
<td>License manager</td>
</tr>
<tr>
<td>IEAVNPED</td>
<td>00:00:00.033</td>
<td>PROD</td>
</tr>
<tr>
<td>IEAVNP26</td>
<td>00:00:01.683</td>
<td>SMS</td>
</tr>
<tr>
<td>IEAVNP9E5</td>
<td>00:00:05.086</td>
<td>LNKLIST</td>
</tr>
<tr>
<td>IEAVNP9D</td>
<td>00:00:00.718</td>
<td>Load pageable device support modules</td>
</tr>
<tr>
<td>IEAVNP88</td>
<td>00:00:00.269</td>
<td>Allocation move EDT II</td>
</tr>
<tr>
<td>IEAVNPA1</td>
<td>00:00:29.221</td>
<td>CONSOLE</td>
</tr>
<tr>
<td>IEAVNPDC</td>
<td>00:00:00.153</td>
<td>WLM</td>
</tr>
<tr>
<td>IEAVNP16</td>
<td>00:00:01.358</td>
<td>EXCP appendages</td>
</tr>
<tr>
<td>IEAVNP13</td>
<td>00:00:00.127</td>
<td>Prepare NIP/MSI interface</td>
</tr>
<tr>
<td>IEAVNP17</td>
<td>00:00:00.033</td>
<td>GTF Monitor Call interface</td>
</tr>
<tr>
<td>IEAVNP8G</td>
<td>00:00:00.048</td>
<td>VSM defined monitor call enablement</td>
</tr>
<tr>
<td>IEAVNP18</td>
<td>00:00:00.614</td>
<td>PARMLIB Scan Routine interface</td>
</tr>
<tr>
<td>IEAVNP2</td>
<td>00:00:00.327</td>
<td>Process IOS=</td>
</tr>
</tbody>
</table>
## IPCS Display IPL Statistic...

<table>
<thead>
<tr>
<th>Command</th>
<th>Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEAVNP15</td>
<td>00:00:01.401</td>
<td>Process VATLST</td>
</tr>
<tr>
<td>IEAVNPRR</td>
<td>00:00:00.020</td>
<td>RRS</td>
</tr>
<tr>
<td>IEAVNPOE</td>
<td>00:00:01.512</td>
<td>USS</td>
</tr>
<tr>
<td>IEAVNPLE</td>
<td>00:00:00.112</td>
<td></td>
</tr>
<tr>
<td>IEAVNPUN</td>
<td>00:00:02.481</td>
<td>Unicode</td>
</tr>
<tr>
<td>IEAVNPXE</td>
<td>00:00:00.029</td>
<td></td>
</tr>
<tr>
<td>IEAVNP1B</td>
<td>00:00:00.381</td>
<td>Close catalog</td>
</tr>
<tr>
<td>IEAVNIPX</td>
<td>00:00:00.001</td>
<td>Nip final cleanup</td>
</tr>
</tbody>
</table>

**Total NIP Time:** 00:03:53.732
IPCS Display IPL Statistic...

*** IEEVIPL Statistics ***

IEETRACE 00:00:00.013 Master trace
ISNMSI 00:00:01.526 SPI
UMPECBM 00:00:02.734 CONSOLE address space
ENPFC005 00:00:00.000 CONSOLE ready ENF
IEFSCHIN 00:00:00.872 IEFSCHAS address space
IEFJSINT 00:00:00.003 Subsystem interface
IEFSJLOD 00:00:00.134 JESCT
IAZINIT 00:00:00.250 JESXCF address space
IAEFSII 00:00:00.104 FSI trace
IEFQBINI 00:00:00.134 SWA manager
IEFAB4I0 00:00:00.673 ALLOCAS address space

IEEVIPL 00:00:06.448 Uncaptured time: 00:00:00.000

*** IEEMB860 Statistics ***

ILRTMRLG 00:00:00.687 ASM
IECVIOSI 00:00:38.264 IOS dynamic pathing
ATBINSYS 00:00:00.012 APPC
IKJEFSXR 00:00:00.194 TSO
IXGBLF00 00:00:00.029 Logger
ILMINITM 00:00:00.036 License manager
COMMNDXX 00:00:24.721 COMMANDxx processing
SMFWAIT 00:00:00.098 SMF
SECPROD 00:00:19.375 Security server
IEFJSIN2 00:00:10.062 SSN= subsystem
IEFB4I2 00:00:00.194 ALLOCAS - UCB scan
CSRINIT 00:00:00.010 Windowing services
FINSHMSI 00:00:00.000 Wait for attached CMDs

IEEMB860 00:01:33.612 Uncaptured time: 00:00:00.098

Tip: in the IPCS dialog, to display the last IPL statistic using in-storage source rather than an SVC dump, proceed as follows:

1. Select IPCS option 6 (commands)
2. Issue DROPD MAIN
3. (delete data from a previous IPCS session using in-storage as source)
4. Issue VERBX BLSAIPST MAIN

Total Time: 00:05:46.708
## Terms and Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASM</td>
<td>Auxiliary Storage Manager</td>
</tr>
<tr>
<td>ENF</td>
<td>Event Notification Facility</td>
</tr>
<tr>
<td>IOCP</td>
<td>I/O Configuration Program</td>
</tr>
<tr>
<td>IOS</td>
<td>Input/Output Supervisor</td>
</tr>
<tr>
<td>IRIM</td>
<td>IPL Resource Initialization Module</td>
</tr>
<tr>
<td>MCH</td>
<td>Machine Check Handler</td>
</tr>
<tr>
<td>MIH</td>
<td>Missing Interrupt Handler</td>
</tr>
<tr>
<td>NIP</td>
<td>Nucleus Initialization Phase</td>
</tr>
<tr>
<td>RIM</td>
<td>Resource Initialization Module</td>
</tr>
<tr>
<td>RTM</td>
<td>Recovery Termination Manager</td>
</tr>
<tr>
<td>SRM</td>
<td>System Resource Manager</td>
</tr>
<tr>
<td>SYSRES</td>
<td>System residence Volume</td>
</tr>
<tr>
<td>UCB</td>
<td>Unit Control Block</td>
</tr>
<tr>
<td>UIM</td>
<td>Unit Information Module</td>
</tr>
<tr>
<td>DAT</td>
<td>Dynamic Address Translation</td>
</tr>
<tr>
<td>IOCDS</td>
<td>I/O Configuration Data Set</td>
</tr>
<tr>
<td>IODF</td>
<td>I/O Definition File</td>
</tr>
<tr>
<td>IPL</td>
<td>Initial Program Load</td>
</tr>
<tr>
<td>JES</td>
<td>Job Entry Subsystem</td>
</tr>
<tr>
<td>MCS</td>
<td>Multiple Console Support</td>
</tr>
<tr>
<td>MSI</td>
<td>Master Scheduler Initialization</td>
</tr>
<tr>
<td>POR</td>
<td>Power-on-Reset</td>
</tr>
<tr>
<td>RSM</td>
<td>Real Storage Manager</td>
</tr>
<tr>
<td>SMS</td>
<td>System managed Storage</td>
</tr>
<tr>
<td>SVC</td>
<td>Supervisor Call</td>
</tr>
<tr>
<td>TOD</td>
<td>Time of Day Clock</td>
</tr>
<tr>
<td>UCW</td>
<td>Unit Control Word</td>
</tr>
<tr>
<td>VSM</td>
<td>Virtual Storage Management</td>
</tr>
</tbody>
</table>
The End...any Questions?