


## z/VM For Beginners Hands On Lab Session 17471, 17480

*Richard Lewis – IBM Washington System Center  
rflewis@us.ibm.com*



SHARE is an independent volunteer-run information technology association that provides education, professional networking and industry influence.

Copyright (c) 2015 by SHARE Inc.  Except where otherwise noted, this work is licensed under <http://creativecommons.org/licenses/by-nc-sa/3.0/>



# Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

APPN*	GDPS*	POWERPC*	Virtual Image Facility
CICS*	Geographically Dispersed Parallel Sysplex	PR/SM	Virtualization Engine
DB2*	HiperSockets	Processor Resource/Systems Manager	VisualAge*
DFSMSMVS	HyperSwap	QMF	VM/ESA*
DFSMS/VM*	IBM*	RACF*	VSE/ESA
DirMaint	IBM eServer	Resource Link	VTAM*
Distributed Relational Database Architecture*	IBM logo*	RMF	WebSphere*
DRDA*	IBMLink	RS/6000*	z/Architecture
e-business logo*	Language Environment*	S/390*	z/OS*
ECKD	MQSeries*	S/390 Parallel Enterprise Server	z/VM*
Enterprise Storage Server*	Multiprise*	System 370	z/VSE
Enterprise Systems Architecture/390*	On demand business logo	System 390*	zSeries*
ESCON*	OS/390*	System z	zSeries Entry License Charge
FICON*	Parallel Sysplex*	System z9	
GDDM*	Performance Toolkit for VM	Tivoli*	
	POWER5	Tivoli Storage Manager	
		TotalStorage*	

\* 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, Windows and Windows NT are registered trademarks 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.

Complete your session evaluations online at [www.SHARE.org/Orlando-Eval](http://www.SHARE.org/Orlando-Eval)

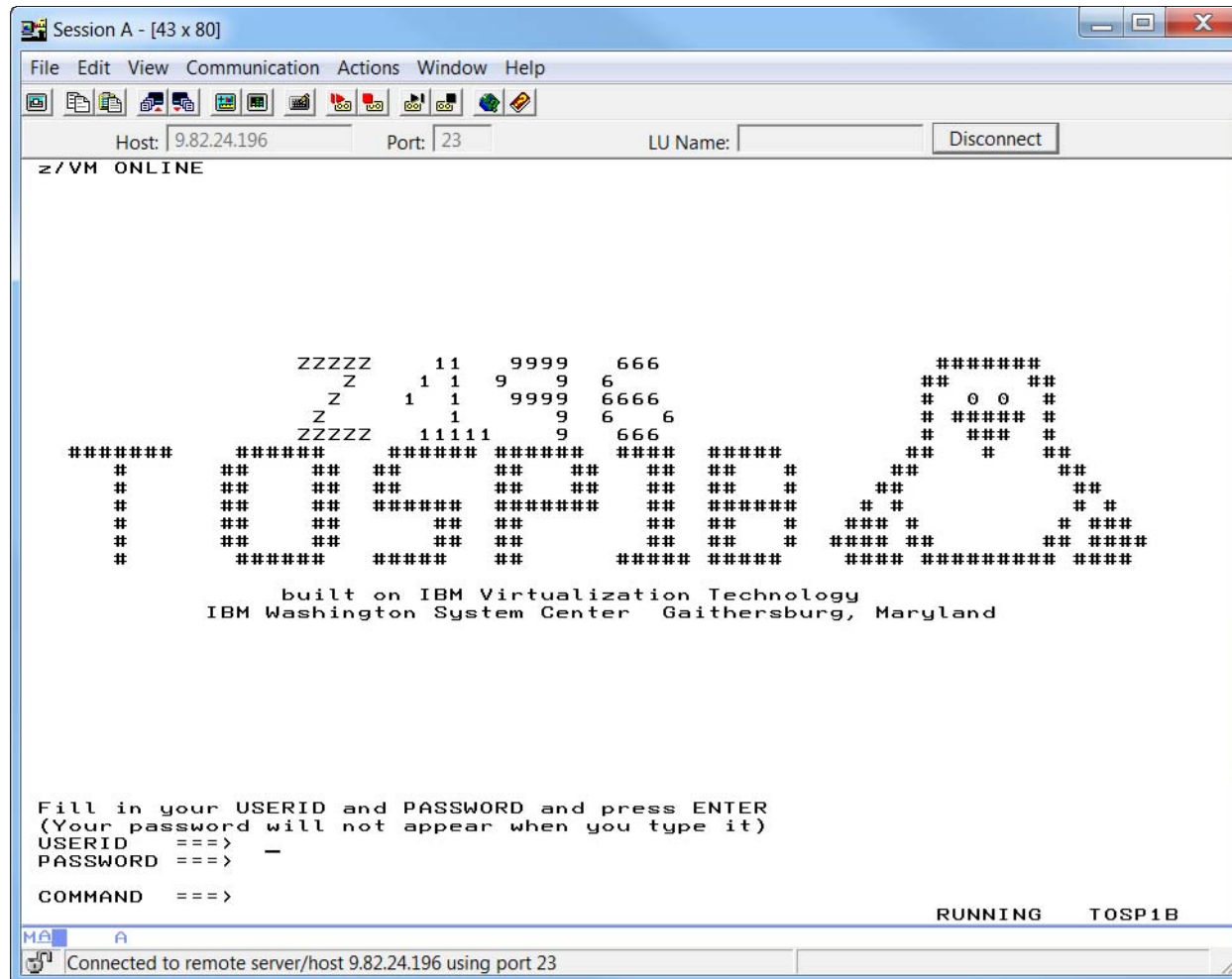
# Agenda

- Logon / disconnect / logoff
- How VM works, virtual machines, CP commands
- CMS operating system, CMS commands, file system, and access
- XEDIT
- Copying and erasing files, console spooling
- Help

# Logging On

- In order to log on to a virtual machine at a 3270 display, you must know:
  - The userid associated with the virtual machine
  - The password
- The logon process begins at the VM Logo screen, where you are prompted for the userid and password

# VM LOGO Screen

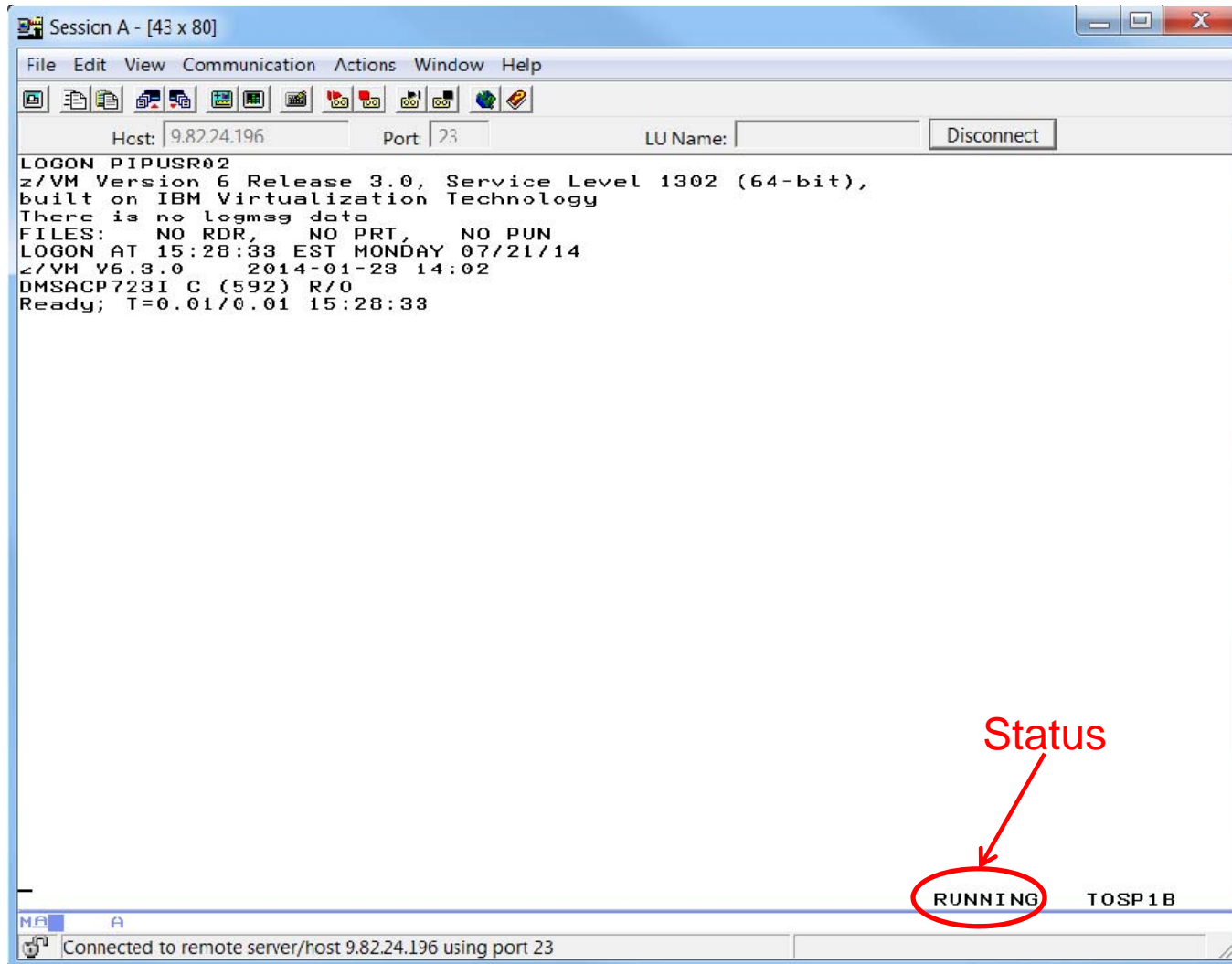


Complete your session evaluations online at [www.SHARE.org/Orlando-Eval](http://www.SHARE.org/Orlando-Eval)

## Logging On (continued)

- Once the userid and password are accepted and verified by VM (and the external security manager, if any), VM creates the virtual machine, displays the virtual console, and IPLs whatever device or operating system has been defined in the virtual machine's user directory definition.
- The virtual console will look something like the image on the following page for a CMS user. For other operating systems, it will contain the same startup messages you would see on the system console if it had been IPLed from a hardware console.

# LOGON PROCESS



Complete your session evaluations online at [www.SHARE.org/Orlando-Eval](http://www.SHARE.org/Orlando-Eval)

# Status Area

The messages in the status area that you may see are:

CP READ	CP issued a read request and is waiting for you to enter something before it continues processing.
VM READ	A response is required from you before any more processing can be done.
RUNNING	Your session is ready for you to enter your next command or is still processing the last command.
MORE...	The output display area is full and there are more lines to display. Clear the screen to continue. An automatic clear will occur in 60 seconds (can be changed)
HOLDING	The display filled and priority messages from CP are displayed or you pressed ENTER when the MORE... status indicator was displayed. Clear the screen.
NOT ACCEPTED	Previous input was not processed. CP locks the keyboard for about 3 seconds while it displays NOT ACCEPTED, then reverts to its previous status.



## Disconnecting

- Once a virtual machine has been logged on, you may want it to continue running without the virtual console connected to a display. The disconnect command will disassociate the virtual machine from a particular display, without stopping or resetting it.
  - To resume the connection between the virtual machine and a display, you must logon the virtual machine. This is called reconnecting. Sometimes you will see ‘CP READ’ at the bottom of the screen when the virtual machine reconnects. Enter ‘B’ or ‘BEGIN’ to continue. The virtual machine does not have to reconnect to the same display it disconnected from.

## Logging Off

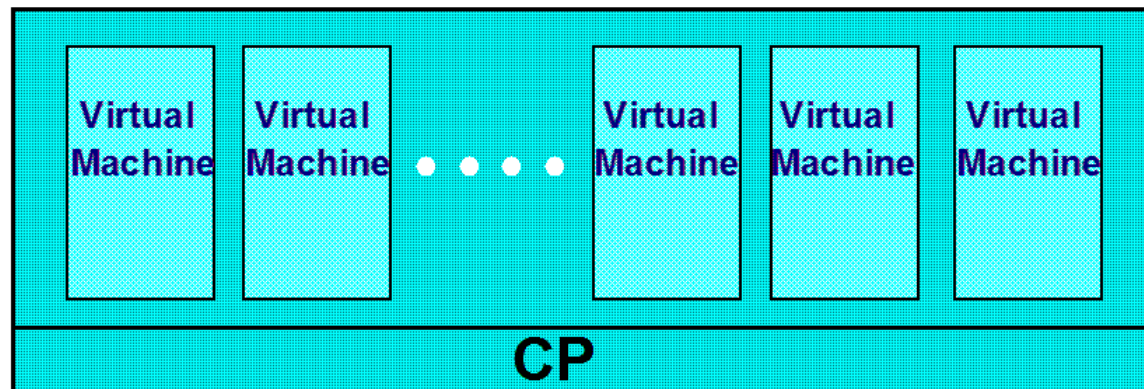
- When you no longer have work for a virtual machine to perform, you log it off using the LOGOFF command. When you log off:
  - The virtual machine is deleted. Everything that was assigned to it (virtual storage, devices, etc.) is released. Only the contents of permanent minidisks remain.
  - All applications, commands, or operating systems running in the virtual machine stop.

## Lab 1

- Logon to your virtual machine using userid and password:
  - PIPUSRxx and PIPUSRxx
- Type DISC (for disconnect) and press enter to get back to VM logo
- Logon again as above
- Type LOGOFF and press enter to get back to VM logo
- Logon again as above

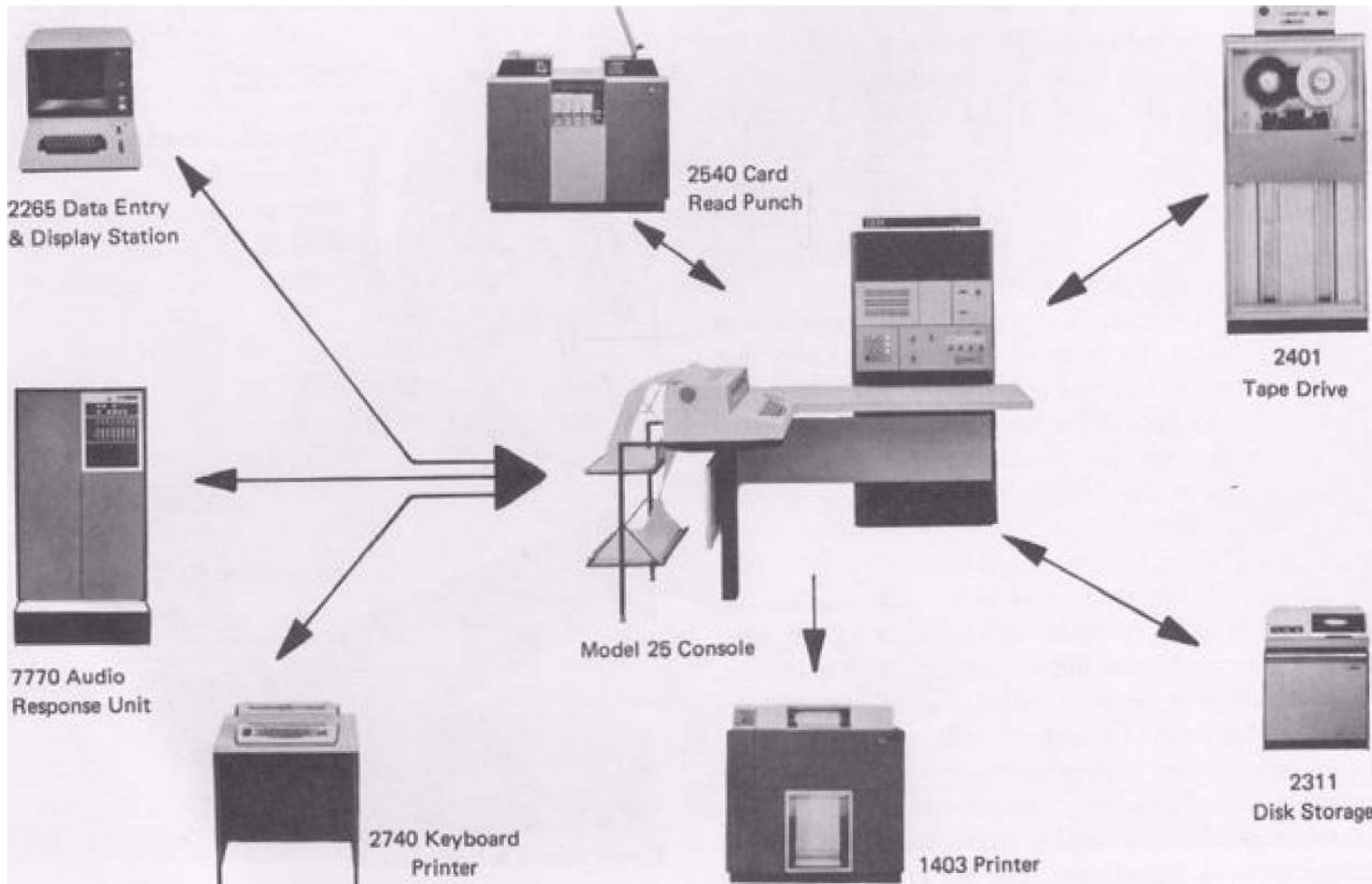
## How Does VM Work?

The VM operating system uses the real resources assigned to it to create and control “Virtual Machines”, control real devices, and simulate devices for the use of virtual machines.



The Control Program (CP) creates the virtual machines, distributes resources and control real hardware.

# Elements of a Real Computer System (of the 1960s)



Complete your session evaluations online at [www.SHARE.org/Orlando-Eval](http://www.SHARE.org/Orlando-Eval)

# The Virtual Machine

- A virtual machine
  - Is a functional simulation of all of the elements of a real computer system
  - Is identified by a Userid or Identity
  - Is created when logged on
  - Initial configuration is defined in the User Directory
  - Configuration may be modified dynamically
  - Is completely isolated from
    - Other virtual machines
    - The Control Program

# Sample Single Configuration Virtual Machine Directory Entry

**USER** VMUSER1 VMPW1 128M 256M G

**MACHINE** ESA

**IPL** CMS

**CONSOLE** 009 3215

**SPOOL** 00C 2540 READER \*

**SPOOL** 00D 2540 PUNCH \*

**SPOOL** 00E 1403 A

**LINK** MAINT 0190 0190 RR

**LINK** MAINT 019D 019D RR

**LINK** MAINT 019E 019E RR

**LINK** TCPMAINT 592 592 RR

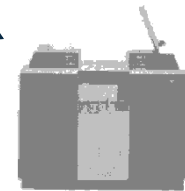
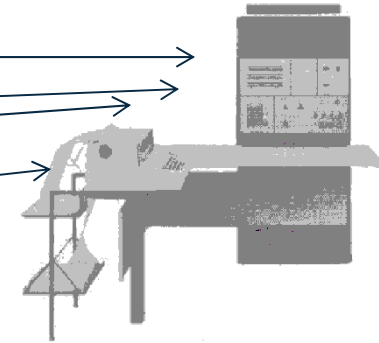
**MDISK** 191 3390 0400 0050 USR004 MR

**MDISK** 200 3390 0001 END VMG001 MR

**MDISK** 201 3390 0001 1500 VMG002 MR

**MDISK** 202 3390 0001 END VMG003 MR

**MDISK** 203 3390 0020 100 VMG004 MR



## CP Commands

- CP commands are used to communicate with CP to alter or query the state of the virtual machine or the VM system and all associated real and virtual devices
  - The CP commands a user is allowed to issue are restricted by the user's assigned privilege class
  - CP commands can be entered using any combination of upper and lower case. CMS will automatically translate console commands to upper case
  - You can communicate directly with CP, even when other operating systems are loaded in the virtual machine. Use the PA1 key to enter the CP environment and issue commands. You can also precede the command with a #CP, to bypass operating system processing



# The QUERY Command

- CP Query commands are used to display the status of the virtual machine, the system, and real and virtual devices. Responses to Query commands may differ based on the privilege class of the user. There are over 190 CP Query commands
- QUERY USERID – logon name
- QUERY VIRTUAL DASD – virtual machine disks
- QUERY PF – setting of program function keys
- QUERY TIME – current date/time and logon info
- QUERY NAMES – virtual machines running on system
- QUERY VIRTUAL ALL – see all virtual devices
- QUERY COMMANDS – see all CP commands authorized to execute

# CP Privilege Classes

- CP commands are divided into eight groups, each represented by a privilege class. The privilege class indicates the type of user from whom the system accepts commands.
  - The system programmer or system administrator who creates the user directory assigns each user one or more privilege classes as part of the user's entry in that directory.
  - The response to certain CP commands will change based on the class of user issuing the command.

# CP Privilege Class (continued)

Class	User and Function
A	System Operator: Controls the VM system. The system operator is responsible for the availability of the VM system and its resources
B	System Resource Operator: Controls all the real resources of the VM system, except those controlled by the system operator and spooling operator
C	System Programmer: Updates or changes system-wide parameters of the VM system
D	Spooling Operator: Controls spool files and the system's real reader, printer, and punch equipment allocated to spooling use.
E	System Analyst: Examines and saves system operation data in specified VM storage areas
F	Service Representative: Obtains and examines in detail data about input and output devices connected to the system
G	General User: Controls functions associated with a particular virtual machine.
Any	Commands belonging to this class are available to any user, regardless of the user's privilege class.

Complete your session evaluations online at [www.SHARE.org/Orlando-Eval](http://www.SHARE.org/Orlando-Eval)

## Lab 2

- Logon to your PIPUSRxx virtual machine if you are not already logged on
- Enter: QUERY USERID – to see your virtual machine name
- Enter: QUERY NAMES – to see all virtual machines running
- Enter: QUERY COMMANDS – to see all CP commands you can execute
- Enter QUERY VIRTUAL DASD – to see all disk devices available to your virtual machine
- Enter QUERY DISK – CMS command to show all minidisks currently accessed
- Enter QUERY ACCESSED – CMS command to show all SFS directories and minidisks currently accessed
- Enter QUERY VIRTUAL ALL – CP command to see all virtual devices associated with your virtual machine

# The DEFINE and SET Commands

- DEFINE and SET are used to change the configuration and settings of the virtual machine and (with the right privilege class) the z/VM system.
  - The actions of these commands only remain in effect until the virtual machine is logged off or the system is shutdown.
- DEFINE VFB-512 – create a virtual disk in storage
- DEFINE NIC – create a virtual network interface
- COUPLE – connect a virtual interface to a switch or lan
- DEFINE STORAGE – change memory size
- SET PF – assign command to PF key
- SET DATEFORMAT – change how date/time displayed
- DEFINE CPU – create another virtual processor

# Starting an Operating System

- Initial Program Load (IPL)
  - System z architected method to cause a program to be read from a designated device and to initiate execution of that program
- CP IPL command
  - CP command that simulates initial program load function for a virtual machine
  - Target device may be an attached tape drive, disk device, 2540 card reader, or named saved system
- Examples:
  - IPL CMS – performs ipl on named saved system cms
  - IPL 300 – performs IPL on disk device at address 300

## Lab 3

- Enter: query virtual storage – make note of response
- Enter: define storage 128m – at this point cms is no longer running as the virtual machine was reset
- Enter ipl cms – remember to press enter at the VM Read
- Enter query virtual storage – make note of new value
- Enter query cpus – make note of response
- Enter define cpu 01 – to add another virtual cpu to your virtual machine
- Enter query cpus – notice the new value now
- Function key 12 is already set as the command recall key
- Enter query pf12 – note response
- Enter query pf11 – note this pf key is not defined
- Enter set pf11 retrieve forward
- Enter query pf11 – note new value and experiment with pf12 and pf11 scrolling backward and forward through previous commands

# The CMS Operating System

- The Conversational Monitor System (CMS) is the part of the z/VM system that provides the interactive user environment, the file systems, the application programming platform, and various tools. To get the full use of z/VM, you need to know how to use CMS.
  - To start or restart CMS, use the CP IPL command:
    - IPL CMS – starts the Named Saved System named CMS
    - IPL 190 – loads the CMS operating system from device 190
  - There is no command to “shut down” CMS
    - Just logoff or enter an IPL command



# CMS Commands

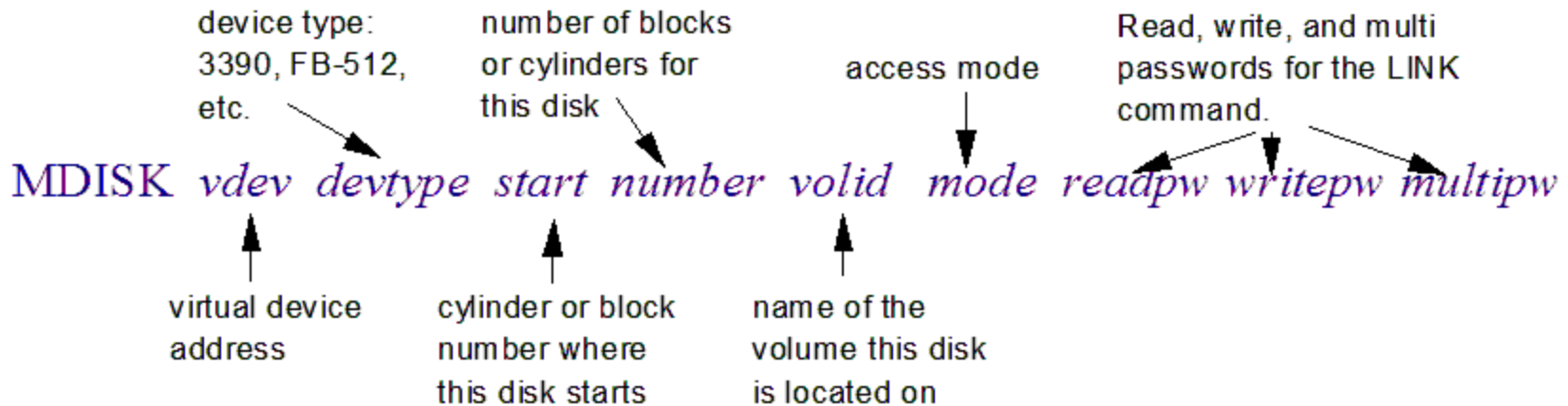
- CMS commands let you create, modify, and in general handle the CMS file system and its associated files for a virtual machine
  - CMS commands can be entered using any combination of upper and lower case. CMS will automatically convert all commands to upper case
  - Most CMS and CP commands can be abbreviated. When you see a command with mixed upper and lower case letters, the uppercase letters denote the shortest acceptable abbreviation (Examples: Access, Query, Set, DEFine, COPYfile, Help)

## What is a MiniDisk?

- A minidisk is a permanent, simulated, or temporary disk device available to the virtual machine.
  - Contiguous number of cylinders on a real 3390 or FBA device
  - A simulated 9336 device existing entirely in memory
  - Contiguous number of cylinders on a real 3390 or FBA device owned by CP, provided for temporary use.

# MDISK Directory Statement

The MDISK statement defines a minidisk (permanent, simulated, or temporary) for a virtual machine.

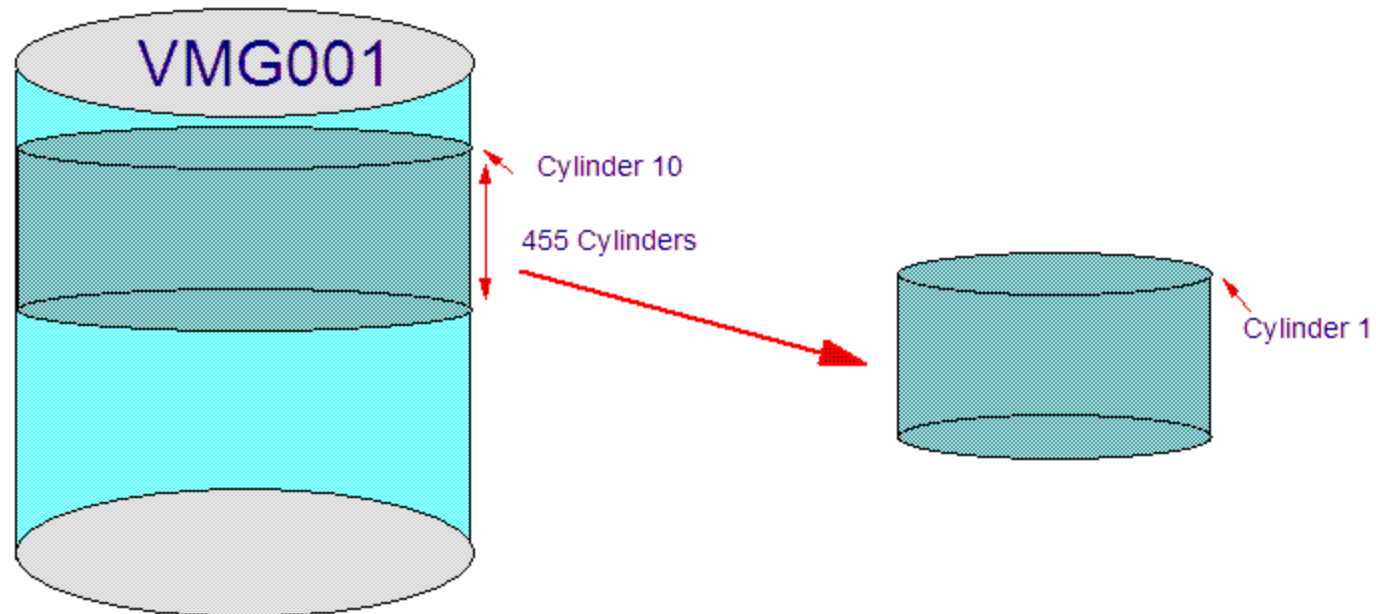


Example:

**MDISK 400 3390 0010 0455 VMG001 MR RREAD WWRITE MMULTI**

# MDISK Directory Statement Example

```
MDISK 400 3390 0010 0455 VMG001 MR RREAD VWRITE MMULTI
```



Real Device type: 3390  
Real Address: 174B  
Volume Label: VMG001  
Number of Cylinders: 3339

Virtual Device Type: 3390  
Virtual Address: 400  
Volume Label: **None**  
Number of Cylinders: 455

# CMS Minidisk File System

- The file is the essential unit of data in CMS. When you create a file in CMS, you name it using a file identifier (file ID) consisting of three fields:
  - File Name (fn)
  - File Type (ft)
  - File Mode (fm)
- When you use CMS commands and programs to modify, update, or refer to files, you must identify the file using these fields. Some CMS commands allow you to enter only the file name, or the file name and file type.

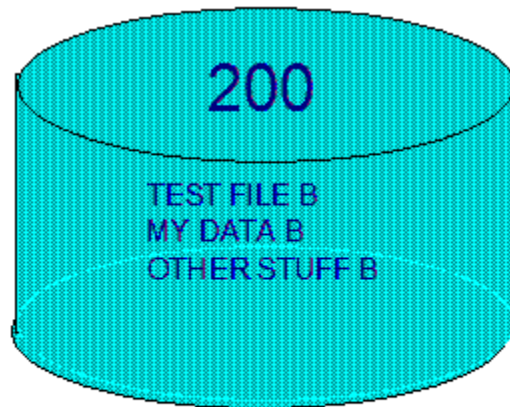
## The CMS File System (continued)

- The file name and file type can each be from 1 to 8 characters (A-Z, a-z, 0-9, \$, #, @, +, -, :, and \_)
- The file mode has 2 parts: the file mode letter and file mode number.
  - The file mode letter is an alphabetic character (A-Z) assigned to the minidisk where you want the file to reside. Minidisks are assigned a file mode when they are ACCESSed.
  - The file mode number is a number from 0 – 6. Each number has a special meaning to CMS. The default number is 1. It is used for reading and writing files.

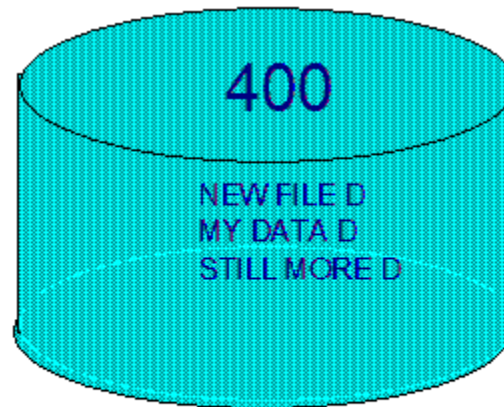
## ACCESS Command

- The Access command is used to identify a minidisk to CMS, make a list of the files on the minidisk available to your virtual machine, and associate a file mode letter with the minidisk. The minidisk must already be defined to the virtual machine through the user directory or the LINK command.
- Syntax: Access vaddr filemode
- Example: ACCESS 999 Z
- The CMS files on the minidisk located at virtual address 999 are now available and can be identified as filename filetype Z.

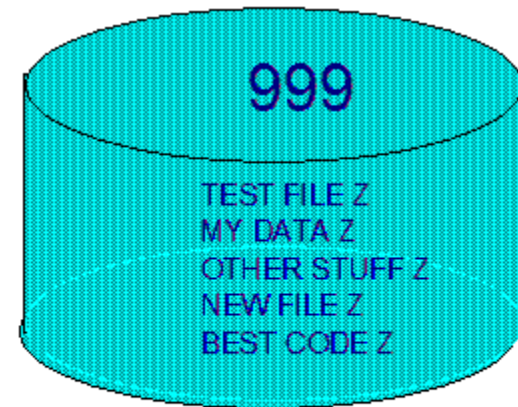
# ACCESS Example



ACCESS 200 B



ACCESS 400 D



ACCESS 999 Z



## Search Order

- The file mode letter specified on the Access command also establishes the CMS search order, file modes A through Z. This is used by CMS to search for commands or files.
  - No searching is performed for a fully specified file identifier: example TEST FILE Z
  - Searching is performed if the file mode is omitted or specified as \*: example TEST FILE \*
  - The search for commands only uses the CMS search order, in other words you cannot specify which disk a command is read from
    - An executable file on you're A disk will override a CMS command with the same name

## Listing Accessed Disks

- There are a few ways to acquire information about minidisks that have been accessed by CMS. These commands provide some common information in different formats:
  - QUERY DISK
  - QUERY ACCESSED
- Minidisks that exist in the virtual machine through the user directory or the link command, that have not been accessed will not be listed by these commands. The CP command QUERY VIRTUAL DASD will list all minidisks including unaccessed minidisks, but will not display CMS related information.

## Listing Files on Accessed Minidisks

- FILELIST – display a list of files on one or more CMS minidisks, and information about those files in an xedit context
  - The xedit context allows for commands to be executed against files
- LISTFILE – provides similar information to FILELIST but not in an xedit environment. Output is just sent to the console.

## Lab 4

- Enter: `filelist * exec *` - where the first \* represents any file name, then a filetype of EXEC, and the second \* represents any file mode
- Enter: `listfile * exec *` (date – notice the similarity of output to filelist without the xedit environment)
- Enter `query accessed` – notice sfs directory at mode A, and cms minidisks at other file modes

## Creating and Editing Files

- XEDIT is the VM system editor providing a powerful full-screen environment for creating and updating CMS files.
  - Subcommands allow you to move through the file, search for and replace text, and alter the editing environment
  - A profile can be used to alter the default XEDIT environment
  - User-written macros can issue XEDIT subcommands to perform editing, logic, and display functions
  - Invoked through the XEDIT command



# Editing with XEDIT

- Scrolling:
  - You can scroll through the file using predefined PF keys (PF8 = forward, PF7 = Backward) or XEDIT commands (Up nnn, Down nnn, Top, Bottom, :linenumber)
- Manipulating Lines:
  - Commands are available that allow you to add, change, delete, move and copy lines in a file. These commands are issued from the command line or the prefix area.
- Finding Character Strings in a File:
  - LOCATE and / (forward slash) can be used to locate a character string within the file
- Saving and Exiting XEDIT:
  - SAVE – saves the file to disk without exiting XEDIT
  - FILE – saves the file to disk and exits
  - QUIT – exit without saving the file

# Common Prefix Commands

- C – Copy a line
- M – Move a line
- I – Insert a line(s)
- A – Add a line (this is the same as I)
- D – Delete a line(s)
- “ – Duplicate a line(s)
- For C, M, D, and “, a block of lines may be operated on by using CC, MM, DD, or “” on the first and last lines of the block
- P – For a copy or move operation, place the line(s) immediately preceding this line.
- F – For a copy or move operation, place the line(s) immediately following this line.



## Lab 5

- Enter: XEDIT TEST FILE A – to open an xedit session on empty new file named TEST FILE A
- Move cursor to command line and type: INPUT
- Type: This is a new – then press Enter key
- Type: file for this class – then press Enter key
- Type: I am adding new lines – then press Enter key twice
- Press PF7 to scroll to top of file
- Move cursor to prefix area on line 1 and type i then press Enter.
- Type test on the new blank line – then press Enter
- Type “ in prefix area on line 2 and press Enter
- Type d in prefix area on line 3 and press Enter - to remove second line of test
- Type “” in prefix area on line 1 and “” in prefix area on line 4 and press Enter – to duplicate block of lines
- Move cursor to the command line and type file then press Enter to save file contents and exit XEDIT.

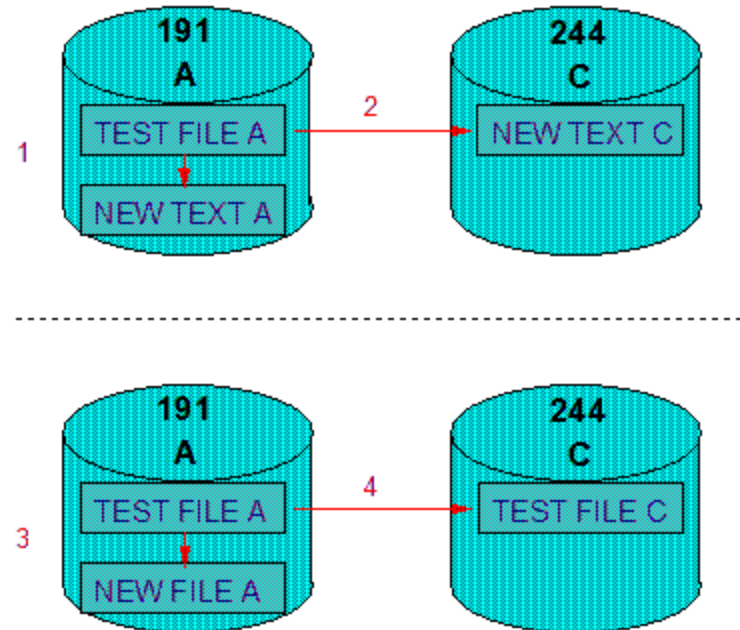
# Copying Files

**COPYFILE** will copy the contents of a file to a new file.

**COPYfile** *sourcefn sourceft sourcefm targetfn targetft targetfm*

Examples:

1. COPY test file a new text a
2. COPY test file a new text c
3. COPY test file a new = a
4. COPY test \* a = = c



# Erasing Files

ERASE will delete a file.

ERASE *fn ft fm*

Examples:

ERASE DATA FILE A

**Deletes: DATA FILE A**

ERASE DATA \* A

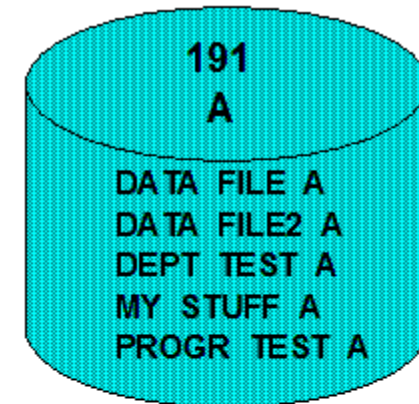
**Deletes: DATA FILE A, DATA FILE2 A**

ERASE \* TEST A

**Deletes: DEPT TEST A, PROGR TEST A**

ERASE DATA FILE \*

**Delete: DATA FILE A, and DATA FILE on any other accessed disk.**



# Working With Unit Record Devices

- QUERY {RDR, PRT, PUN}
- RDRLIST
- RECEIVE
- SENDFILE
- DISCARD
- PURGE
- PEEK

# Console Spooling

- To capture all line mode activity on the virtual console:
  - Enter spool console command
  - E.g. SPOOL CONS \* CONT START – will start spooling the console, and place the output in your virtual card reader when the console is spooled close
    - SPOOL CONS CLOSE will create a rdr file of spooled console output and open a new console spool file to continue capturing console output
- Use Unit Record Device commands listed earlier to manipulate the spooled console file

# Getting Help

- All VM components, commands, and utilities provide on-line help.
  - The Help function consists of files and menus that provide syntax and usage information for commands and tasks
  - Accessed through the CMS HELP command
    - HELP message – information relating to an error or informational message
    - HELP command – syntax and usage information for a command
    - HELP component – lists commands for the component (e.g. CMS, CP, XEDIT)
    - HELP TASKS – displays a menu of common tasks that can be selected. Each selection leads to another menu where you can again select a task, with each menu more specific until help for the correct command is displayed.

# HELP Menu

```

Session A - [43 x 80]
File Edit View Communication Actions Window Help
Host: 9.82.24.196 Port: 23 LU Name: Disconnect
HELP TASKS Task Help Information Line 1 of 40
(c) Copyright IBM Corporation 1990, 2013

z/VM HELP, main panel

The help panels listed below provide information about various
z/VM functions, topics, and tasks.
To view a help panel, move the cursor to any character of the name
or description and press the ENTER key or the PF1 key.

HELP - z/VM HELP Facility topics
MENUS - z/VM help menus
TASKS - Basic z/VM tasks
AVS - AVS commands
CMS - CMS commands
CP - CP commands
DIRMAINT - DirMaint commands
DIRMAINT - DirMaint topics
DUMPS - Dump commands, subcommands, and utilities
DYNIO - Dynamic I/O tasks
GLOSSARY - Definitions of terms
LE - Language Environment commands
LIBRARY - z/VM bibliography
MACROS - CMS assembler macros (menu)
MACROS - Assembler macros and functions (types)
MESSAGES - Messages and codes
OPEN - OpenExtensions services and APIs
PERFKIT - Performance Toolkit topics
PIPE - CMS Pipelines stages and subcommands
PIPE - CMS Pipelines tasks
QUERYSET - QUERY and SET commands and subcommands
ROUTINES - CMS callable services (menu)
ROUTINES - CMS routines (types)
RSCS - RSCS Networking commands and link parameters
STATMENTS - REXX, EXEC 2, and EXEC statements
SUBCMDS - Subcommand groups, such as XEDIT
TCP/IP - TCP/IP commands
PF1= Help 2= Top 3= Quit 4= Return 5= Clocate 6= ?
PF7= Backward 8= Forward 9= PFkeys 10= 11= 12= Cursor

====> - Macro-read 1 File
MA A
Connected to remote server/host 9.82.24.196 using port 23

```

## Lab 6

- Enter: spool cons \* cont start – this will start spooling your virtual console
- Enter: copyfile test file a = newfile a – this will copy contents of test file on mode a to test newfile on mode a
- Enter: listfile \* \* a – to see files on mode A
- Enter: erase test newfile a – this will remove the file just created (test newfile a)
- Enter: listfile \* \* a – to see files on mode A
- Enter: sp cons close – this will create a file in your virtual card rdr, please note the number
- Enter peek <the spool file number noticed above: do not enter the brackets> - this will open an xedit session on the console spool file
- Press PF3 to leave the xedit session on the console spool file
- Enter help cms menu – this will display a menu of cms commands. Move your cursor to any command name and press enter to see help for that command. Press PF3 to leave command help and return to the menu, Press PF3 to leave the menu and return to the CMS command line



# Summary

- A VM system is comprised of a collection of virtual machines
  - Nothing happens in a VM system outside the context of a virtual machine
  - The control program creates virtual machines and dispatches virtual processors on physical processors
- A virtual machine is a functional simulation of the real hardware it is running on
- CMS is an operating system included with VM that provides an end user interface to a VM system
  - CMS has it's own command set
  - Provides a file system for storing files
  - Provides an editor, XEDIT for creating and modifying files
  - Is a good environment for creating and deploying programs to manage and automate a VM system
  - Provides a help system for all VM components



Complete your session evaluations online at [www.SHARE.org/Orlando-Eval](http://www.SHARE.org/Orlando-Eval)