IBM Systems

Introduction to z/VM System Administration

z/VM Education Club – November 18, 2019

Bill Bitner
bitnerb@us.ibm.com

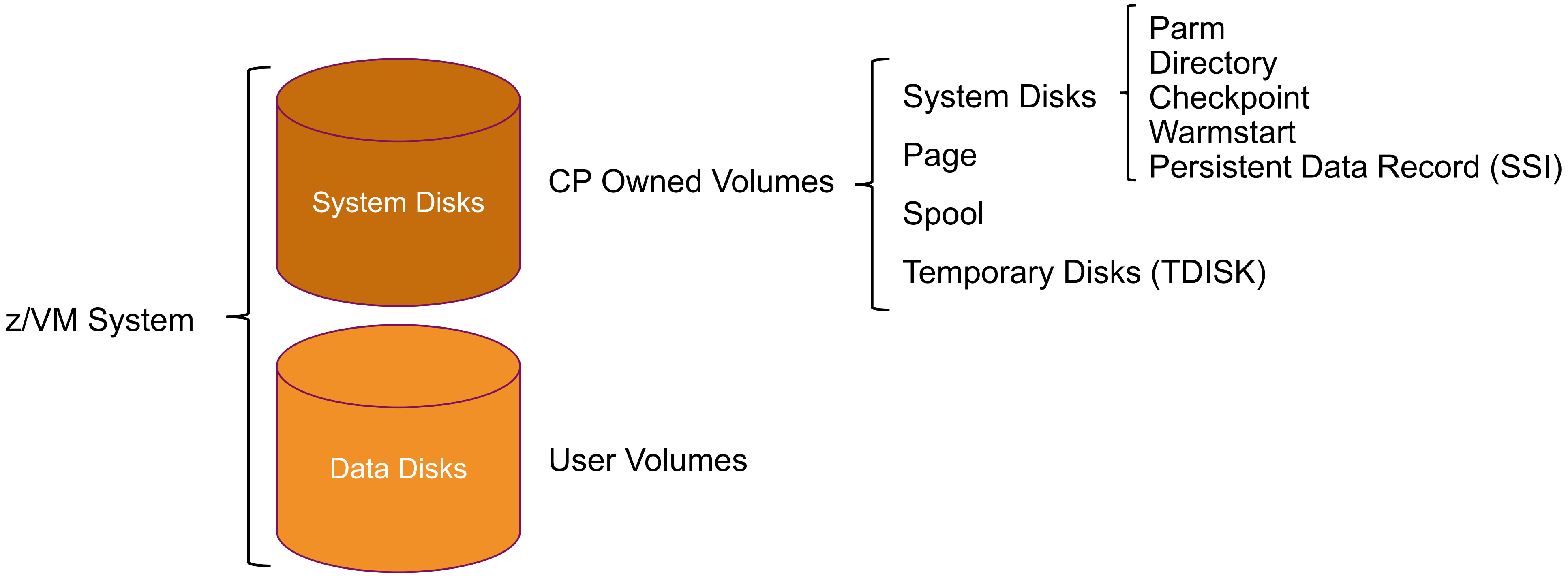
z/VM Development Client Focus & Care

Introduction

- Going beyond just running a z/VM System, understanding:
  - The critical pieces that make up a z/VM system
  - How those pieces are connected
  - How to make changes to some of those pieces

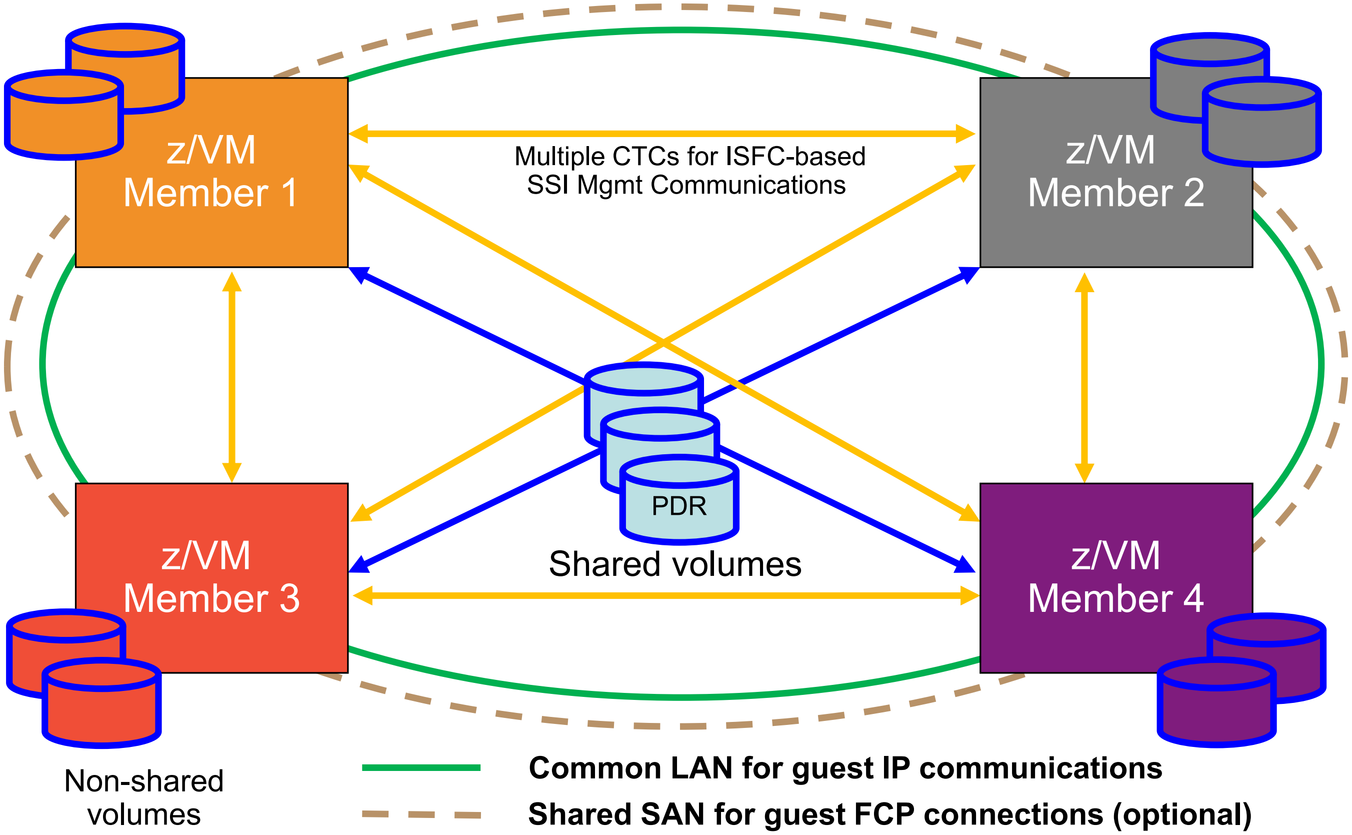
- This presentation borrows heavily from:
  - "Configuring, Customizing, and Modifying Your z/VM System" by John Franciscovich
  - "z/VM Begins" by Timothy Greer http://www.vm.ibm.com/library/presentations/vm-begin.pdf

50,000 Foot View



- z/VM System
  - System Disks — CP Owned Volumes
    - System Disks
      - Parm
      - Directory
      - Checkpoint
      - Warmstart
      - Persistent Data Record (SSI)
    - Page
    - Spool
    - Temporary Disks (TDISK)
  - Data Disks — User Volumes

Single System Image Cluster Configuration



z/VM Member 1

z/VM Member 2

z/VM Member 3

z/VM Member 4

Multiple CTCs for ISFC-based SSI Mgmt Communications

PDR

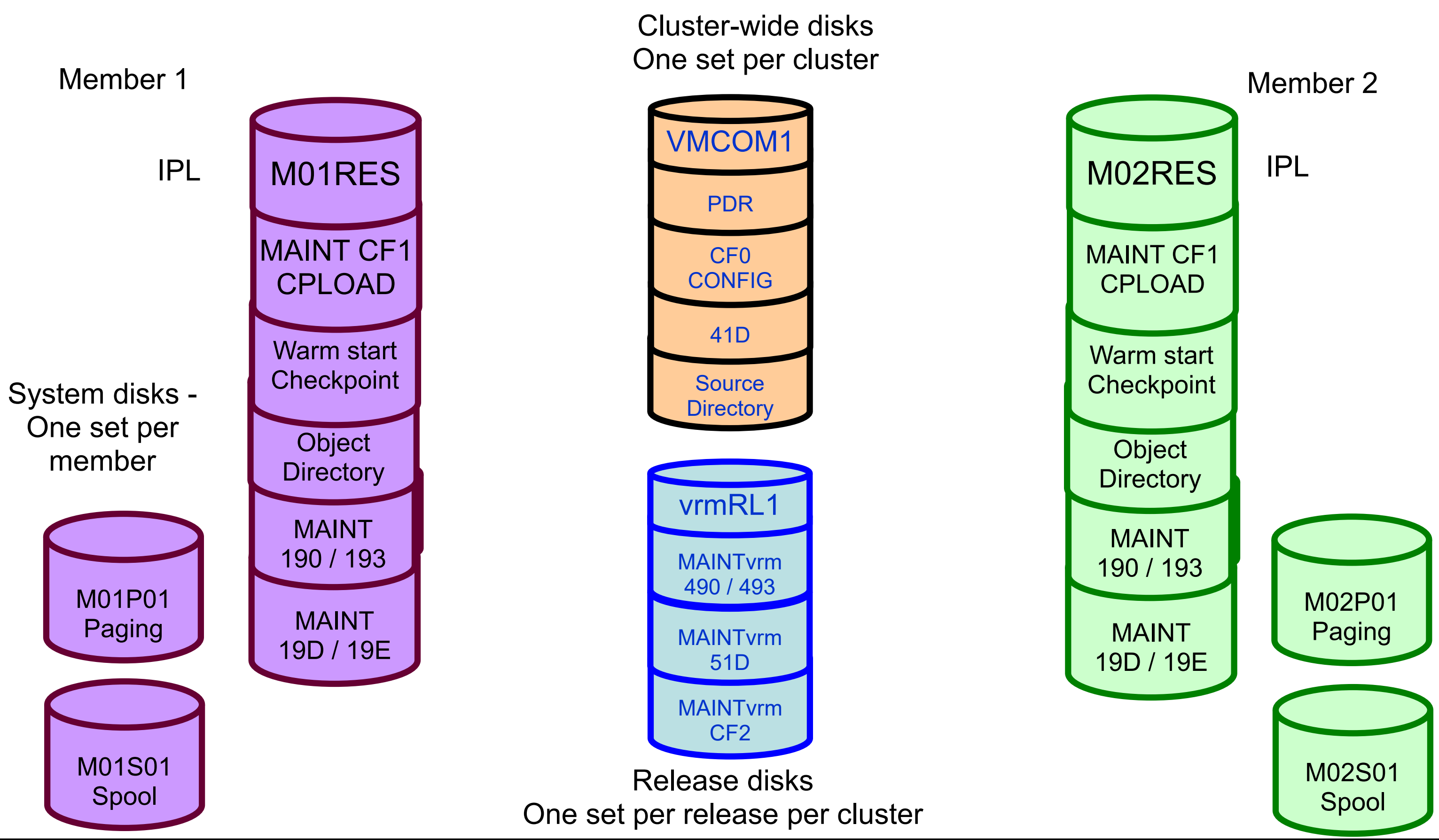
Shared volumes

Non-shared volumes

Common LAN for guest IP communications

Shared SAN for guest FCP connections (optional)

z/VM Single System Image PARM and IPL Disk Layout



Recipe for a z/VM System

- `CPLOAD MODULE`
  - Nucleus or kernel in a module that can be loaded into memory
- `SYSTEM CONFIG`
  - System configuration file describes which devices are involved, features to enable, virtual machines involved or to be started
- `LOGO CONFIG`
  - The logo displayed on terminals connected to z/VM
- `USER DIRECT`
  - User directory describes virtual machines (users) that you want defined to the system

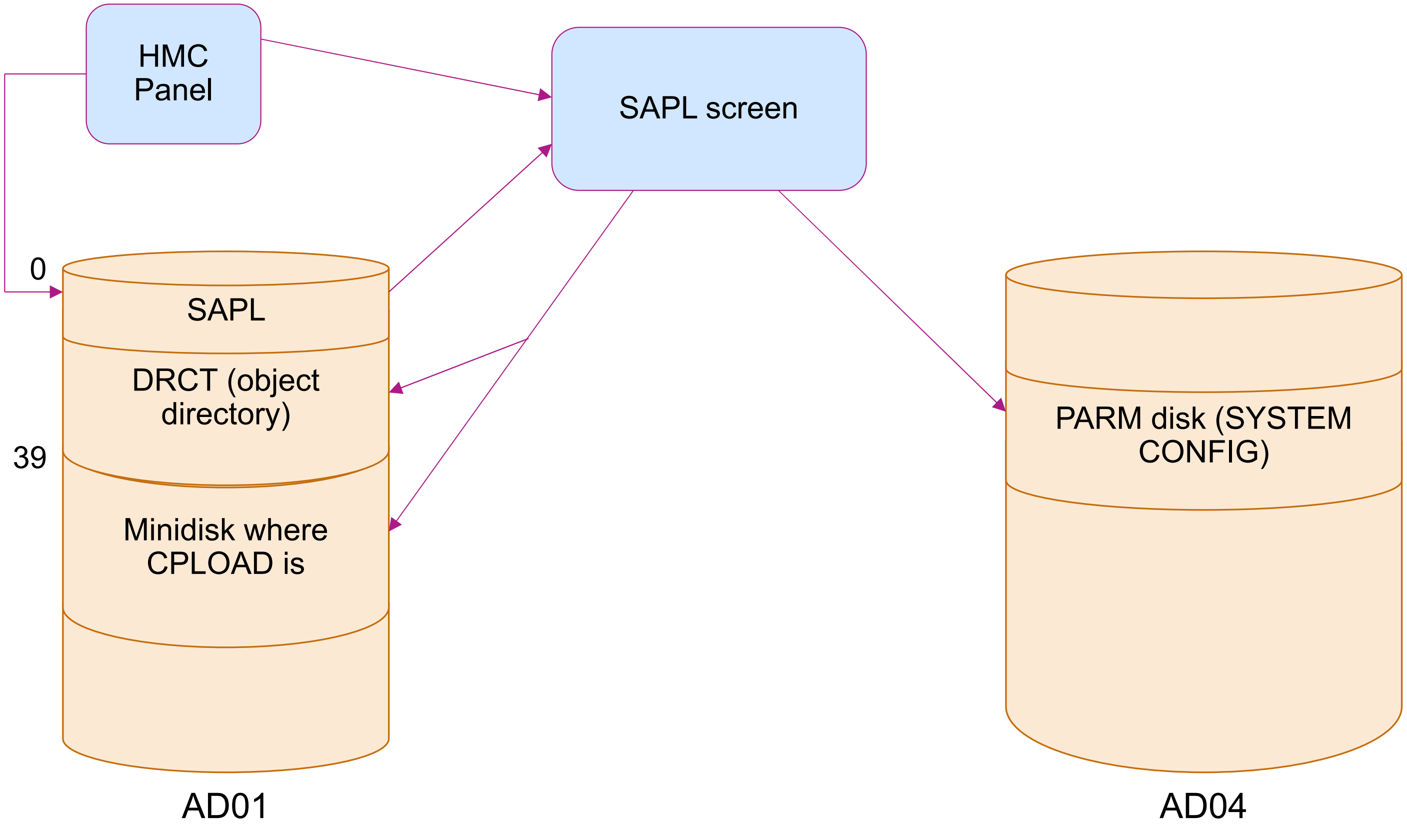
- Other key disk locations:
  - Checkpoint area: where z/VM records accounting and other information on disk instead of memory
  - Warm Start area: where z/VM saves information on the spool files and system to use across outages
  - Page space: paging areas for virtual memory that does not fit in real memory
  - Spool space: space for spool files both from users and system maintained, such as dump space

How do we get there?

- Stand Alone Program Loader (SAPL)
  - –Type of bootstrap program that can be used to start other programs, in our case the z/VM control program kernel (**`CPLOAD MODULE`**)
  - –Is aware of the structure used in z/VM systems
  - –Normally configured to automatically start the z/VM system
  - –Can be used to:
    - Modify certain settings
    - Choose a different module (kernel) or system configuration file
    - Direct system to use a different console
    - Set an environment variable
    - Start other utilities (e.g. standalone dump)

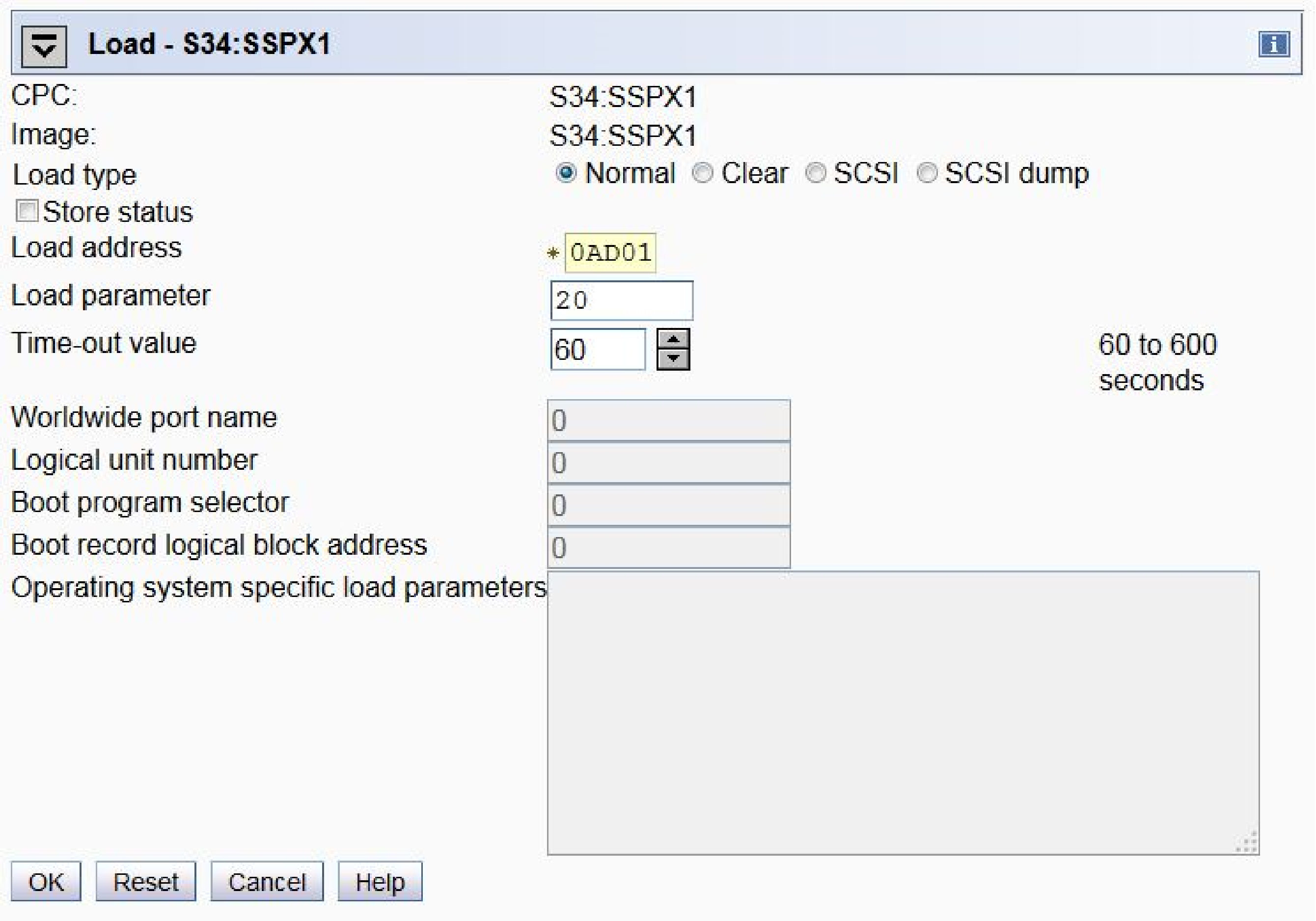
- Created and installed onto an IPL (Initial Program Load) device via **`SALIPL MODULE`** which is part of the z/VM product.

Finding the first pieces



HMC Load panel

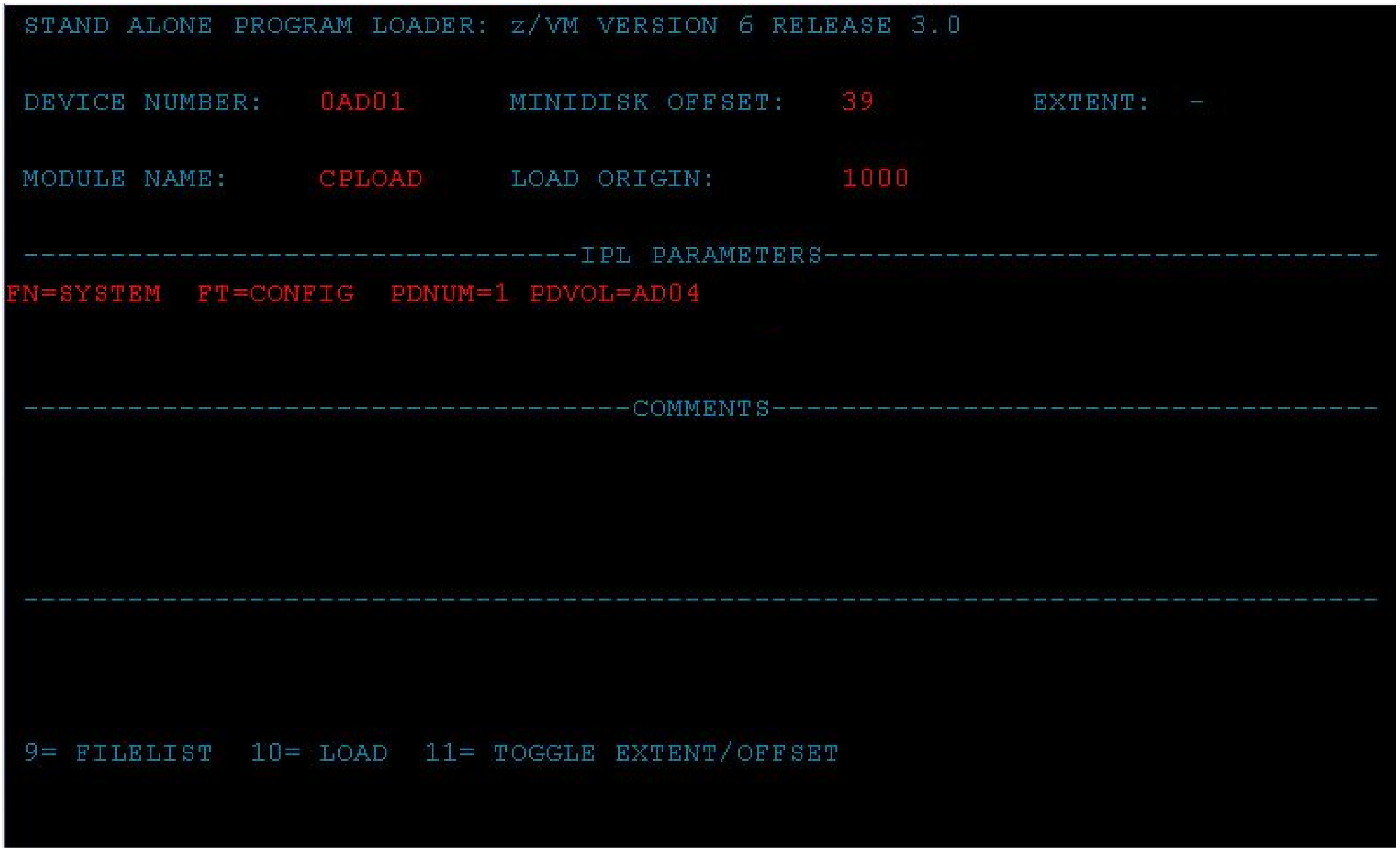
- Load type is Normal since our SAPL is on an ECKD device.
- Load address is AD01 which is where SAPL is located
- Load parameter shown here is the console device we're using for our z/VM system



Load - S34:SSPX1

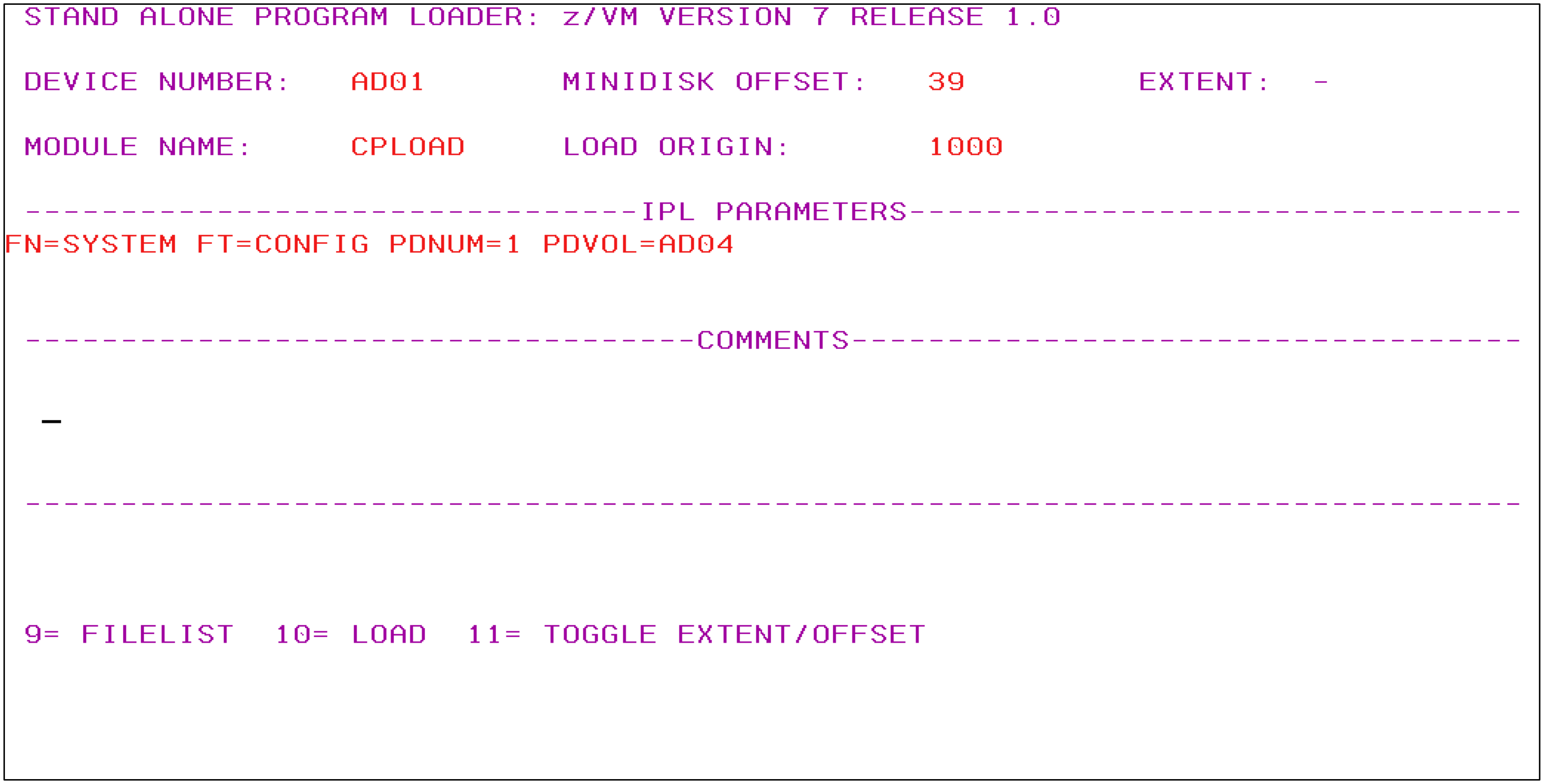
CPC:	S34:SSPX1
Image:	S34:SSPX1
Load type	● Normal ○ Clear ○ SCSI ○ SCSI dump
□ Store status	
Load address	* 0AD01
Load parameter	20
Time-out value	60 (60 to 600 seconds)
Worldwide port name	0
Logical unit number	0
Boot program selector	0
Boot record logical block address	0
Operating system specific load parameters	

Stand-Alone Program Loader (SAPL) screen



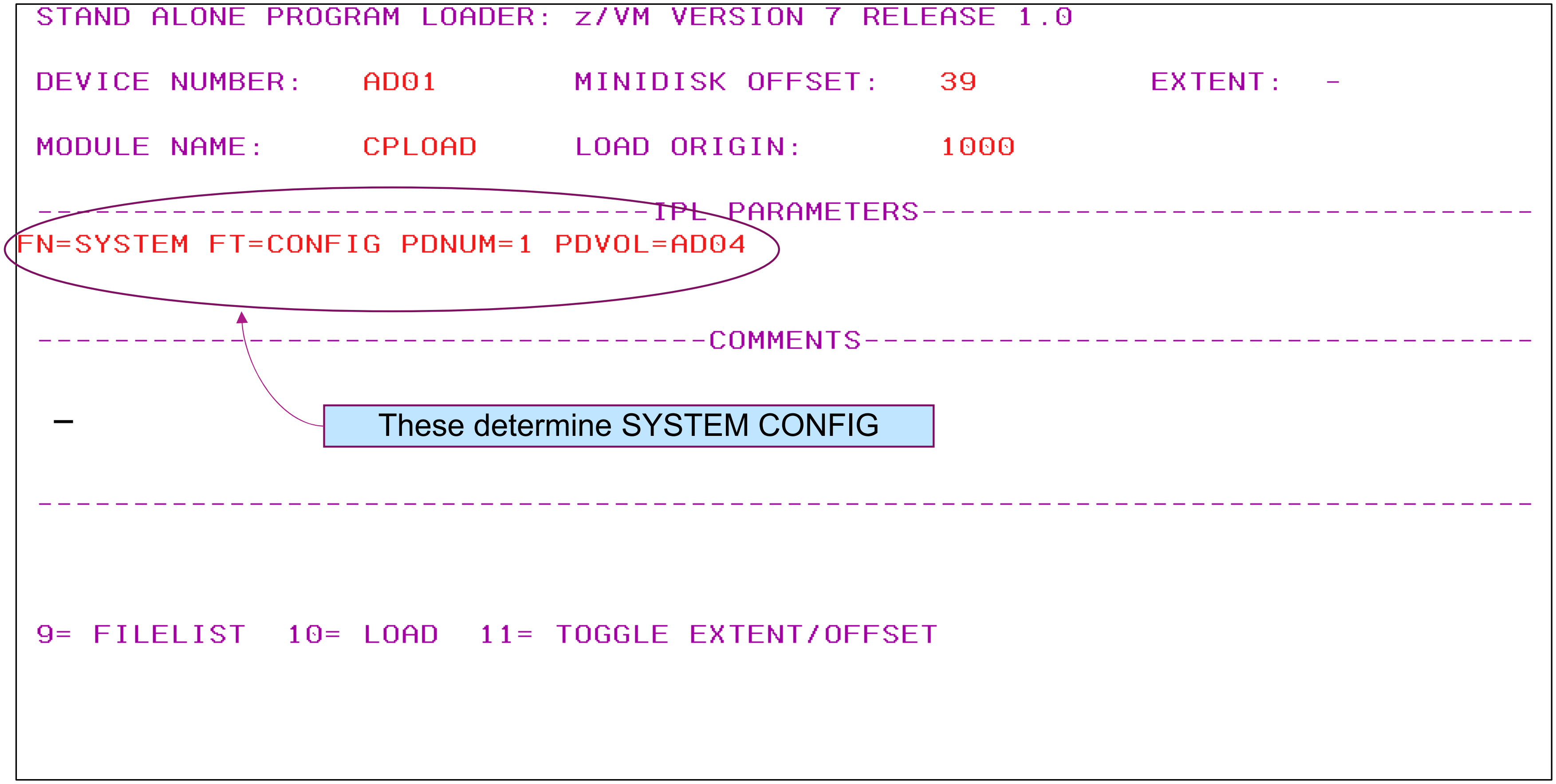
```
 STAND ALONE PROGRAM LOADER: z/VM VERSION 6 RELEASE 3.0

 DEVICE NUMBER:    0AD01      MINIDISK OFFSET:    39         EXTENT:  -

 MODULE NAME:      CPLOAD     LOAD ORIGIN:        1000

 --------------------------------IPL PARAMETERS---------------------------------
FN=SYSTEM  FT=CONFIG  PDNUM=1 PDVOL=AD04


 -----------------------------------COMMENTS------------------------------------




 -------------------------------------------------------------------------------




 9= FILELIST  10= LOAD  11= TOGGLE EXTENT/OFFSET
```

Stand-Alone Program Loader (SAPL) screen



```
 STAND ALONE PROGRAM LOADER: z/VM VERSION 7 RELEASE 1.0

 DEVICE NUMBER:    AD01        MINIDISK OFFSET:    39          EXTENT:   -

 MODULE NAME:      CPLOAD      LOAD ORIGIN:        1000

 --------------------------------IPL PARAMETERS--------------------------------
FN=SYSTEM FT=CONFIG PDNUM=1 PDVOL=AD04


 -----------------------------------COMMENTS-----------------------------------

  _

 ------------------------------------------------------------------------------



 9= FILELIST  10= LOAD  11= TOGGLE EXTENT/OFFSET
```

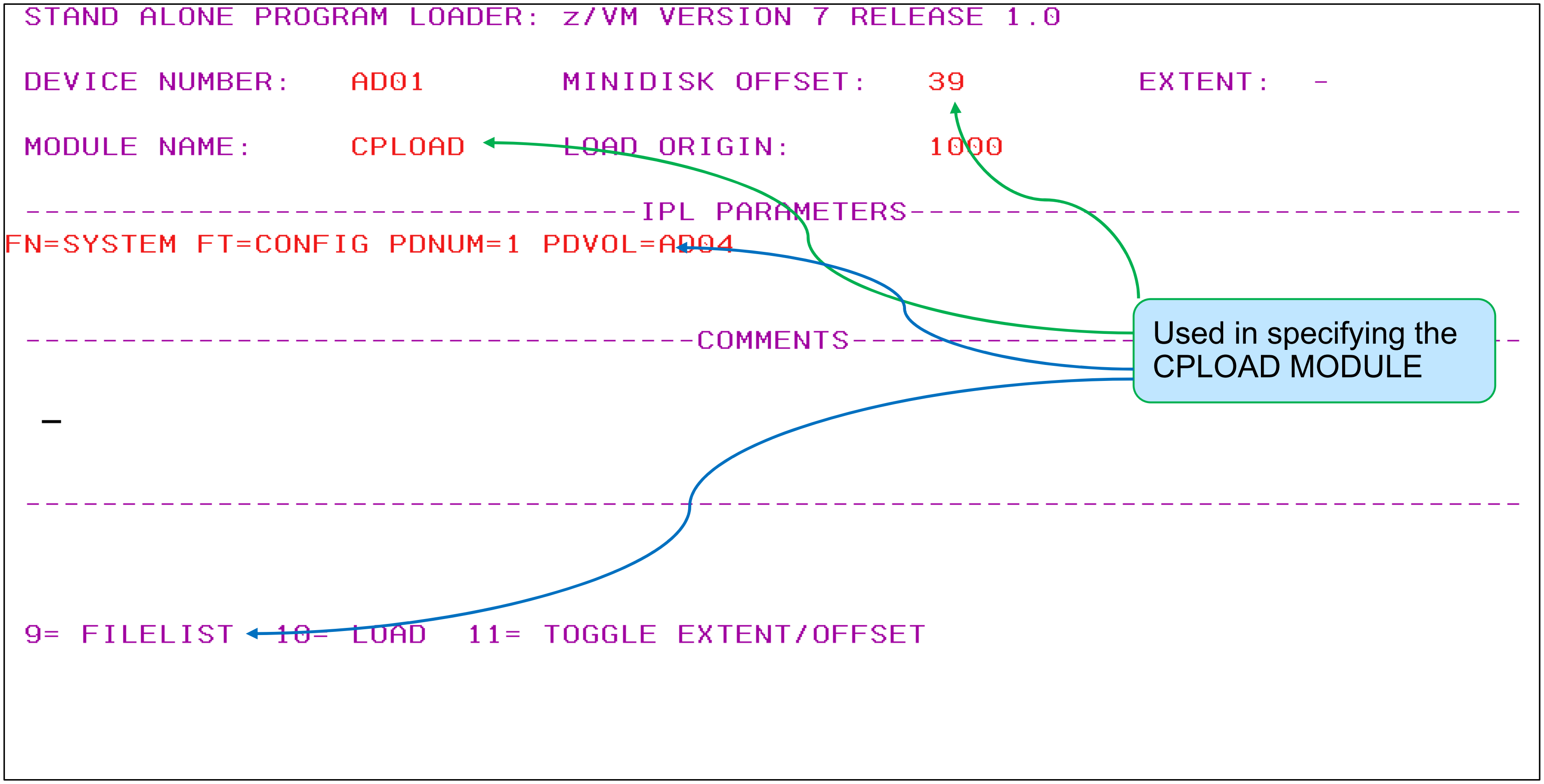
Stand-Alone Program Loader (SAPL) screen



```
STAND ALONE PROGRAM LOADER: z/VM VERSION 7 RELEASE 1.0

DEVICE NUMBER:    AD01        MINIDISK OFFSET:    39          EXTENT:   -

MODULE NAME:      CPLOAD      LOAD ORIGIN:        1000

-------------------------------------IPL PARAMETERS---------------------------------------
FN=SYSTEM FT=CONFIG PDNUM=1 PDVOL=AD04


-----------------------------------------COMMENTS-----------------------------------------

 _


------------------------------------------------------------------------------------------



9= FILELIST  10= LOAD  11= TOGGLE EXTENT/OFFSET
```

These determine SYSTEM CONFIG

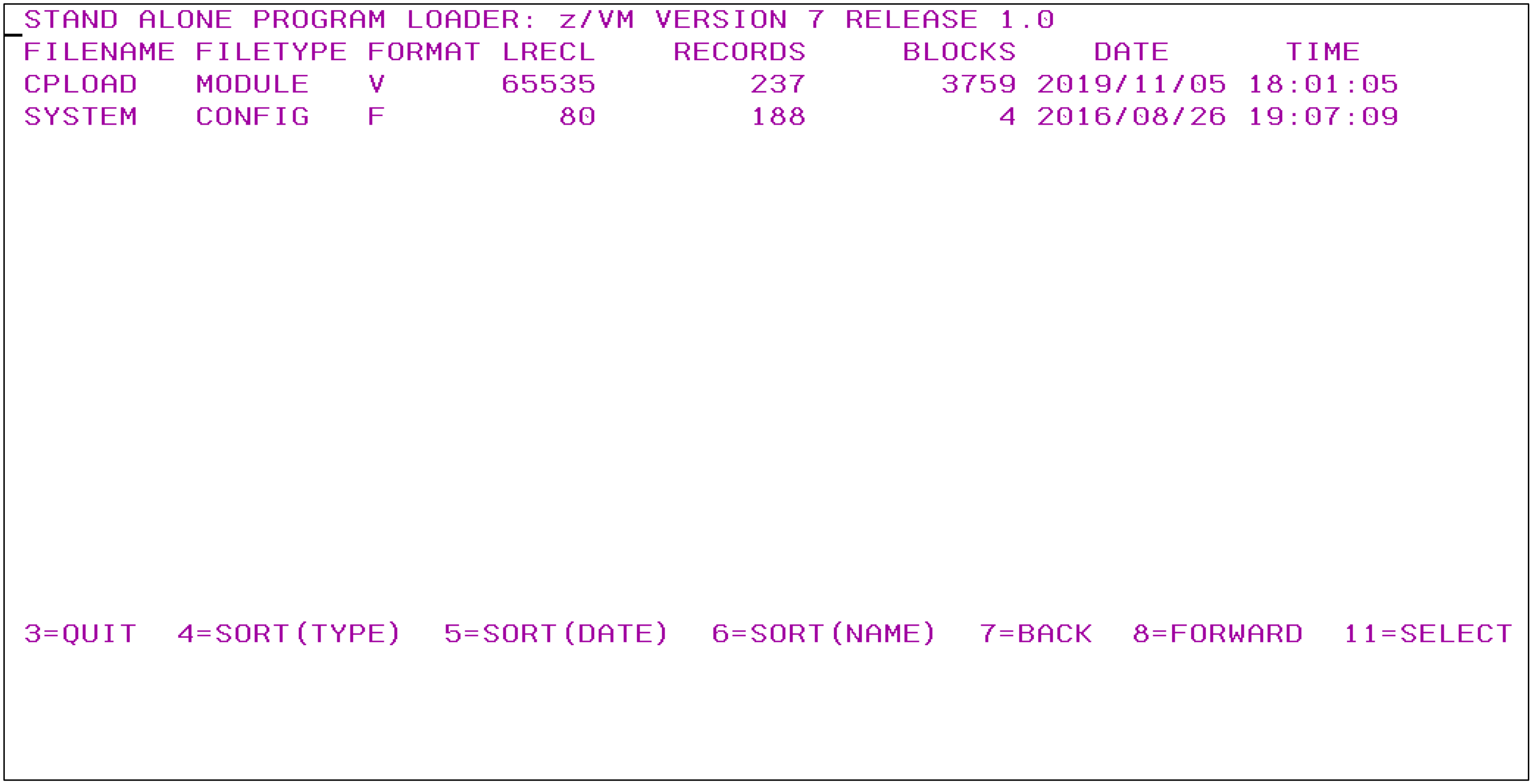
Stand-Alone Program Loader (SAPL) screen



```
STAND ALONE PROGRAM LOADER: z/VM VERSION 7 RELEASE 1.0

DEVICE NUMBER:    AD01        MINIDISK OFFSET:    39          EXTENT:   -

MODULE NAME:      CPLOAD      LOAD ORIGIN:        1000

--------------------------------IPL PARAMETERS--------------------------------
FN=SYSTEM FT=CONFIG PDNUM=1 PDVOL=AD04


-----------------------------------COMMENTS-----------------------------------

 _

------------------------------------------------------------------------------



9= FILELIST  10= LOAD   11= TOGGLE EXTENT/OFFSET
```

Used in specifying the CPLOAD MODULE

Using Filelist from SAPL



```
 STAND ALONE PROGRAM LOADER: z/VM VERSION 7 RELEASE 1.0
_FILENAME FILETYPE FORMAT LRECL    RECORDS     BLOCKS    DATE       TIME
 CPLOAD   MODULE   V      65535        237       3759 2019/11/05 18:01:05
 SYSTEM   CONFIG   F         80        188          4 2016/08/26 19:07:09











 3=QUIT  4=SORT(TYPE)  5=SORT(DATE)  6=SORT(NAME)  7=BACK  8=FORWARD  11=SELECT
```

Parm Disk

- Holds system definition information required at IPL

- CMS-formatted minidisk that CP can read

- z/VM Product Conventions:
  - PMAINT CF0 – main system configuration file and logo configuration file (default label VMCOM1 in ECKD environment).
  - MAINT CF1 – parm disk where CPLOAD MODULE is located
  - MAINT710 CF2 – staging area used by z/VM service process

System Configuration File

- A file you can set up and edit in CMS, but that CP reads as part of start-up and initialization. Typically called `SYSTEM CONFIG`

- Statements described in CP Planning and Administration http://www.vm.ibm.com/library/710pdfs/71627100.pdf

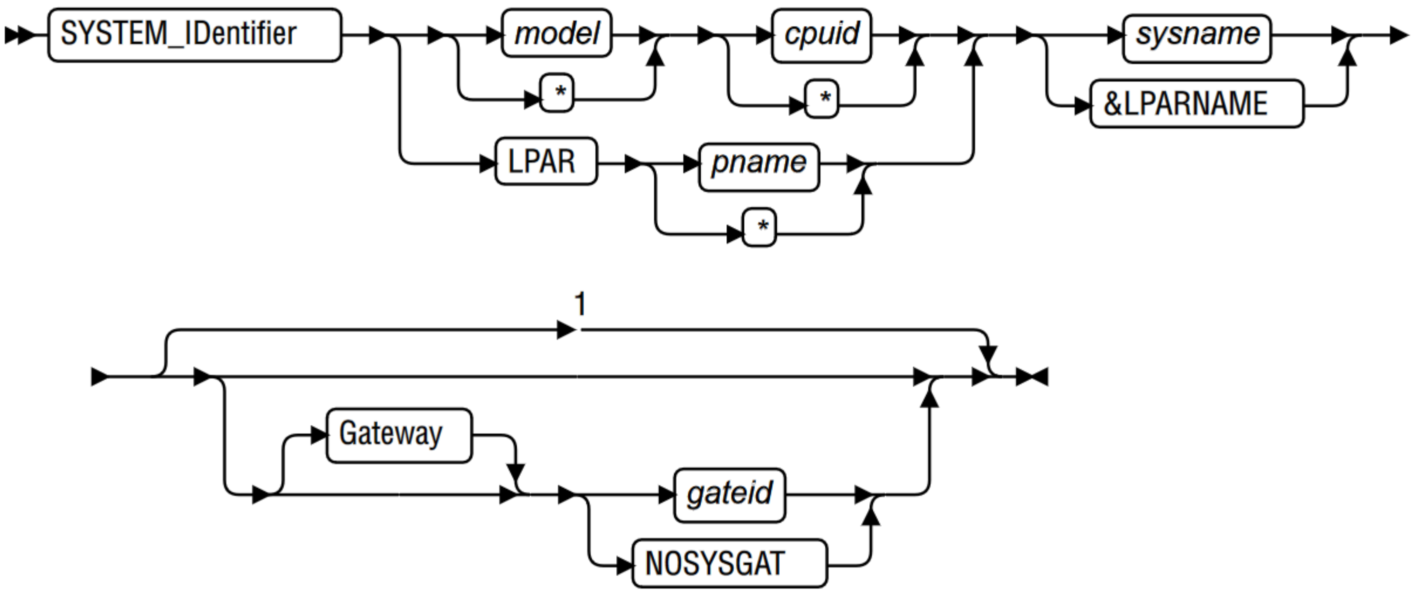
- Rules / Concepts
  - Valid comments as in REXX: /* comment */
    - Block or Line
  - Comma ‘,’ is the continuation character
  - Record qualifiers for specific systems as often have one system configuration file that applies to multiple systems (e.g. SSI Cluster)
    - E.g. `BOBVM1: Operator_Consoles 00F2`
    - Often used with `BEGIN` and `END`
- Tool to validate the configuration file:
  - `CPSYNTAX`

Anatomy of the System Configuration File – System Identifier

- Need to define the system name. Two system configuration statements allow this:
  - `System_Identifier_Default` allows you to have a name assigned if you don't want to specifically associate it with a particular physical or logical configuration.
  - `System_Identifier` has multiple options to associate the environment with a particular name.

```
/**********************************************************************/
/*                   System_Identifier Information                    */
/**********************************************************************/

System_Identifier 8561 031757 GUERNSEY /* For z15 with SN# 031757     */
System_Identifier * * &LPARNAME        /* Use the name of the LPAR    */
System_Identifier LPAR WHITE LEGHORN   /* LPAR name WHITE is LEGHORN  */
System_Identifier_Default BEAR         /* BEAR is the default name    */
```

Anatomy of the System Configuration File – System Identifier - Syntax



Notes:

[1] The default gateway name is the system name.

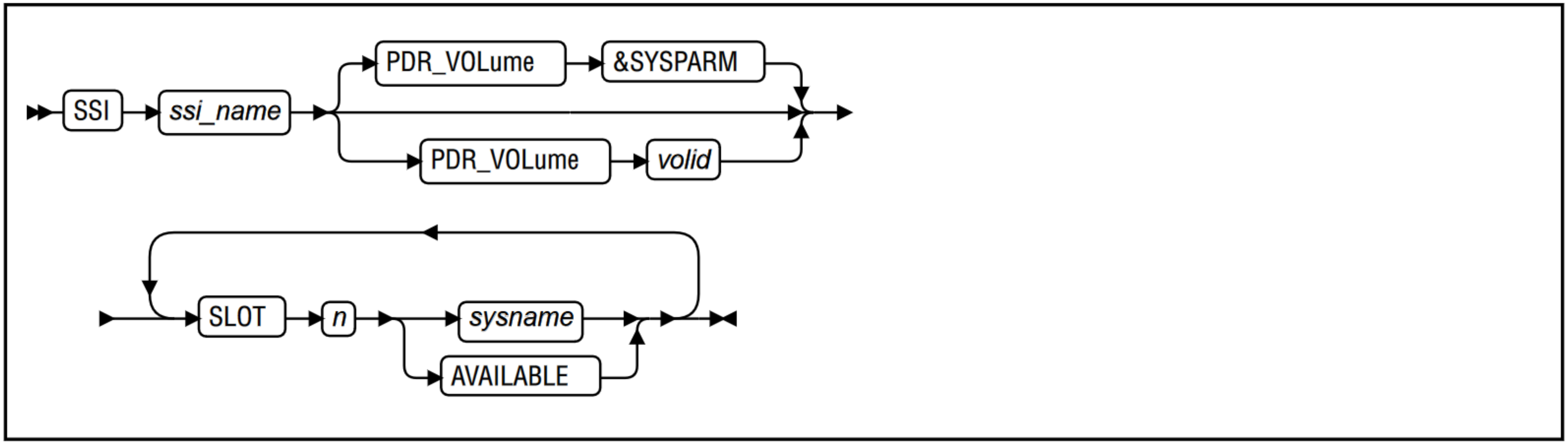
Anatomy of the System Configuration File – SSI Statement

- If we're running a Single System Image (SSI) Cluster we need to define that as well. This is done with the `SSI` statement.

In the z/VM Education Club we'll do a non-SSI environment first for simplicity.

```
/**********************************************************************/
/*         SSI Statement required for VMSSI feature                   */
/**********************************************************************/

  SSI CLUBSSI  PDR_VOLUME VMCOM1,
        SLOT 1 MEMBER1,
        SLOT 2 MEMBER2,
        SLOT 3 MEMBER3,
        SLOT 4 MEMBER4
```

Anatomy of the System Configuration File – SSI Statement - Syntax

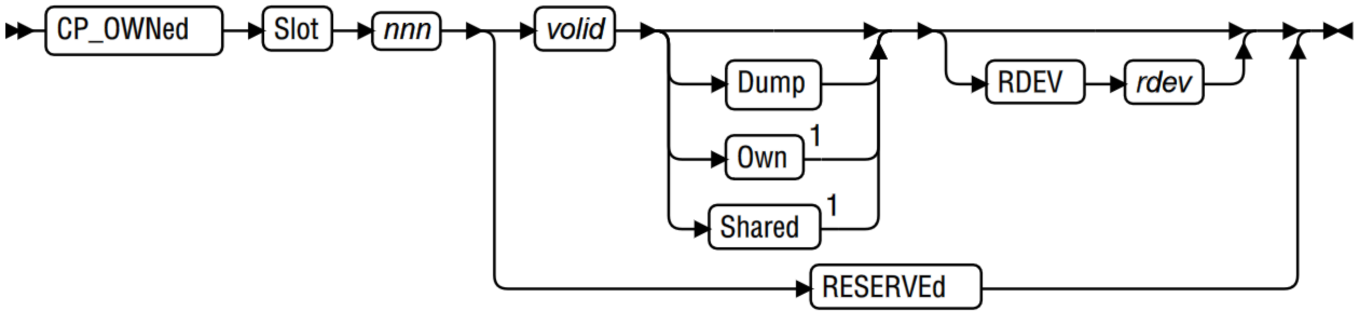


Anatomy of the System Configuration File – CP owned volumes

- Telling z/VM which volumes are system volumes
- Up to 255 volumes organized in ‘slots’.
- Order in the slots is important for spool, so usually have them counting up, and page/others count down..
- Volumes included should be: system residence volume and any volumes containing paging, spooling, dump, directory, and temporary disk space.

```
/**********************************************************************/
/*                     CP_Owned Volume Statements                     */
/**********************************************************************/

 CP_Owned   Slot   1  ZEDRES      /* Resident volume                  */
 CP_Owned   Slot  10  ZEDS01      /* Spool volume 1                   */
 CP_Owned   Slot  11  ZEDS02      /* Spool volume 2                   */

/* Room here for additional spool volumes counting up slot count      */

 CP_Owned   Slot 255  ZEDP01      /* Page volume 1                    */
```

Anatomy of the System Configuration File – CP owned volumes - Syntax



Notes:

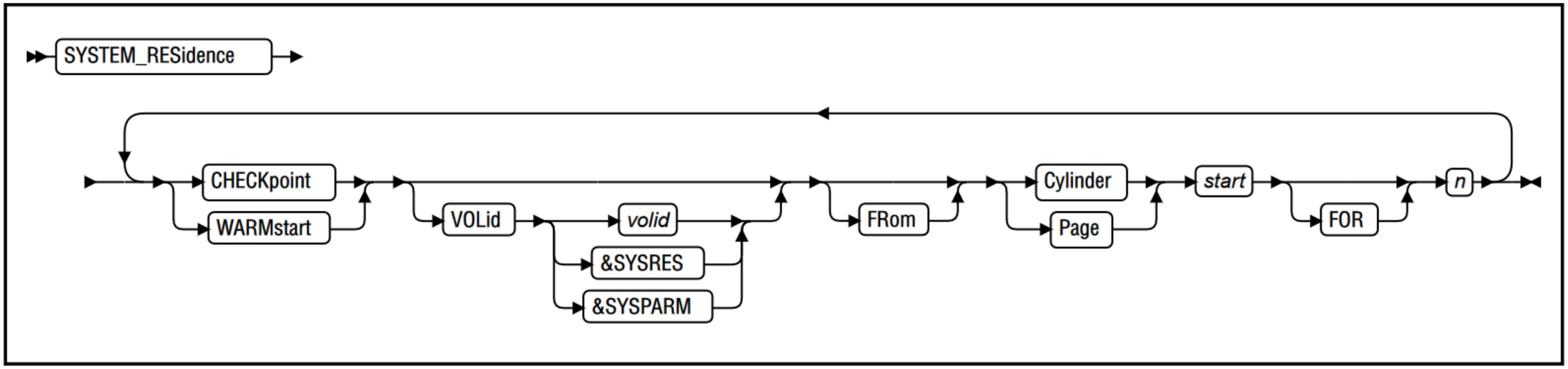
[1] The OWN and SHARED operands are ignored and are included for compatibility only.

Anatomy of the System Configuration File – System Areas

- The checkpoint and warmstart areas are important for ‘hardening’ information used
- These are defined with the `System_Residence` statement
- Maximum of 9 cylinders on ECKD or 2000 blocks on FBA (SCSI)

```
/**********************************************************************/
/*                Checkpoint and Warmstart Information                */
/**********************************************************************/

  System_Residence,
    Checkpoint  Volid M01RES    From CYL 21  For 9 ,
    Warmstart   Volid M01RES    From CYL 30  For 9
```

Anatomy of the System Configuration File – System Areas – Syntax



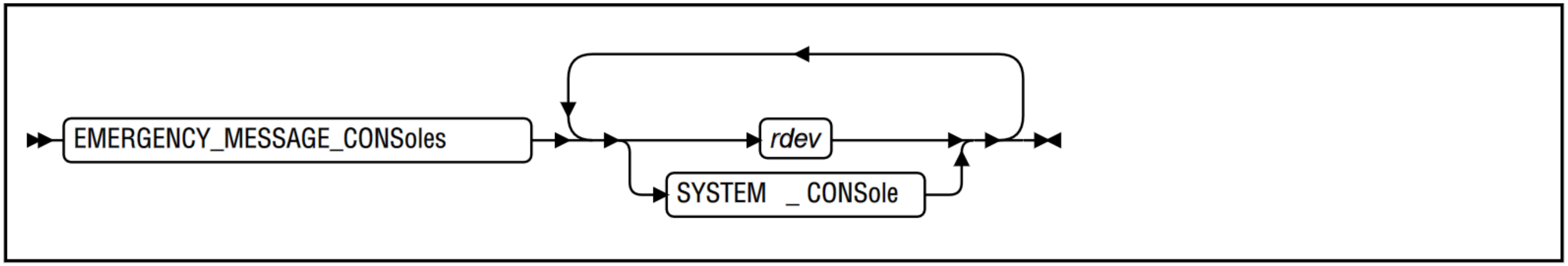
Anatomy of the System Configuration File – Consoles

- Need to let z/VM know which consoles it should enable and use for the system operator.
  - z/VM will use the first available device number or location to logon the system operator.

- Also let z/VM know where to send emergency messages
  - z/VM will send messages (e.g. abend, shutown, etc) to all consoles on list

```
/**********************************************************************/
/*                     Console Definitions                            */
/**********************************************************************/

 Operator_Consoles     0020 0021 0022 0023 0E20 0E21 1020 ,
                       System_3270 System_Console

 Emergency_Message_Consoles   0020 0021 0022 0023 0E20 0E21 1020 ,
                              System_Console
```

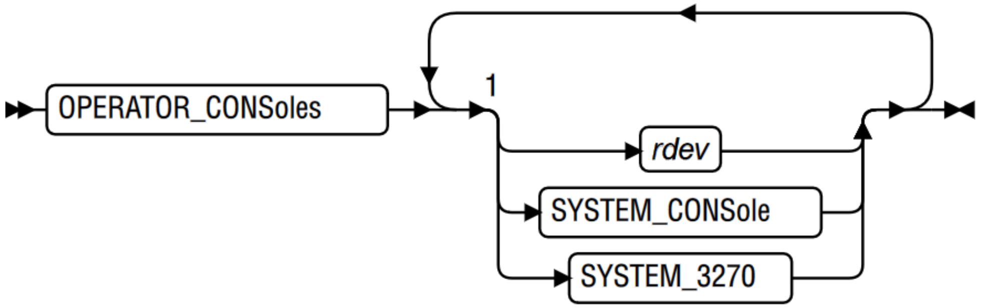
Anatomy of the System Configuration File – Consoles - Syntax



EMERGENCY_MESSAGE_CONSoles

rdev

SYSTEM _ CONSole



OPERATOR_CONSoles

1

rdev

SYSTEM_CONSole

SYSTEM_3270

Notes:

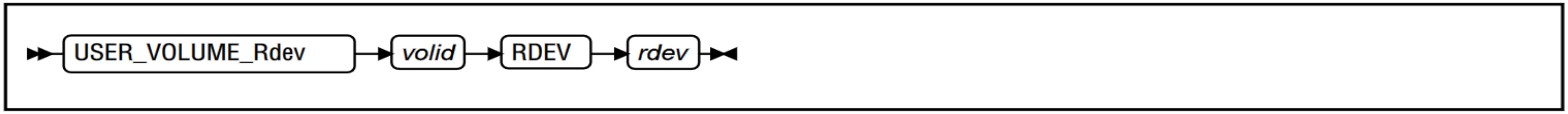
[1] The maximum number of device numbers that you can specify is 500.

Anatomy of the System Configuration File – User volumes

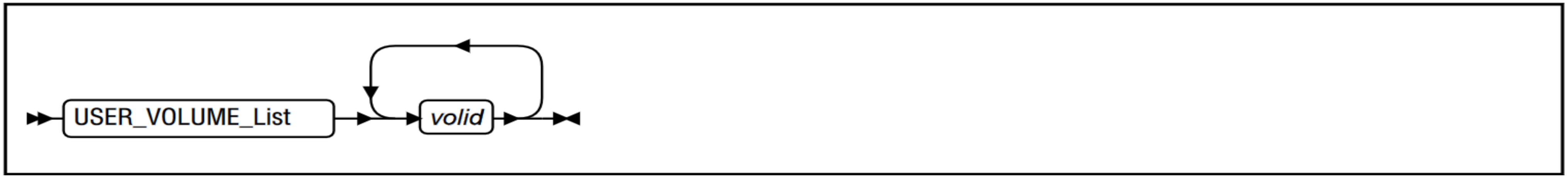
- Beyond CP owned volumes, you’ll often have other disks that virtual machines will use, but that are not critical to bring up and run z/VM. Typically contain minidisks.
- Statements that help accomplish this:
  - `User_Volume_List` gives the volume(s) to be part of the user volumes list
  - `User_Volume_RDEV` helps with duplicate volsers challenge
  - `User_Volume_Exclude` allows declaring volumes to not add to the list; utilizes wildcard charcters
  - `User_Volume_Include` allows declaring volumes to add to the list; utilizes wildcard characters

```
/**********************************************************************/
/*                          User Volume List                          */
/**********************************************************************/

    User_Volume_List USER01           /* User volume 1                */
    User_Volume_List USER02           /* User volume 2                */
    User_Volume_List PERF01           /* Monwrite volume              */
    User_Volume_Exclude ZOS*          /* Anything starting with ‘ZOS’ */
    User_Volume_Include ZOSSH%        /* ‘ZOSSH’ any 1 character      */
    User_Volume_Rdev WORK01 RDEV 0F23 /* The WORK01 at number 0F23    */
```

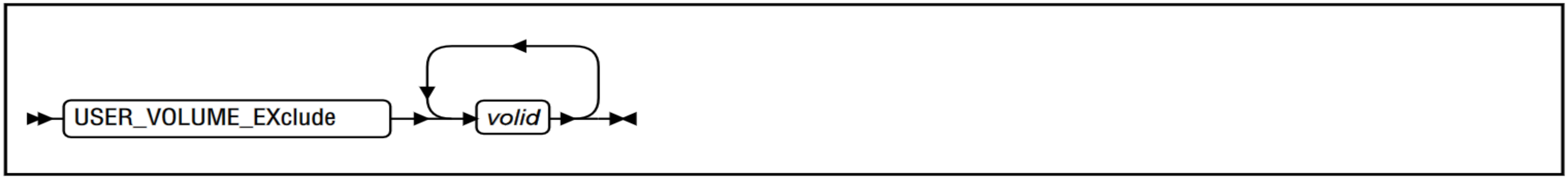
Anatomy of the System Configuration File – User volumes - Syntax



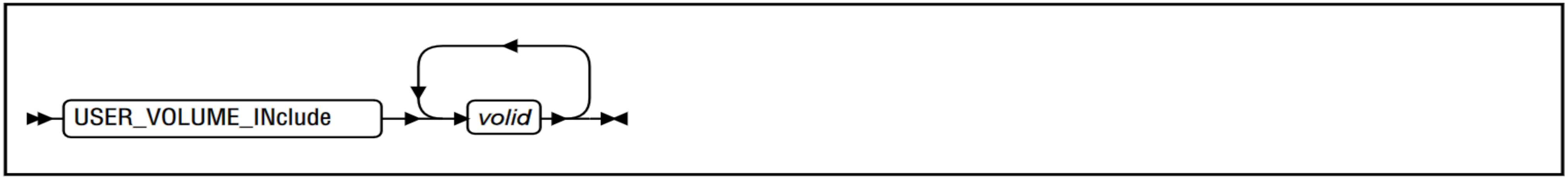
USER_VOLUME_Rdev *volid* RDEV *rdev*



USER_VOLUME_List *volid*



USER_VOLUME_EXclude *volid*



USER_VOLUME_INclude *volid*

Anatomy of the System Configuration File – Features

- The `Features` statement allows you to enable or disable z/VM control program features.
- Recommend using multiple statements instead of trying to create one large one. Especially when going to a new release. (Not like the following example.)

```
/**********************************************************************/
/*                         Features Statement                         */
/**********************************************************************/

 Features ,
   Retrieve ,                        /* Retrieve options              */
     Maximum  255 ,                  /* Maximum.... default is 255    */
   Disconnect_Timeout OFF,           /* Can be OFF, default is 15 min */

   Enable ,                          /* Enable the following features */
     Clear_TDisk ,                   /* Clear TDisks at IPL time      */
     Set_Privclass ,                 /* Allow SET PRIVCLASS command   */
   Disable ,                         /* Disable the following features */
      Validate_Shutdown              /* Don't require system name     */
```

Anatomy of the System Configuration File – I/O

- I/O gets very involved because of all the options. Multiple statements affect I/O devices, including:

– `Devices`	Determine which devices z/VM senses and brings online
– `RDEVICE`	Real devices defined
– `Define VSwitch`	One option of the Define statement, for a virtual switch
– `Crypto_APVirtual`	Shared Crypto Express adapter information
– `CU`	Control Unit
–`IODF`	Indicate HCD is to be used

- How devices might be treated:

– Accepted:	build real device control blocks so they can be used
– Initialize:	communicate with the device and prepare for use
– Sensed:	issue a sense ID request to the device to determine the device class or type
– Dynamically:	allow device to be changed dynamically
– Shared:	shared with independent operating systems (not part of an SSI cluster)

Anatomy of the System Configuration File – I/O

```
/**********************************************************************/
/*                        Status of Devices                           */
/**********************************************************************/

Devices
   NotAccepted 2F00-2F0F,                  /* Don't accept this string*/
   Accepted 2F0C,                          /* ... except this one     */
   Dynamic_I/O 0000-FFFF,                  /* Dynamic changes are ok  */
                                           /* on all the devices…     */
   NotDynamic_I/O 1900-19FF,               /* Except these!           */
   Initialized_at_IPL 0000-FFFF,           /* Initialize everything...*/
   NotInitialized_at_IPL 08E0-08EF,        /* ... almost.             */
   Offline_at_IPL 0A00-0AFF,               /* Keep these offline      */
   NotSensed 0100-01FF,                    /* Don’t try to sense ...  */
                                           /* ... these devices       */
   NotShared 0000-FFFF,                    /* Don't share anything ...*/
   Shared 1100-114F                        /* ... except these.       */
```

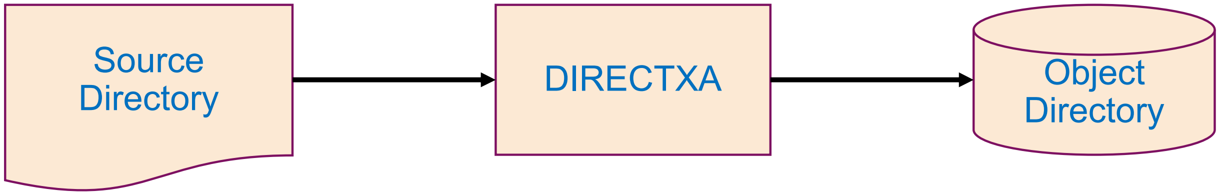
Logo File

- File that describes features of the logo that appears on the logon screen or printer separator page.

- Default one comes with the system, called `LOGO CONFIG`

- May want to change it to:
  - Make logo match a corporate image
  - Include legal or security notices
  - Change certain key phrases for fun

User Directory



- See Chapter 19 of CP Planning and Administration
- Source directory file – one or more CMS files
- DIRECTXA Utility is used to process the Source Directory and write the Object Directory
- Default source directory is `USER DIRECT`
- Typically, use a product to manage the z/VM directory (e.g. DirMaint from IBM)
- Good to know how to manage manually for small, test systems

User Directory - Location

- *Source* directory USER DIRECT starts out on PMAINT.2CC

- z/VM Control Program at initialization will:
  - look for a Directory area on a CP-owned volume, typically the system residence volume
  - brings that directory “online”
  - Most data areas in memory are not created until the virtual machine logs onto z/VM

- Some changes to directory will require the virtual machine to go through logon processing in order to take effect (However, for many of these there are also SET or DEFINE commands for dynamic changes)

User Directory Contents

- Comments indicated by an “*” in column 1
- A series of statements
  - Some statements or options no longer have any effect, but are their for compatibility
- Some statements allow continuation, but might have slightly different rules
- The first statement we need is the DIRECTORY statement, which will tell DIRECTXA where to write the object directory

```
****************************************************************
*  z/VM 7.1.0  SYSTEM DIRECTORY                                *
****************************************************************
*
DIRECTORY 400 3390 ZVMRES
*
* Uncomment following if using SSI                             *
*DIRECTORY SSI 123 ZVMRS1 ZVMRS2 ZVMRS3 ZVMRS4                 *
```

- In this example, it will check for a directory area on device number 400

User Directory – User Definitions Overview

- Different formats or types of virtual machine definitions
  - **Single-configuration** virtual machine – processed to create a single virtual machine
  - **Multiconfiguration** virtual machine – processed to create multiple virtual machines with the same userid, but different configurations, based on which member of an SSI cluster the virtual machine logons (e.g. TCPIP, PERFSVM)
    - Sometimes known as an IDENTITY user
  - **Pool** Users – processed to create a series of identical virtual machines with similar names ending in numbers

For now, for this class, we'll focus on single-configuration virtual machines.

User Definition Example

```
     1       2        3        4
USER PERFMON PASSWORD 32M 128M G
  MACHINE ESA
  IPL 190 PARM AUTOCR
  CONSOLE 0009 3215 T
  SPOOL   00C 2540 READER *
  SPOOL   00D 2540 PUNCH  A
  SPOOL   00E 1403 A
  LINK    MAINT 190 190 RR
  LINK    MAINT 19E 19E RR
  MDISK   191 3390 017 002 USER01 MR ALL ALL ALL
```

1. Virtual Machine Name:
   PERFMON

2. Password:
   PASSWORD

3. Memory
   Initial of 32M
   Max of 128M

4. Privilege Class
   G – General User

User Definition Example

```
USER PERFMON PASSWORD 32M 128M G
1 MACHINE ESA 3
2 IPL 190 PARM AUTOCR
  CONSOLE 0009 3215 T
  SPOOL   00C 2540 READER *
  SPOOL   00D 2540 PUNCH  A
  SPOOL   00E 1403 A
  LINK    MAINT 190 190 RR
  LINK    MAINT 19E 19E RR
  MDISK   191 3390 017 002 USER01 MR ALL ALL ALL
```

1. Machine mode:
   ESA – think of it like the IML mode

2. IPL Statement:
   Device is 190

3. Passed parameters
   Auto carriage return (Enter after IPL)
   Max of 1

User Definition Example

```
 USER PERFMON PASSWORD 32M 128M G
   MACHINE ESA
   IPL 190 PARM AUTOCR
1  CONSOLE 0009 3215 T
2  SPOOL   00C 2540 READER *
3  SPOOL   00D 2540 PUNCH  A
4  SPOOL   00E 1403 A
   LINK    MAINT 190 190 RR
   LINK    MAINT 19E 19E RR
   MDISK   191 3390 017 002 USER01 MR ALL ALL ALL
```

1. Define the console device:
   Device number 0009
   Type 3215
   Spool class T (This is default)

2. Virtual reader defintion:
   Device 00C, Class *

3. Virtual punch definition:
   Device 00D, Class A

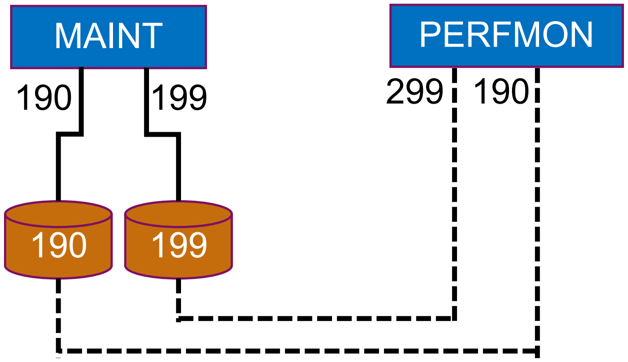
4. Virtual printer definition:
   Device 00E, Class A

User Definition Example

```
USER PERFMON PASSWORD 32M 128M G
  MACHINE ESA
  IPL 190 PARM AUTOCR
  CONSOLE 0009 3215 T
  SPOOL   00C 2540 READER *
  SPOOL   00D 2540 PUNCH  A
  SPOOL   00E 1403 A
1 LINK    MAINT 190 190 RR
2 LINK    MAINT 199 299 RR
  MDISK   191 3390 017 002 USER01 MR ALL ALL ALL
```

1. Links to other virtual machine mdisk:
   MAINT 190 linked as 190
   RR = Readonly

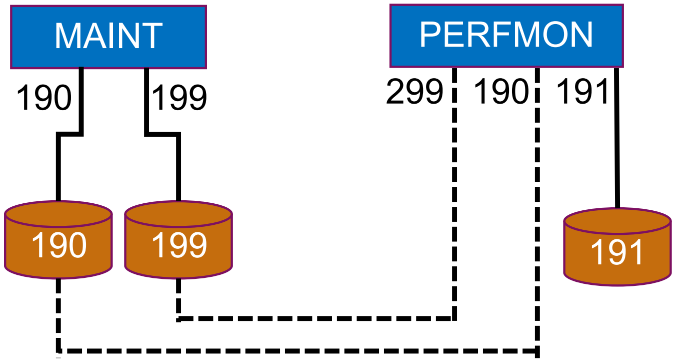
2. Links to other virtual machine mdisk:
   MAINT 199 linked as 299
   RR = Readonly



User Definition Example

```
USER PERFMON PASSWORD 32M 128M G
  MACHINE ESA
  IPL 190 PARM AUTOCR
  CONSOLE 0009 3215 T
  SPOOL   00C 2540 READER *
  SPOOL   00D 2540 PUNCH  A
  SPOOL   00E 1403 A
  LINK    MAINT 190 190 RR
  LINK    MAINT 199 299 RR
1 MDISK   191 3390 017 002 USER01 MR ALL ALL ALL
                   2       2      3   4
```

1. Define a minidisk:
   Device number 191
   3390 type
2. Location:
   On volume USER01
   From cylinder 17 to 18 (2 cylinders)
3. Linked write unless write link exists, then readonly
4. Passwords for linking dynamically
   ALL indicates anyone can link without a password



User Definition Example

```
USER PERFMON PASSWORD 32M 128M G
  MACHINE ESA
  IPL 190 PARM AUTOCR
  CONSOLE 0009 3215 T
  SPOOL   00C 2540 READER *
  SPOOL   00D 2540 PUNCH  A
  SPOOL   00E 1403 A
  LINK    MAINT 190 190 RR
  LINK    MAINT 199 299 RR
  MDISK   191 3390 017 002 USER01 MR ALL ALL ALL
```

- PERFMON
- 32 MB virtual memory
- IPLs 190 at logon
- Seven devices:
  - 009 console
  - 00C reader
  - 00D punch
  - 00E printer
  - 190 disk
  - 191 disk
  - 299 disk

User Directory Profiles

- Many virtual machines will have common or similar definitions. Profiles makes it easy to avoid duplicating the same statements over and over.

```
PROFILE TESTUSER
 SHARE RELATIVE 10
 MACHINE XC 10
 IPL CMS PARM AUTOCR
 CONSOLE 01F 3270 A
 SPOOL   00C 2540 READER *
 SPOOL   00D 2540 PUNCH  A
 SPOOL   00E 1403 A
 LINK    MAINT 190 190 RR
 LINK    MAINT 19D 19D RR

USER BILLY1 AUTOONLY 20M 20M  G
 INCLUDE TESTUSER
 MDISK   191 3390 017 002 USER01 MR
```

User BILLY1 would have all the statements in the profile TESTUSER and then also the MDISK statement.

Dummy Virtual Machines

- Some times define dummy virtual machines to reserve space for system areas, helps with utilities to map disk usage.

```
****************************************************************
*                   System DASD Layouts                        *
****************************************************************
USER $ALLOC$ NOLOG
MDISK   B01 3390 000 001 ONEBIT R
MDISK   B02 3390 000 001 TWOBIT R

USER $CPNUC$ NOLOG
MDISK   B01 3390 019 011 ONEBIT R

USER $DRCT$ NOLOG
MDISK   B02 3390 001 017 TWOBIT R

USER $SYSCKP$ NOLOG
MDISK   B01 3390 001 002 ONEBIT R

USER $SYSWRM$ NOLOG
MDISK   B01 3390 003 002 ONEBIT R
```

Other Utilities

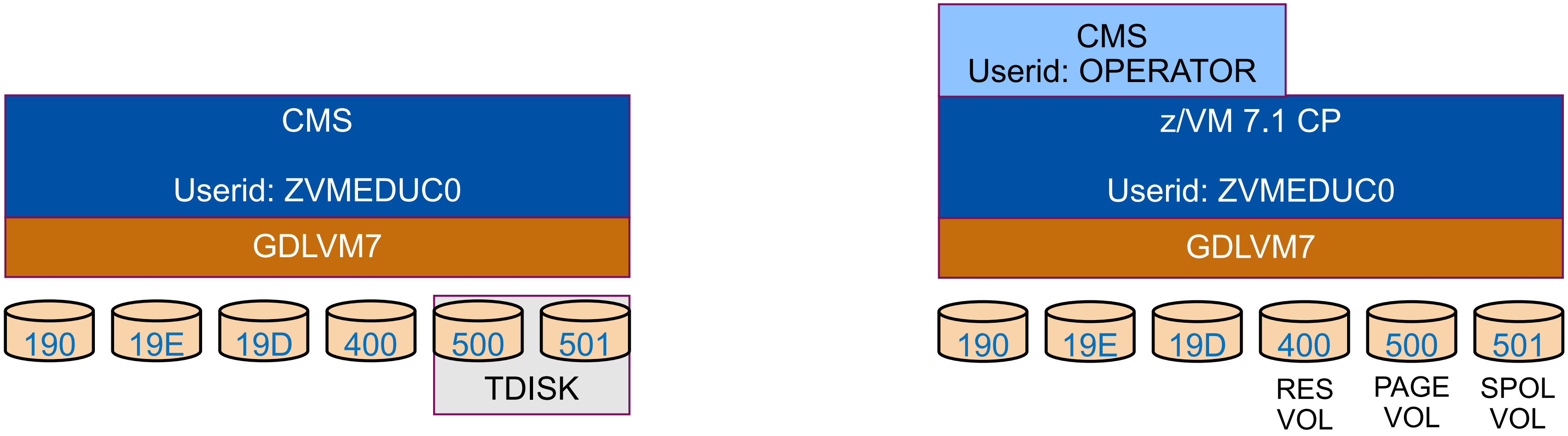
- SALIPL
  - used to install a copy of the Stand-Alone Program Loader (SAPL)
  - can be run in a CMS environment

- CPFMTXA
  - Utility to perform various functions on a Control Program volume
  - Key functions: Format and Allocate
  - Allocation type:
    - DRCT: Space for directory data
    - PAGE: Space for system paging activity
    - PARM: Space for the CMS-formatted minidisk holding System Configuration files
    - PERM: Space for CP user minidisks, checkpoint, warm start, and other static areas
    - SPOL: Space for spooling activity, dump, or paging overflow
    - TDSK: Space for temporary disk space

- DDR
  - Utility to copy, backup, and restore disk areas

Some thoughts on Second Level Systems

- We’ll use the z/VM IPL command instead of the HMC load screen
- We’ll build a very stream-lined z/VM System, taking some short cuts
  - Leverage TDisk from underlying z/VM for page and spool
  - Use 1st level CMS disks
  - Do much of the setup from CMS on the first level system.
  - Because of TDisk, not completely permanent; but if you want you can request additional disk space to make spool permanent.



Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries. For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml: AS/400, DBE, e-business logo, ESCO, eServer, FICON, IBM, IBM Logo, iSeries, MVS, OS/390, pSeries, RS/6000, S/30, VM/ESA, VSE/ESA, Websphere, xSeries, z/OS, zSeries, z/VM

The following are trademarks or registered trademarks of other companies

Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation
Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries
LINUX is a registered trademark of Linus Torvalds
UNIX is a registered trademark of The Open Group in the United States and other countries.
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.
Intel is a registered trademark of Intel Corporation
* 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.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

Any proposed use of claims in this presentation outside of the United States must be reviewed by local IBM country counsel prior to such use.

The information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

Permission is hereby granted to SHARE to publish an exact copy of this paper in the SHARE proceedings. IBM retains the title to the copyright in this paper, as well as the copyright in all underlying works. IBM retains the right to make derivative works and to republish and distribute this paper to whomever it chooses in any way it chooses.