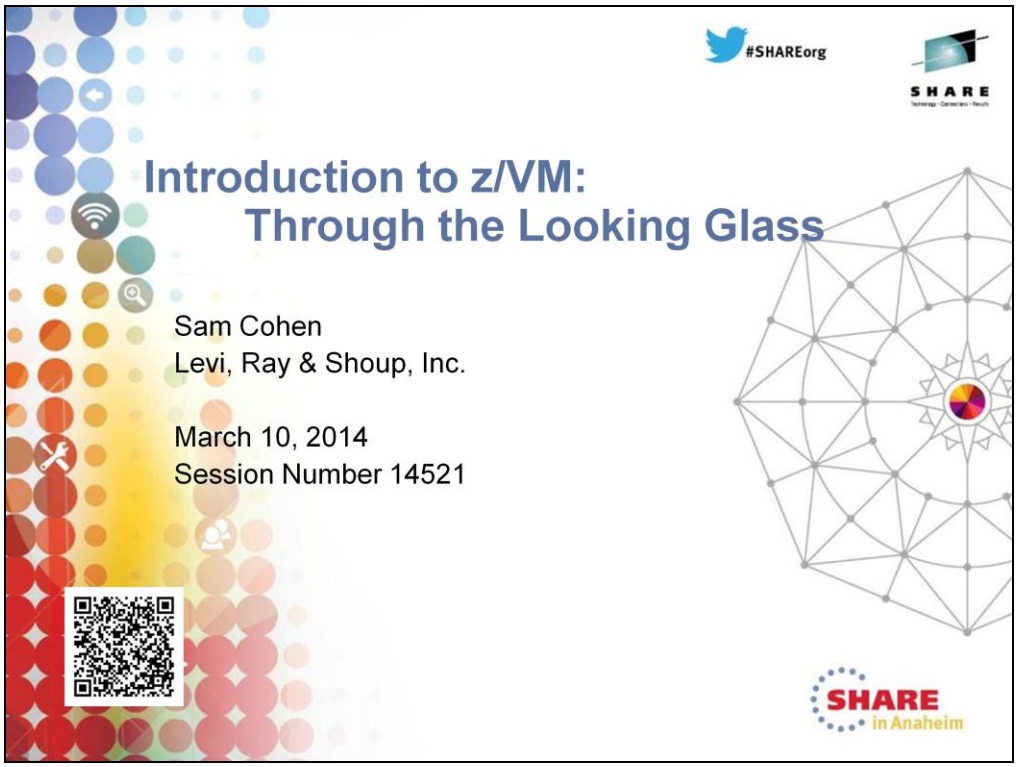
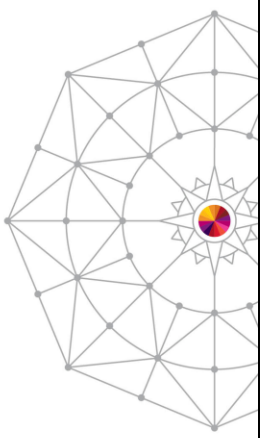


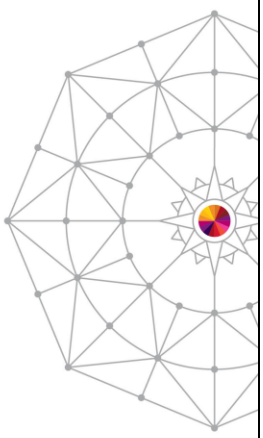
This presentation is designed to present an overview of z/VM to z/OS and Unix/Linux-oriented systems programmers. The presenter is assumed to be well versed in z/VM concepts and terminology, and should be able to understand terms and concepts of z/OS and Linux.

Agenda

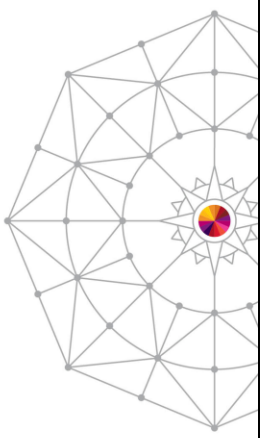
- Computing Resource Responsibilities
- z/VM Concepts
- z/VM Components
- CMS Filesystems
- Common CP/CMS Commands
- More Detailed Topics
- Linux Considerations under z/VM



Computing Resource Responsibilities



The Responsibilities of a Systems Programmer



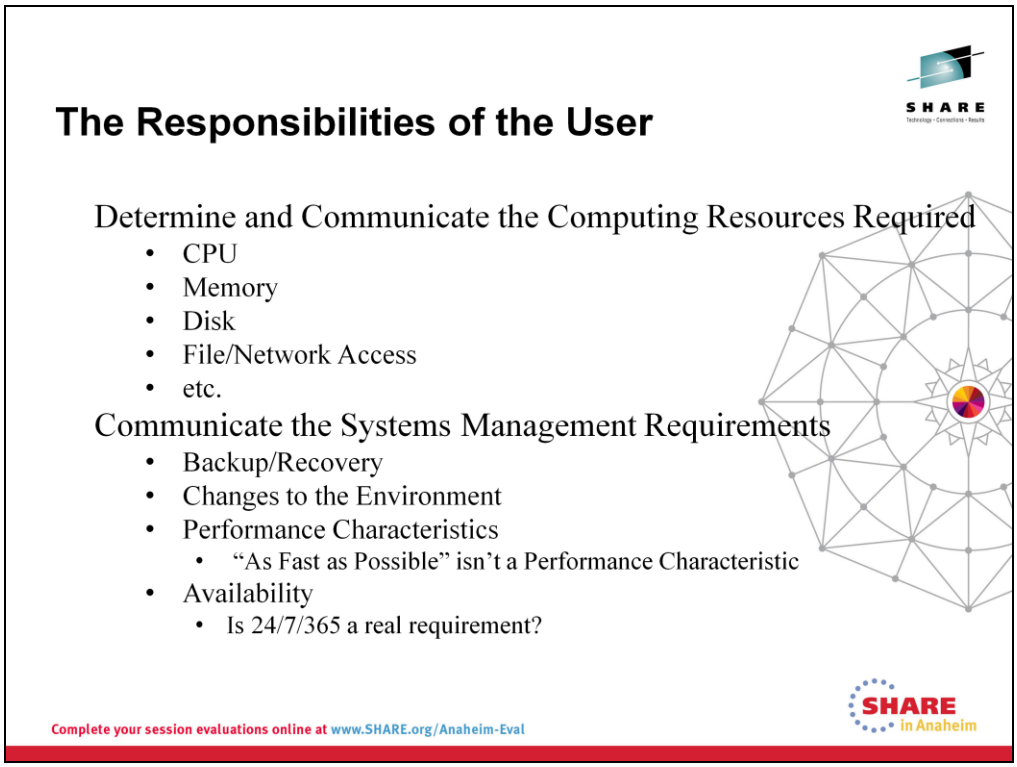
Allocation of Computing Resources

- CPU
- Memory
- Disk
- Network/File Access
- Etc

Management of Computing Resources

- Capacity Planning
- Performance Management
- Availability Management
- Problem Determination/Problem Source Identification (PD/PSI)
- others

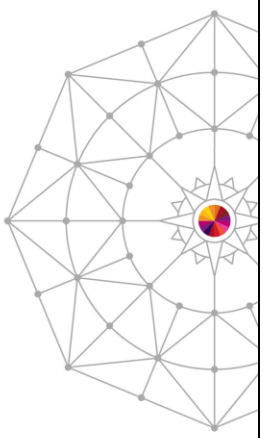
Let's consider the unique roles of the Systems Programmer vs. the Computing Resource End User. Basically, the Systems Programmer allocates and manages Computing Resources (CPU, Memory, Disk space, Network Access, etc)



This User, in this case, is not the end-user like a client running a web browse connecting to a server. This User is the content creator for the server. The End User of a Computing Resource not only creates the application programs, but must communicate to the Systems Programmer the amount of resources required.

One common difficulty for the User in the zSeries environment is the requirement to understand availability and performance requirements by the end-user community. These requirements are often not considered in the xSeries or pSeries environments until after the application has been developed.

z/VM Concept

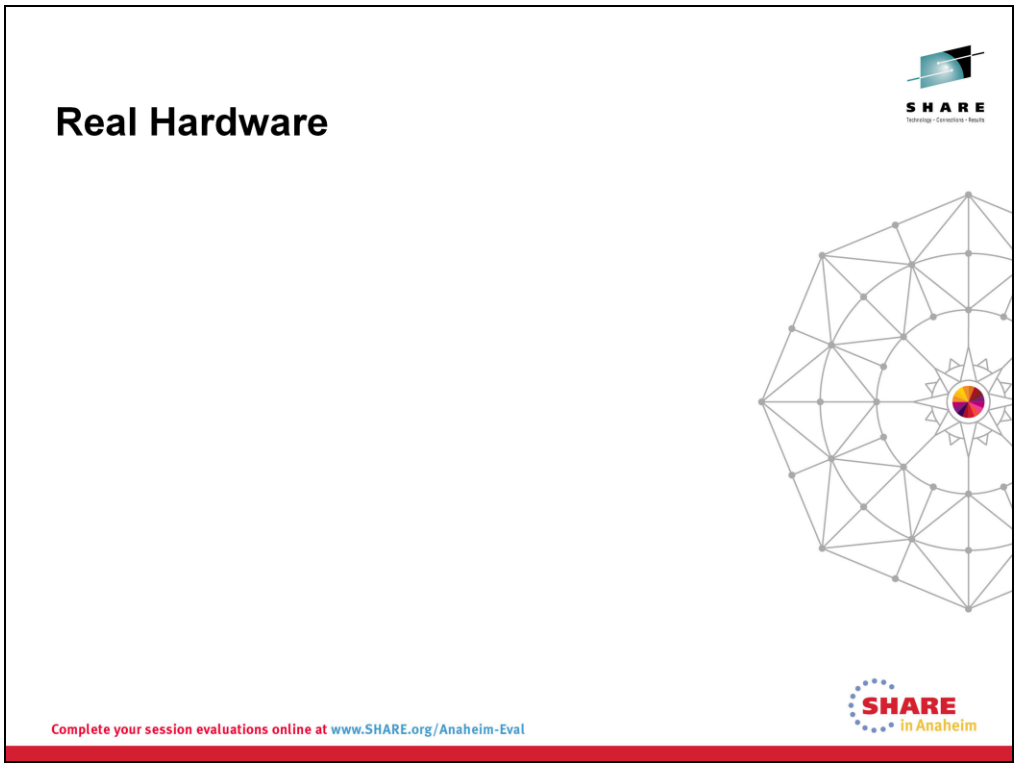


z/VM Concept

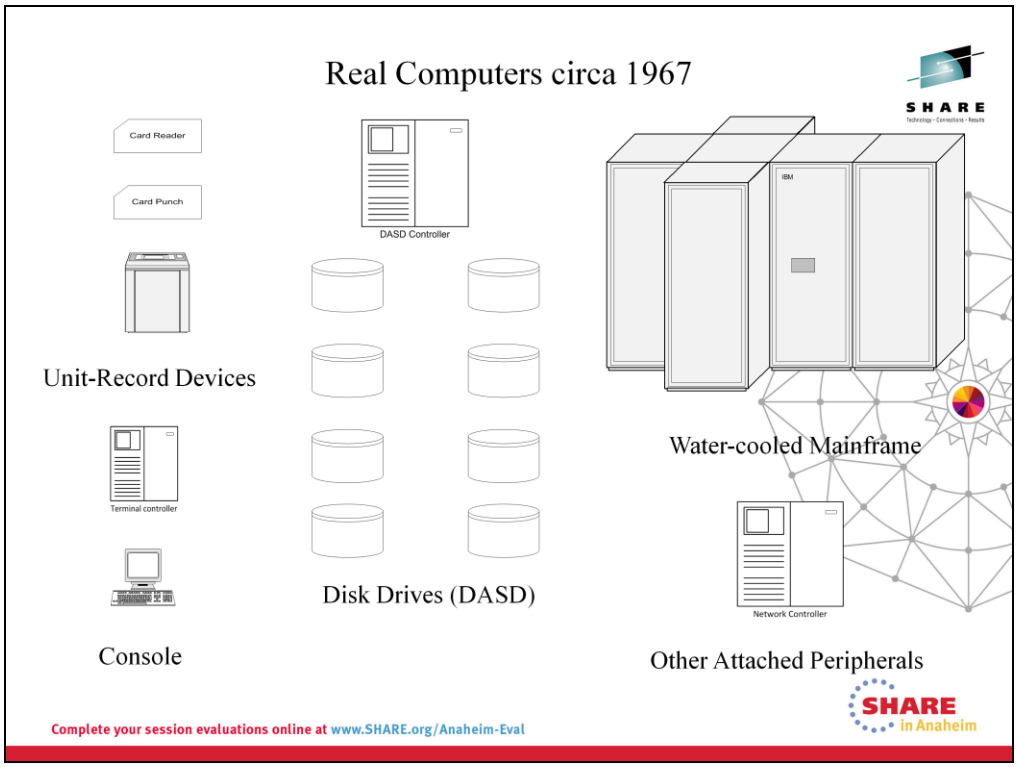
- z/VM Provides for the Definition and Management of Virtual Computers
  - Management of Real Hardware in support of Virtual Machines ("users" in z/VM terminology)
  - Definition of Virtualized Hardware
    - Allocation of Dedicated Real Hardware
    - Sharing of Real Hardware
    - Allocation of Dedicated Virtual Hardware
    - Sharing of Virtual Hardware
    - A Virtual Machine has only virtual hardware

Although a Virtual Machine uses real computer resources, it has no idea what the "real" world looks like, only what it can see. Everything in a virtual machine looks real to the guest operating system.

A Virtual Machine uses real hardware resources, but even with dedicated devices (like a tape drive), the virtual address of the tape drive may or may not be the same as the real address of the tape drive. Hence, a virtual machine only knows virtual hardware that may or may not exist in the real world.

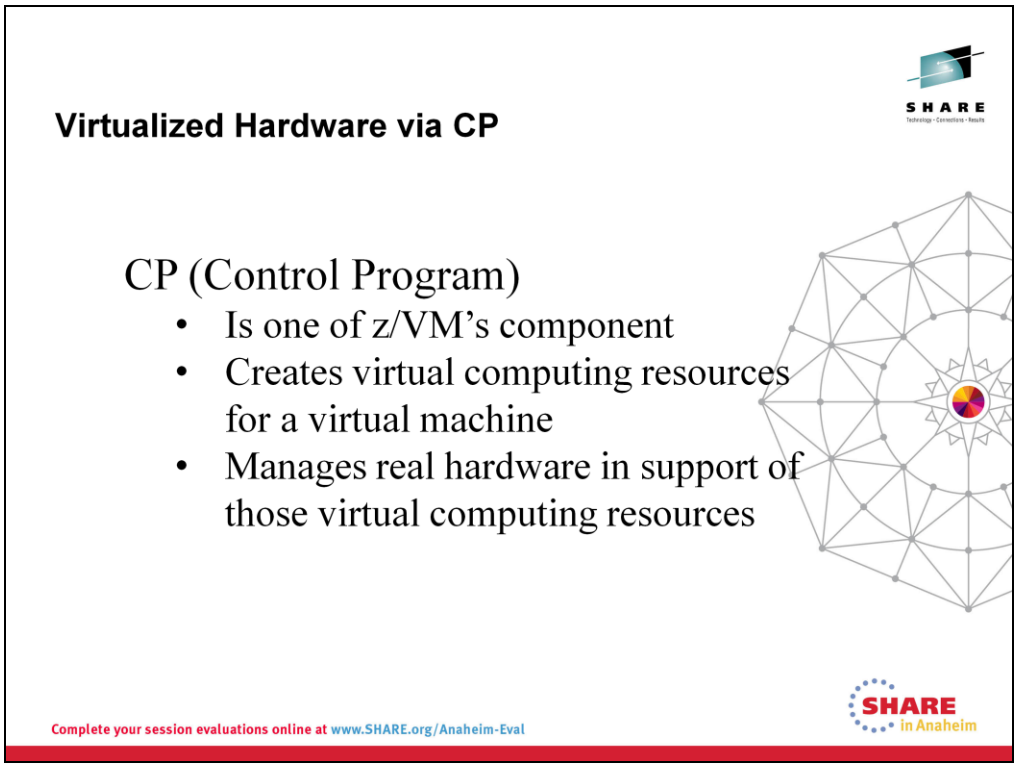


z/VM has its roots in a laboratory project that “escaped” and was known as CP67. Its paradigm, like that of the zSeries, harkens to computers and concepts from the S/360 and S/370 days.



So, what made up a computer in 1967? A computer had:

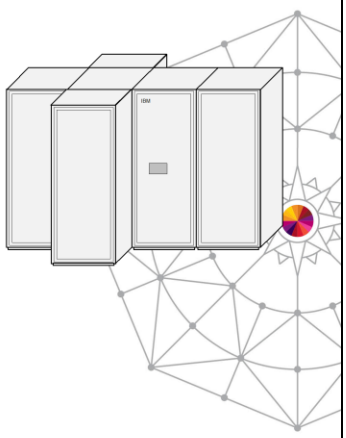
1) A Water-Cooled mainframe computer, with a certain amount of memory (also known as Central Storage)
2) Unit-Record Devices…card reader, card punch and line printer
3) A console for interfacing with the computer
4) Drum or disk drives (known as Direct Access Storage Devices or DASD)
5) Other devices attached via channel cables



CP is the main focus in this presentation. Its job is to manage hardware (real or virtual) and virtual machines.

Virtual CPU

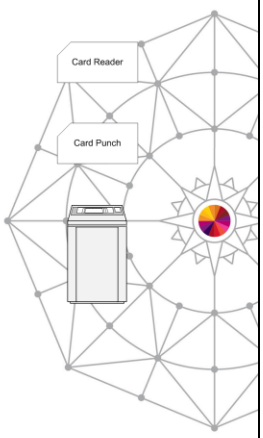
- CP can dedicate real CPU or share CPU between virtual machines
- CP can present more virtual CPUs than really exist
- Virtual CPU is not the same as Logical CPU
  - Physical CPU – the number of actual problem processors on the hardware
  - Logical CPU – the number of problem processors assigned to a Logical Partition
    - Cannot exceed the number of physical CPUs
  - Virtual CPU – the number of problem processors assigned to a Virtual Machine



This series of charts show how CP takes its knowledge of real hardware and allows creation of virtual hardware

Unit-Record Devices

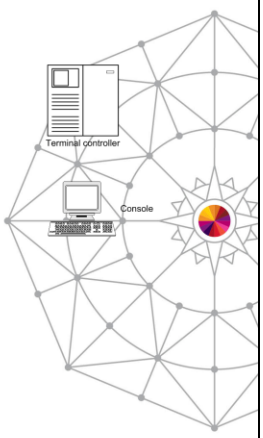
- Card Reader, Card Punch and Line Printer are presented to a virtual machine as Spooled Devices
- Real Unit-Record Devices (including tape drives) must be dedicated / attached (to a user or *system*)



Did you know that SPOOL is an acronym? Simultaneous Peripheral Operations On-Line

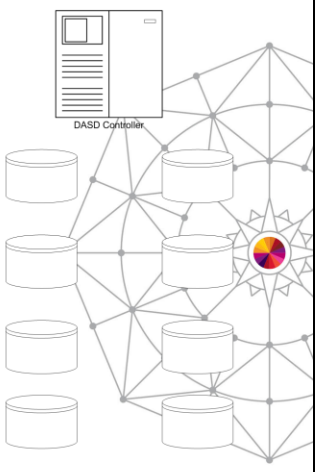
Console Device

- Virtual Console does not have to be physically connected to a real terminal
- If not connected to a real terminal, the virtual machine is considered to be “disconnected”



Disk Devices

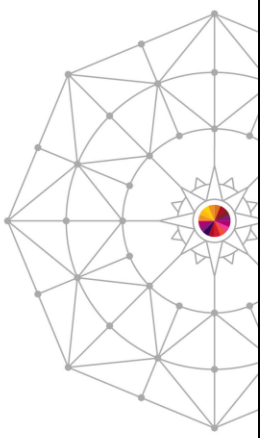
- Disk space can be contiguous subsets of physical disk (termed “minidisk”; similar to disk partitions in the x86 world)
- Disk space can also be “virtual storage” – carved out of memory (similar to a PC-DOS RAM Disk)
- Disk space can be on Count-Key-Data (CKD) or Open Systems (SCSI) devices
  - CKD devices have variable-block physical allocation
  - SCSI devices have fixed-block physical allocation
- Minidisks can be shared between multiple virtual machines
- A minidisk cannot span physical disks

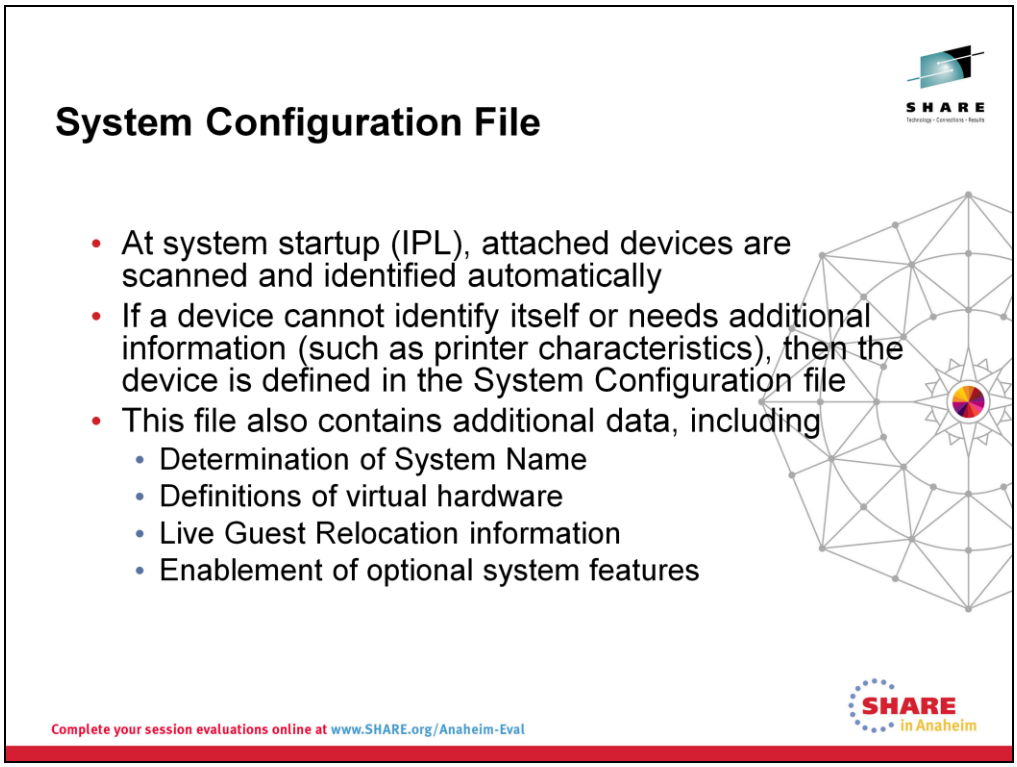


Other Physical and Virtual Devices

- Examples of other physical Peripherals:
  - Physical connections between LPARs/systems
  - FICON Directors
  - SAN Switches
- Virtual Peripherals
  - CTC (3088/ESCON/FICON), IUCV
  - 3270 and ASCII Terminals (GRAFs)
  - LANs (Hipersockets, QDIO Ethernet)
  - Coupling Facilities and Coupling Links
  - Network Switch (Layer 2 or Layer 3 with VLAN awareness)

System Definitions

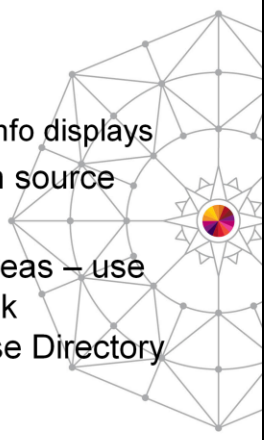


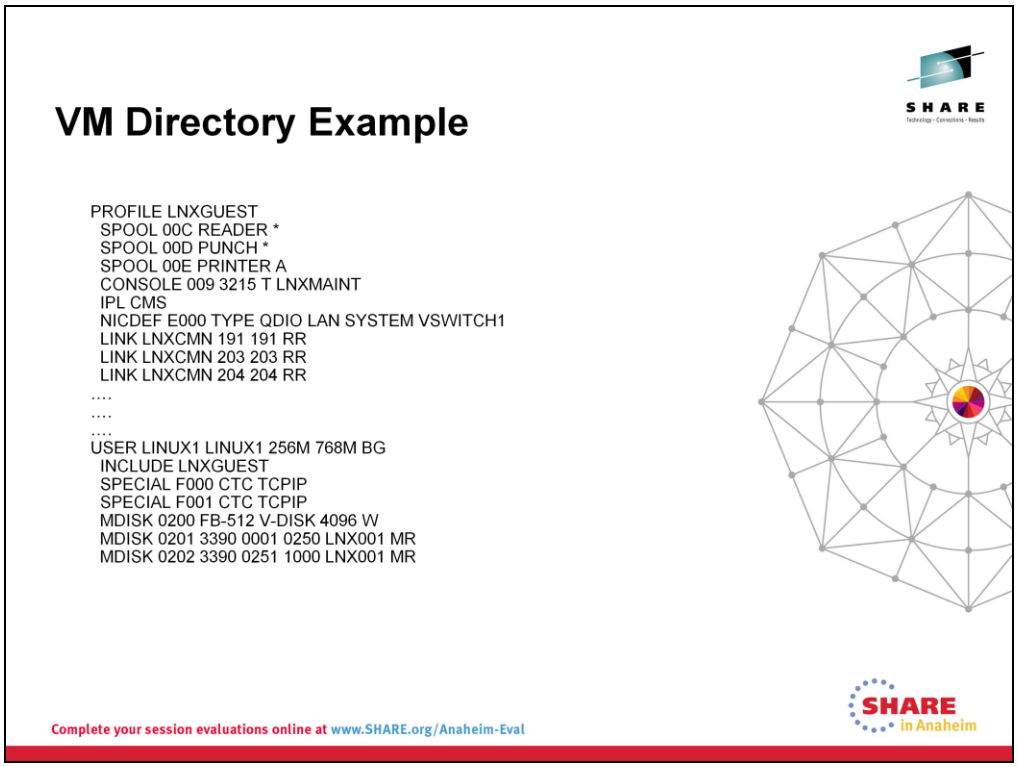


The System Configuration File will be referenced in a later chart when IPL of z/VM is discussed

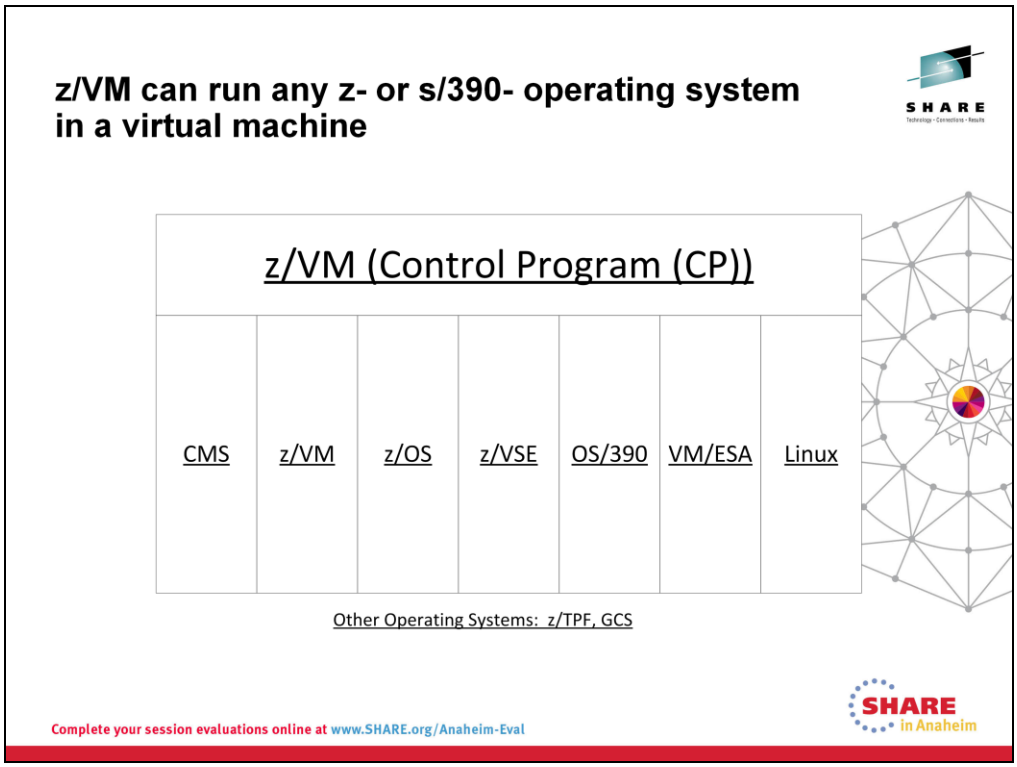
VM Directory

- Defines each virtual machine
  - Identifies initial resource allocations
  - Authorizes certain control commands and info displays
- Created by the DIRECTXA command from source code
- Doesn't care if you overlap disk storage areas – use DIRMAP/DISKMAP EXECs to map out disk allocations from Directory Source file or use Directory Management tool
- No disassembler!
  - Don't lose your source

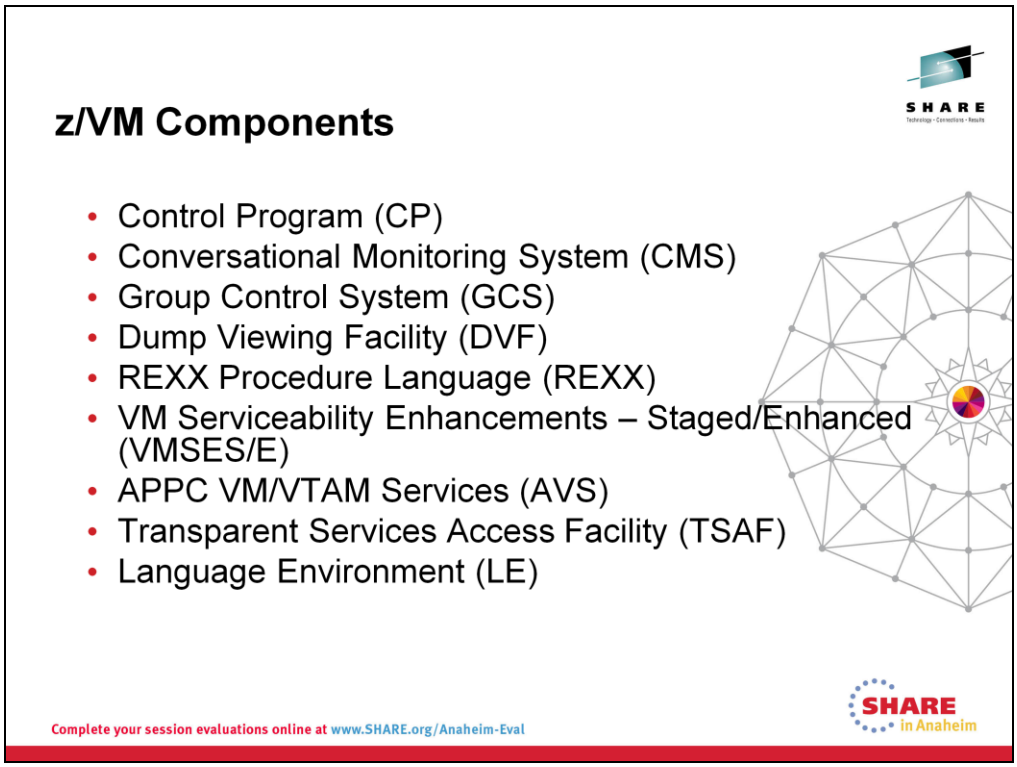




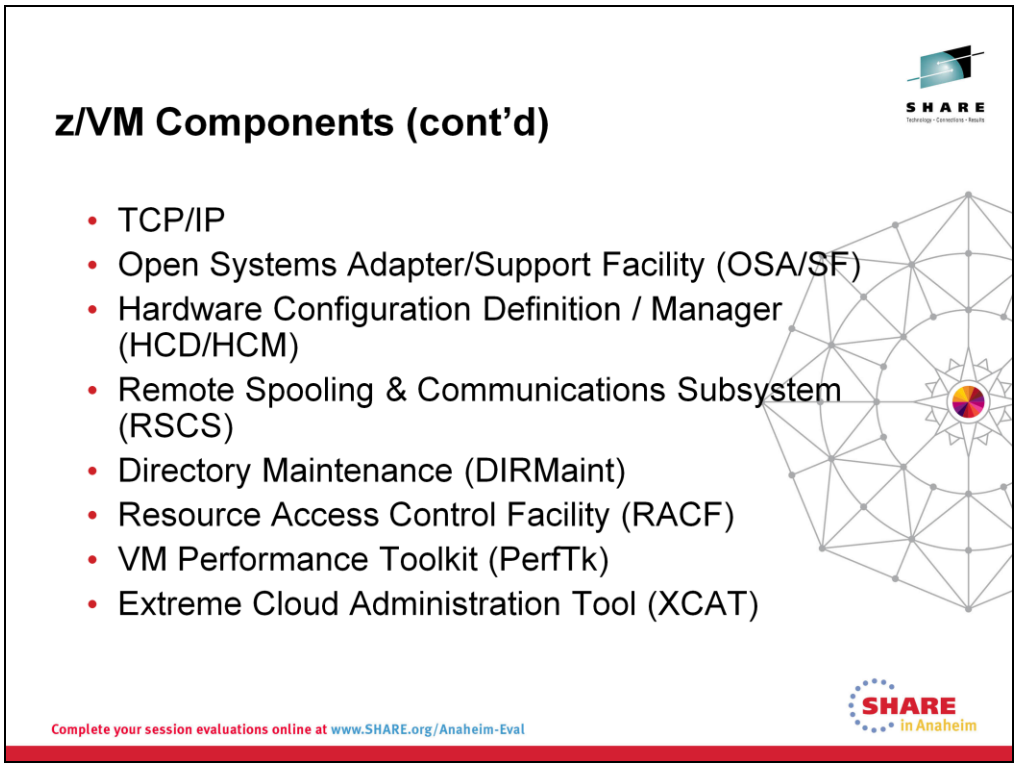
This example shows a virtual machine setup to run Linux using shared read-only minidisks. These minidisks (contained in the profile) contain executables (such as /boot and /usr) that can be shared between Linux instances and managed centrally.



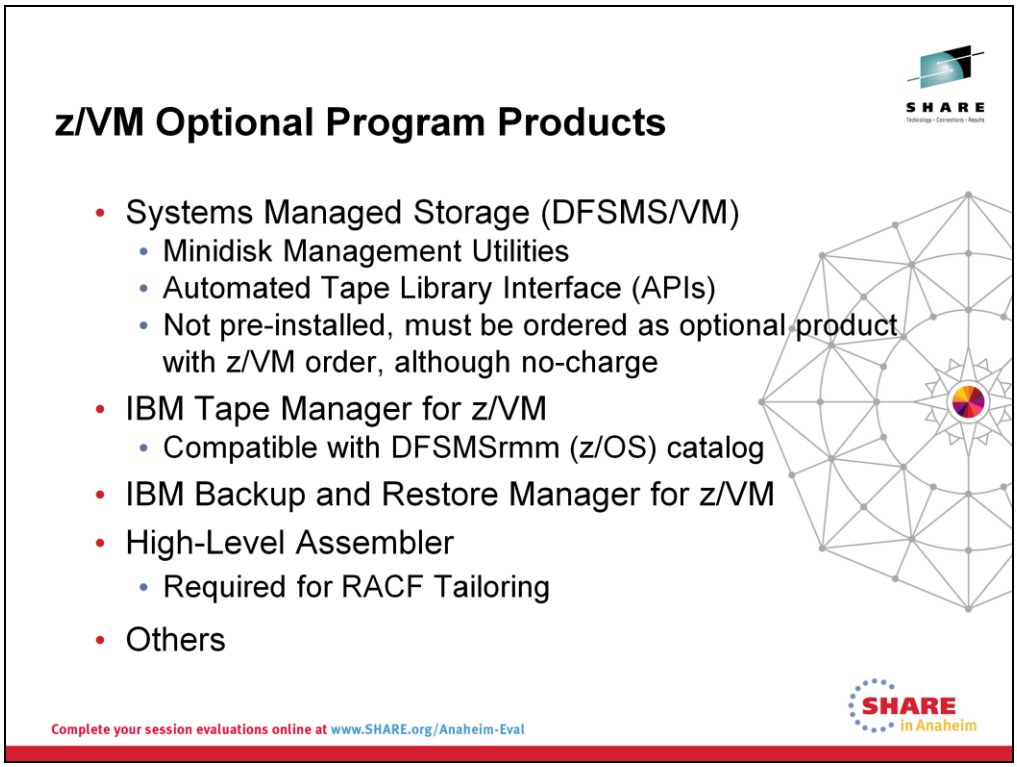
Note that you can run a complete copy of z/VM in a virtual machine. This is good for testing new releases of z/VM or CP modifications prior to putting them into production



We've talked about CP. CMS and GCS are operating systems that have a symbiotic relationship with CP and cannot run on their own. The other components are tools/middleware/libraries.

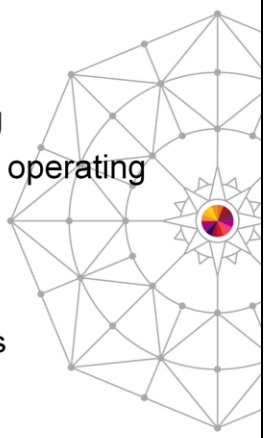


DFSMS/VM is not the same as DFSMS for z/OS. IODF output from z/VM's HCD/HCM is not compatible for use by z/OS guests or z/OS in other LPARs. Note that you do not need to use HCD/HCM…if your real hardware IOCDS is being managed from another LPAR, you can allow z/VM to be fully dynamic in its I/O recognition by using the defaults coded in the SYSTEM CONFIG file.



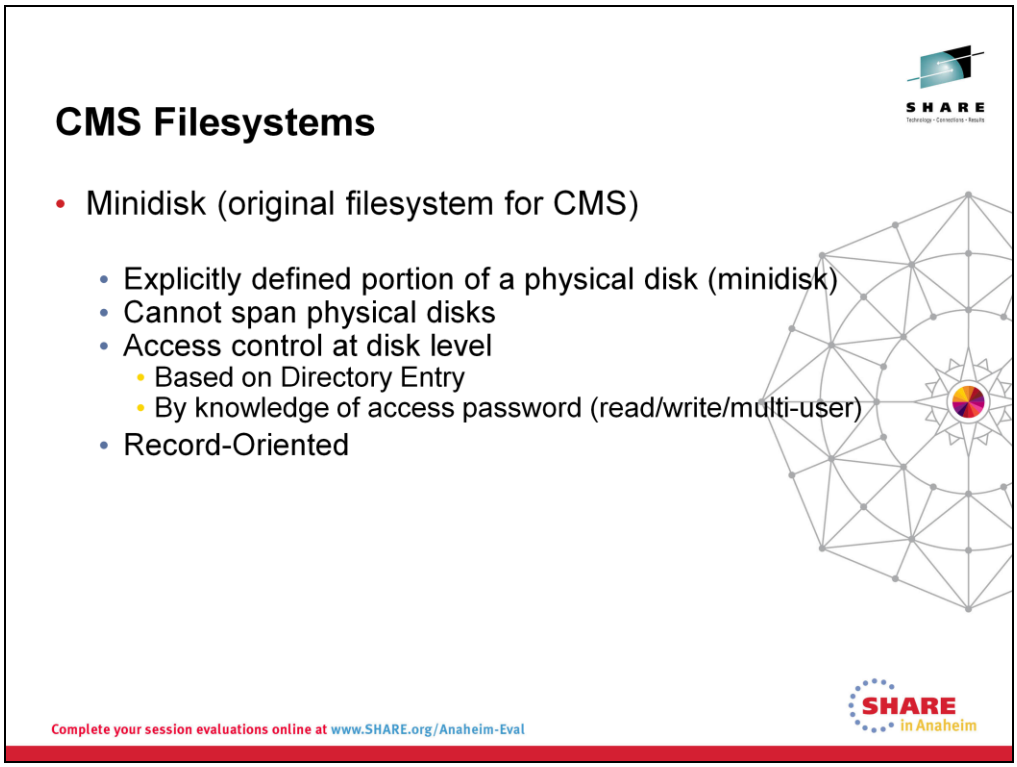
Additional Optional Program Products may be available; see http://www.vm.ibm.com

Conversational Monitoring System (CMS)



- Interactive Operating System for z/VM
- Originally single-tasking, now multi-tasking
- Often used to setup environment for guest operating systems (like Linux)
  - Define/establish communication paths
  - Change virtual machine settings
  - Define/configure additional virtual resources

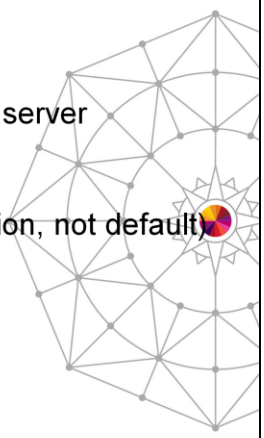
This set of charts discusses CMS and its filesystems



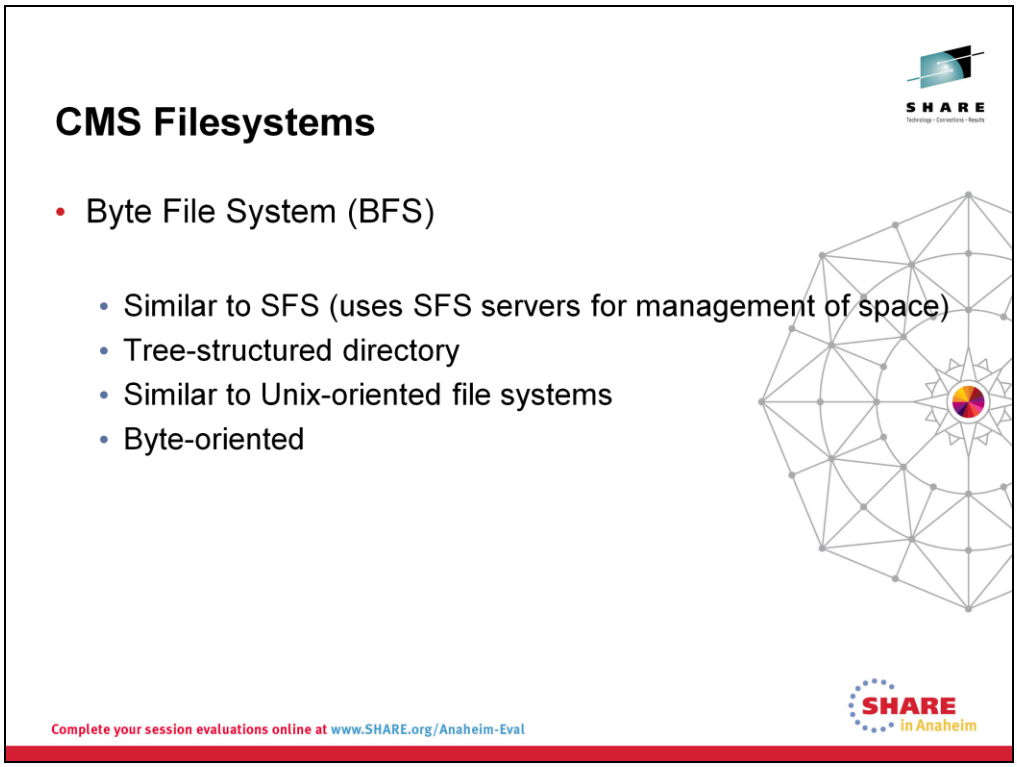
A minidisk formatted for CMS usage could be considered analogous to a PDS, but you don't need to compress it

CMS Filesystems

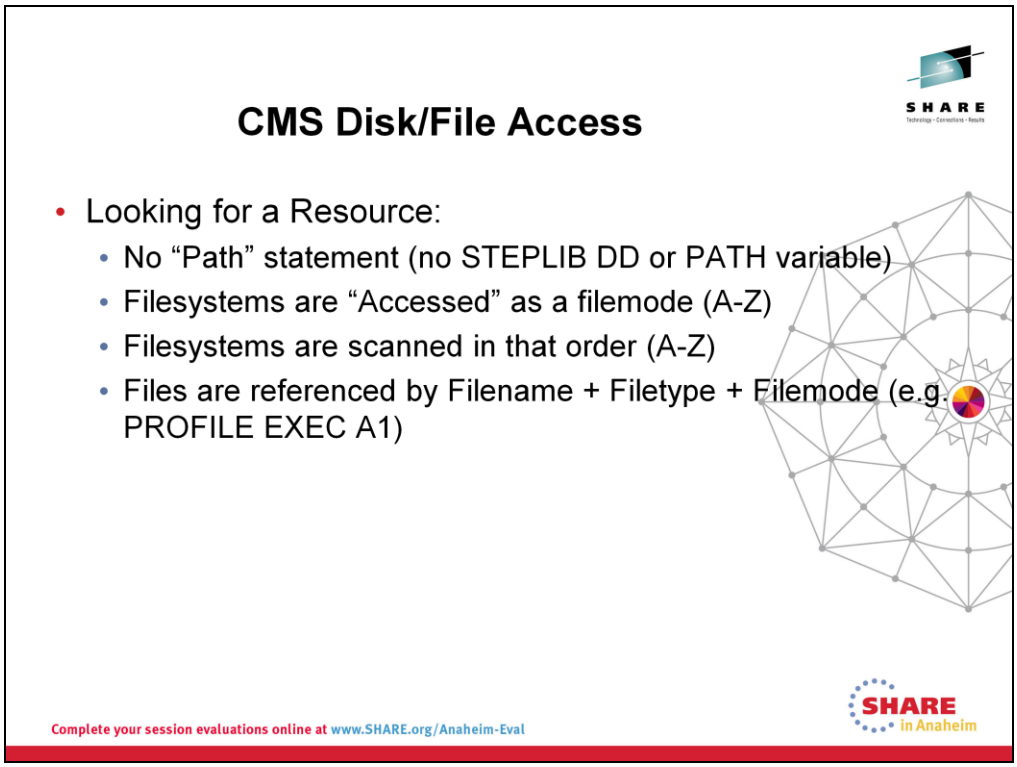
- Shared File System (SFS)
  - Filepools under control of Shared File System server
  - Can span physical disks
  - Can create tree-structure (subdirectories)
  - Access control at file level (disk level is an option, not default)
  - Record-Oriented



SFS provides better access control for shared files and more efficient use of disk space, since filepools are shared between authorized users. Users are assigned size limits (quotas in Linux terms) which can be changed on-the-fly by the filepool administrator



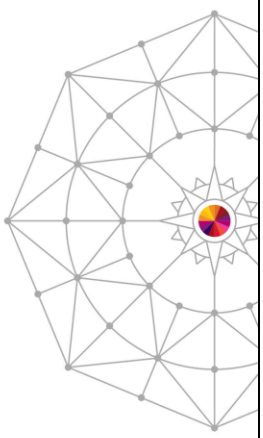
BFS is used by the z/VM Shell and Utilities, which provides some level of POSIX compliance for z/VM. The z/VM LDAP Server uses BFS

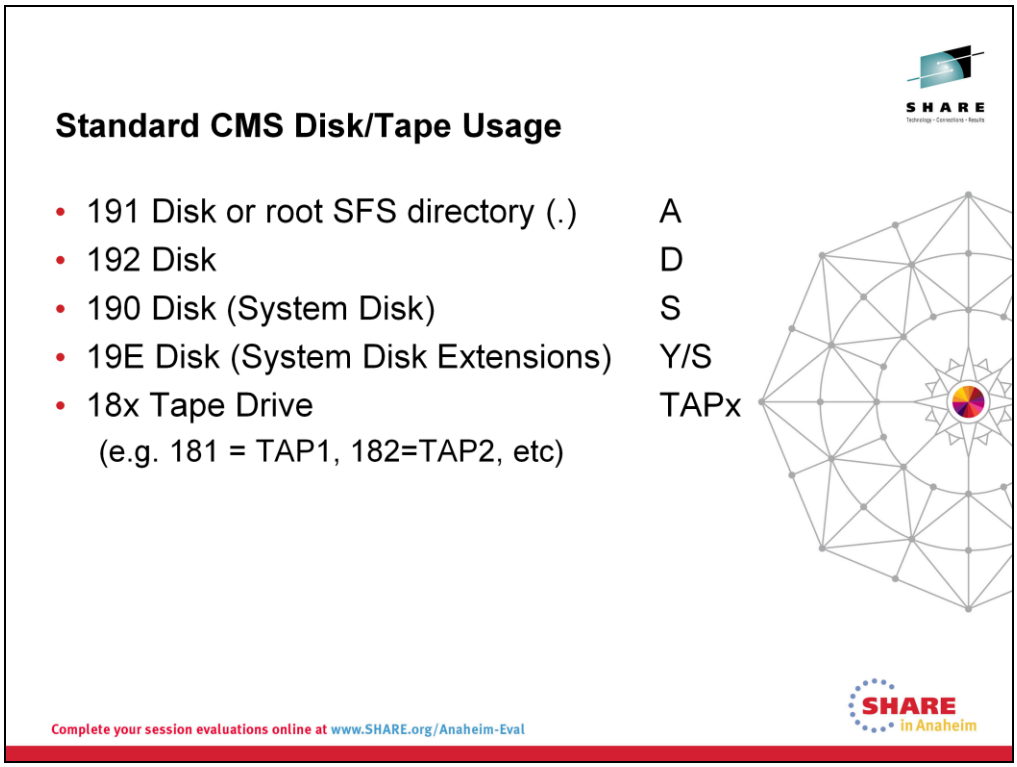


The first entity to be found during the search will be used; i.e. if you are looking for PROFILE EXEC * and this file exists on the B, D and M disks, the one on the B-disk will be used.

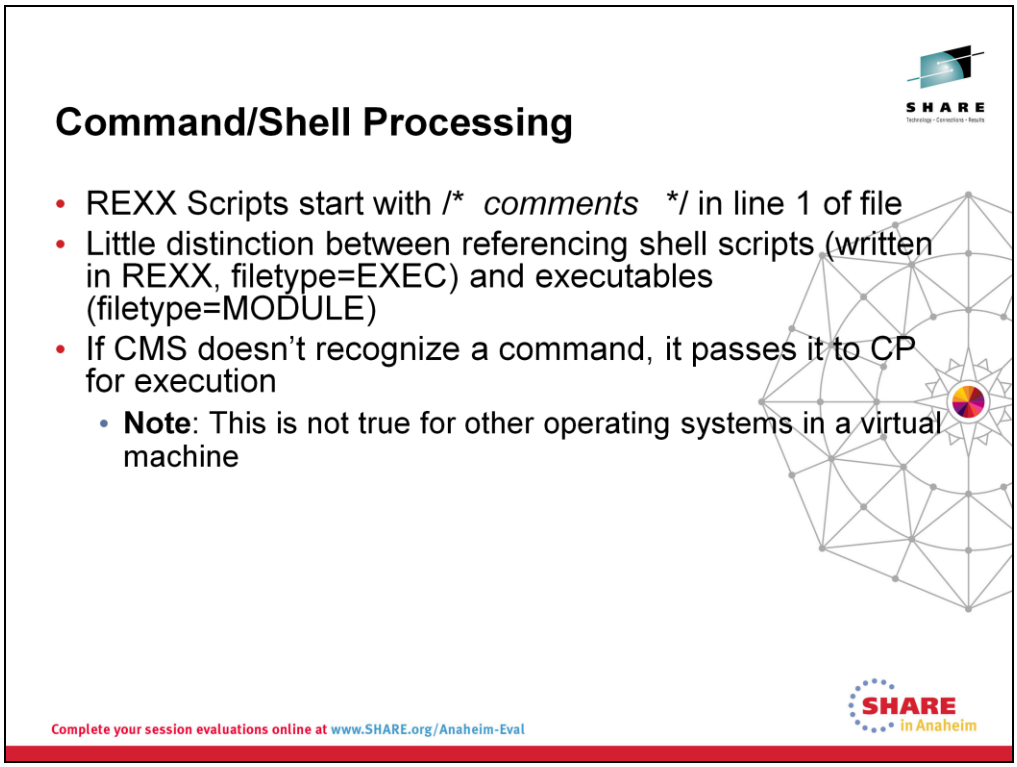
Some Reserved CMS Filetypes

- EXEC – REXX or EXEC2 (shell script)
- HELPxxxx – Help-Related File
- LISTING – Output (Print) File
- MACLIB – Macro Library
- MODULE – Executable Object
- NAMES – CMS Nicknames
- TEXT – Relocateable Object
- TXTLIB – Text Library
- XEDIT – Editor Control File(s)

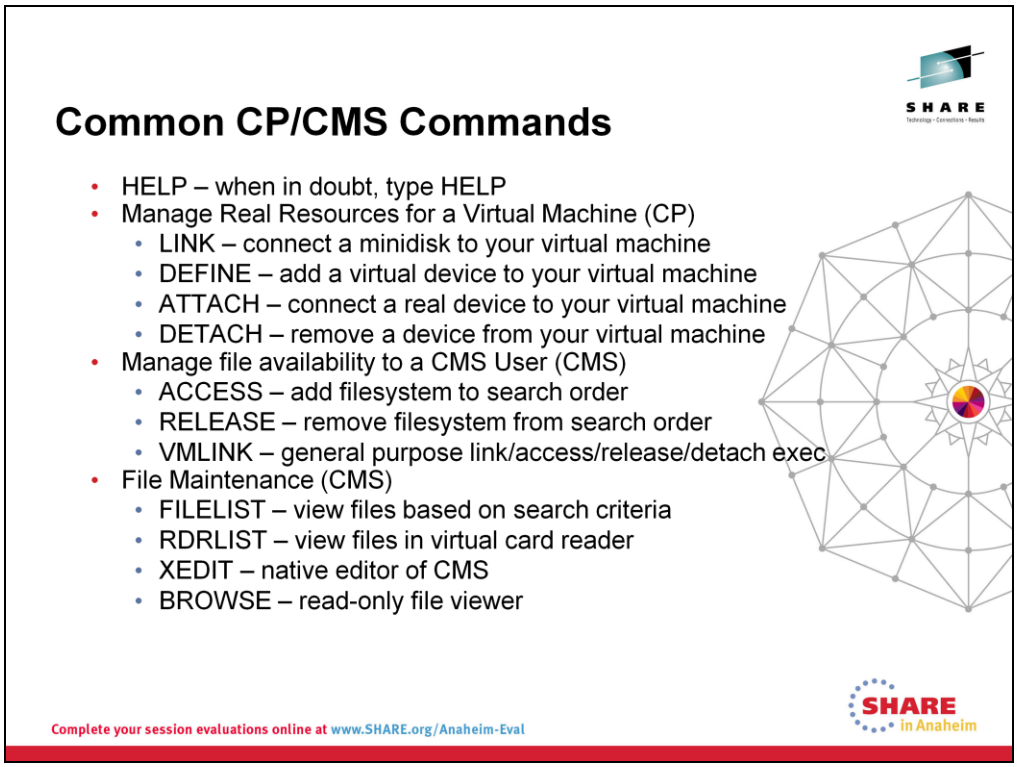




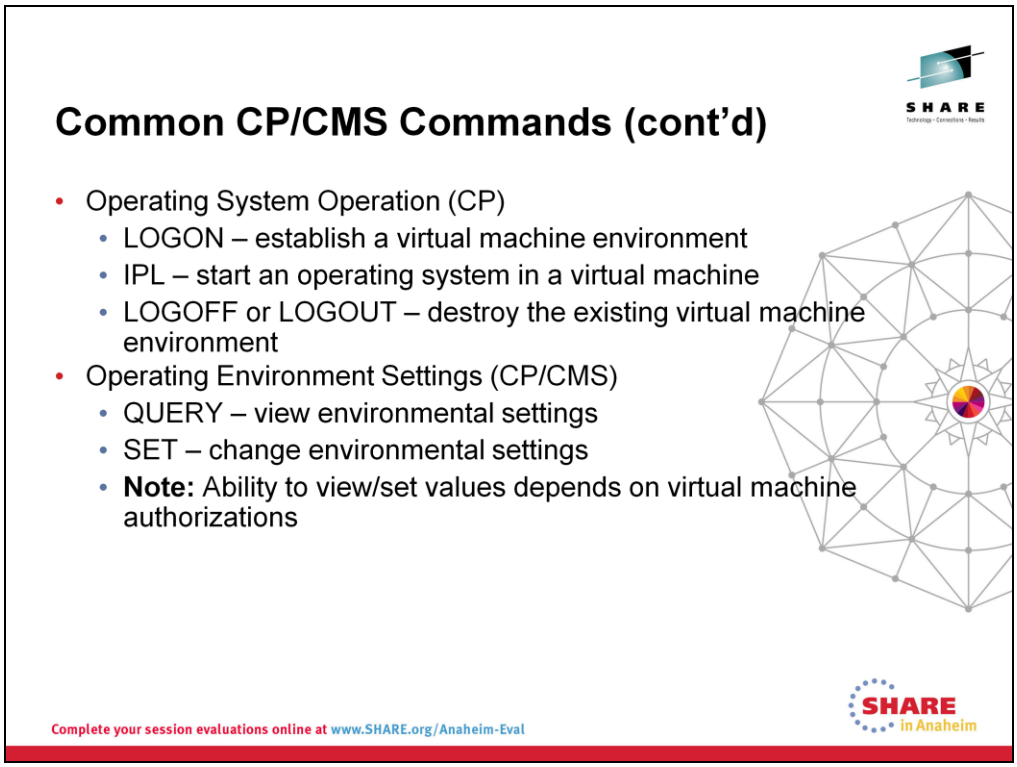
The Y/S indication means that the Y-disk is a read-only extension of the S-disk, and that the Y-disk will be searched right after the S-disk (instead of searching T, U, V, W and X disks before searching the Y-disk). The TAPx is the only use of an esoteric in CMS; there is no other concept of an esoteric is CMS.



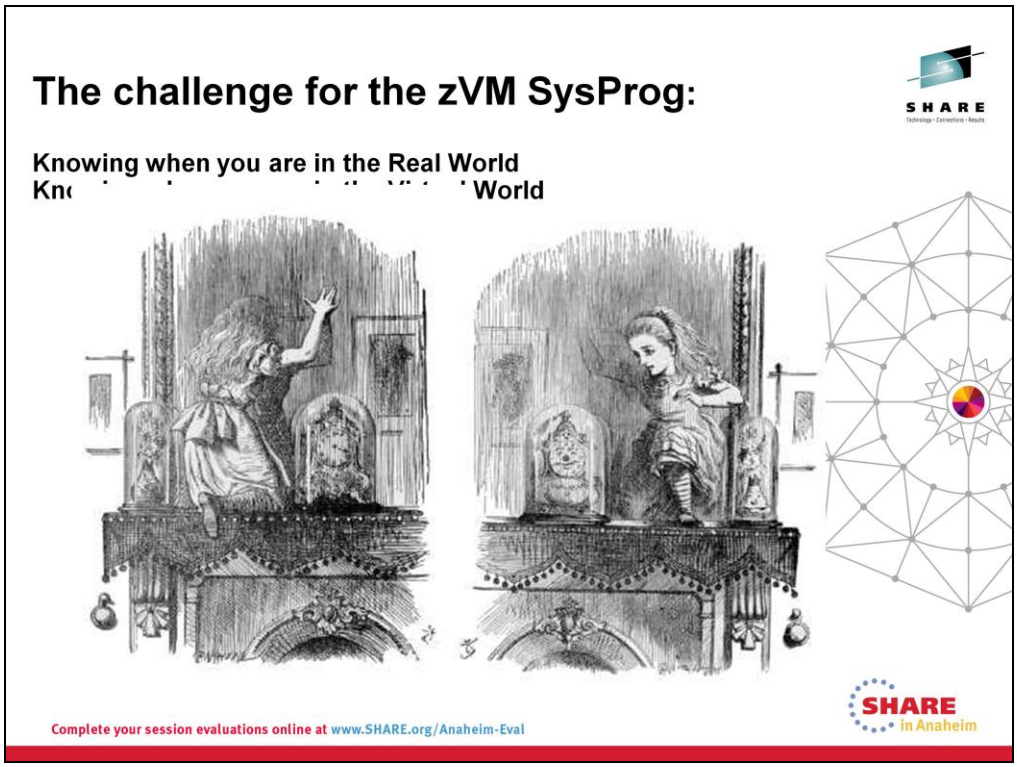
TSO requires the first line of a REXX proc to say "/* REXX */". This is unique to TSO REXX and is not required by any other REXX implementation



HELP contains messages and codes, command syntax and “how-to” info, all in one command. Note that some of the listed commands are CP commands, others are CMS commands



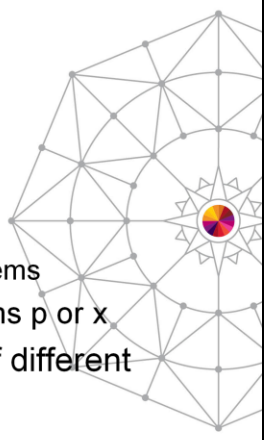
Remember that if you type LOGOUT or LOGOFF when CP is looking for a command (CP READ in the lower right corner of the console), then the virtual machine goes away. Linux virtual machine console users must be aware of their console mode (RUNNING, VM READ, CP READ, etc) before entering a command or the unexpected may occur.

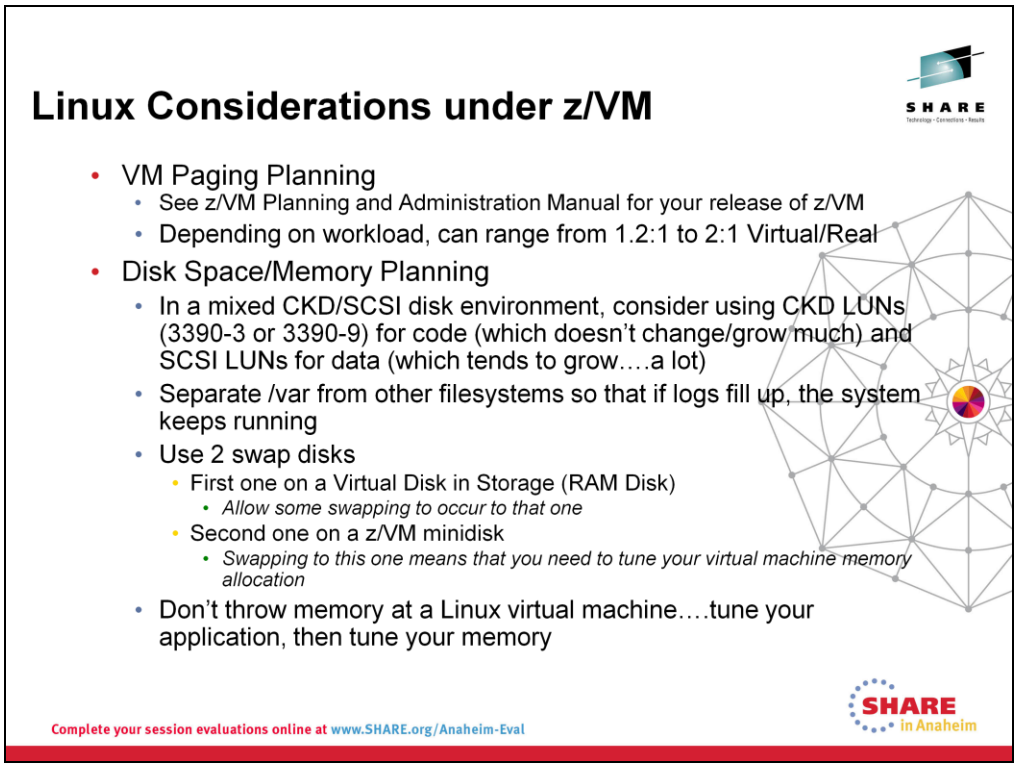


If you are a z/VM Systems Programmer, you must know which side of the "looking glass" you are on…the real side or the virtual side. You must also know when you are requesting information on the real environment and when you are requesting information on the virtual environment.

Challenges between traditional centralized and distributed computing groups

- Terminology
  - Disk/Memory/Tape vs. “Storage”
  - Disk vs. DASD
- Understanding of Hardware Capability
  - System z CPU design/structure
    - Assist Processors
    - Separate I/O Subsystems and other subsystems
  - Open Systems Adapter vs. NICs on Systems p or x
- Respect for Strengths and Weaknesses of different hardware platforms
  - No system is “perfect”

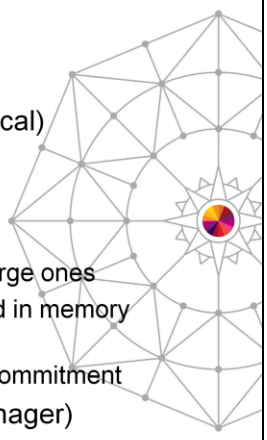




z/VM Paging changed a lot between 6.3 and earlier releases. Use of Expanded Storage was normal prior to z/VM 6.3. Virtual-to-real planning is very dependent on the type of application you are running….database servers need to have all their 'defined' (to the database) memory available, while application servers may be ok to allow some overcommittment of memory.

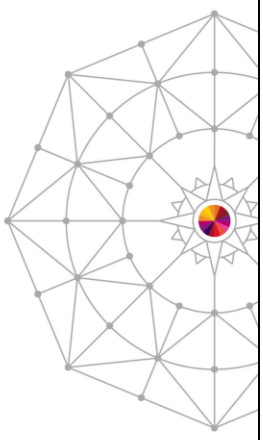
Linux Considerations under z/VM (cont'd)

- Don't use rules-of-thumb from other platforms
- Don't install everything available in a distribution
- Separate filesystems (see previous page)
  - Isolate filesystems that may grow (/home, /usr/local)
  - Don't let root filesystem (/) fill up
    - /var, /tmp sizes vary wildly
- Plan for ongoing tuning of the environment
  - Smaller memory footprints may perform better than large ones
  - Databases should be tuned to what they actually need in memory
    - Little to no memory over-commitment possible
  - Application servers are candidates for memory over-commitment
- Plan for filesystem growth (use Logical Volume Manager)
- Plan for managing code changes
  - Shared r/o minidisks may simplify this



Where to Get More Information

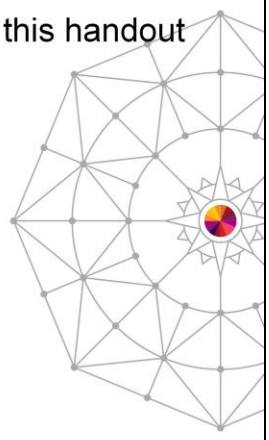
- SHARE sessions this week
- IBM Enterprise Technical University (usually 4Q)
- IBM z/VM Collection Kit (SK2T-2067)
  - Orderable at http://www.ibm.com/publications
- IBM z/VM Web site
  - http://www.vm.ibm.com
- IBM RedBooks
  - http://www.redbooks.ibm.com
- LinuxVM
  - http://www.linuxvm.org



Complete your session evaluations online at www.SHARE.org/Anaheim-Eval

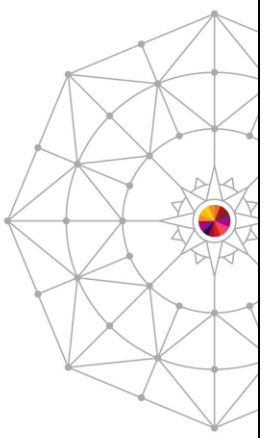
Survey

- Please use the following QR code to access this handout and complete the online evaluation:

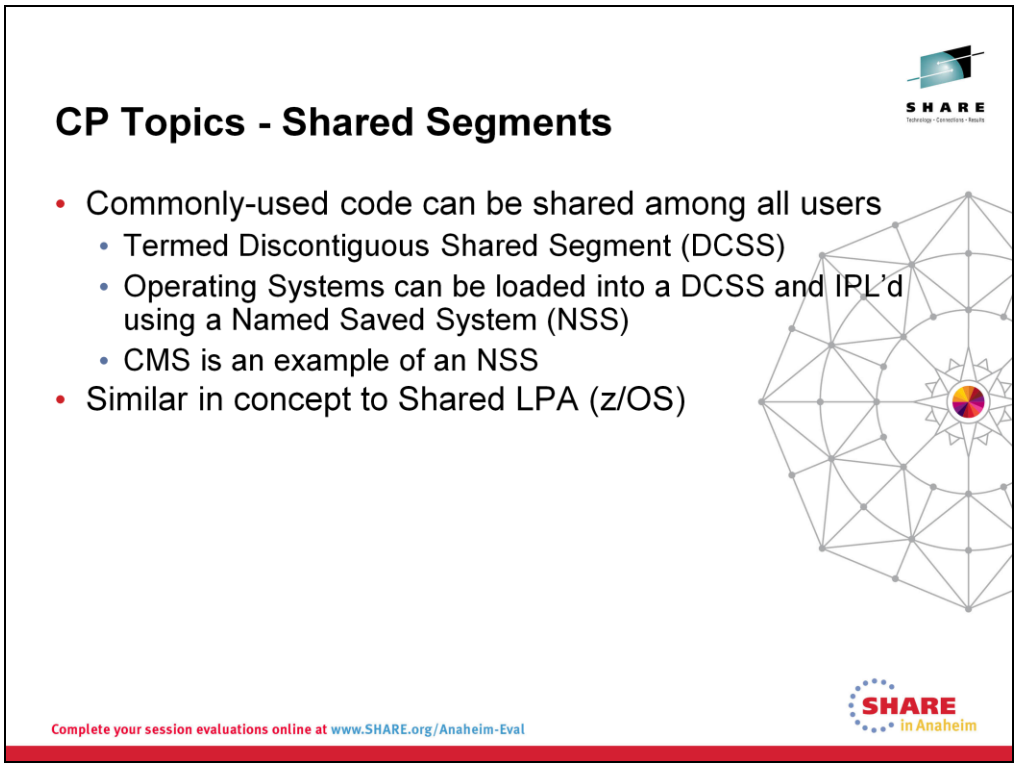


Complete your session evaluations online at www.SHARE.org/Anaheim-Eval

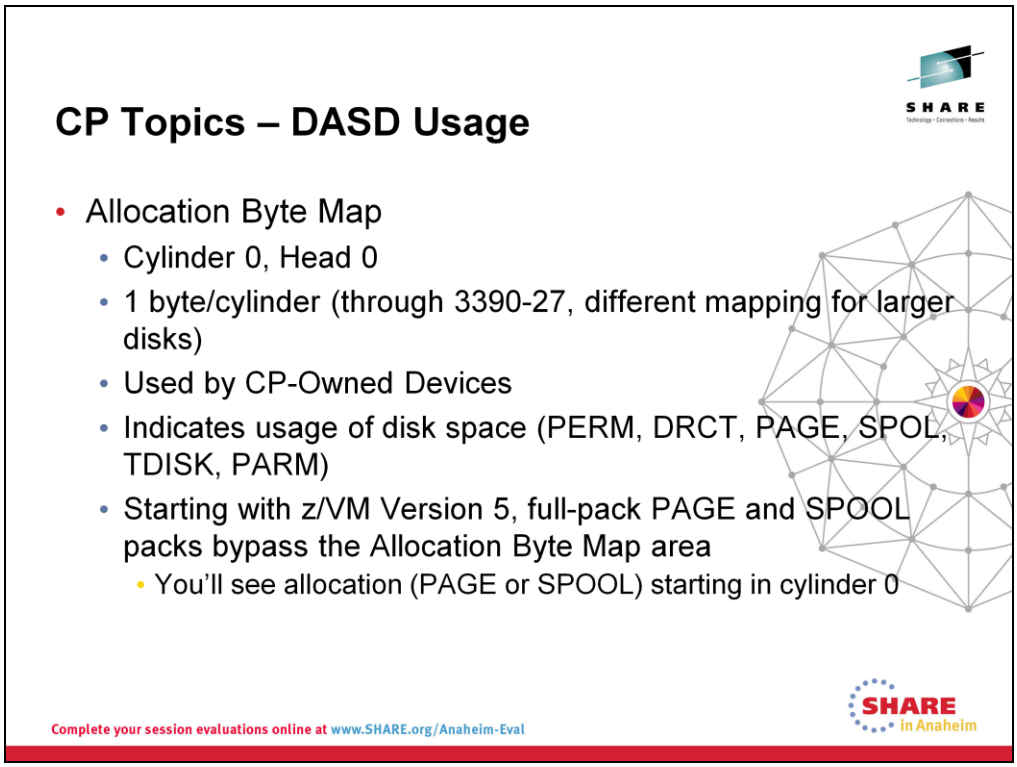
More Details



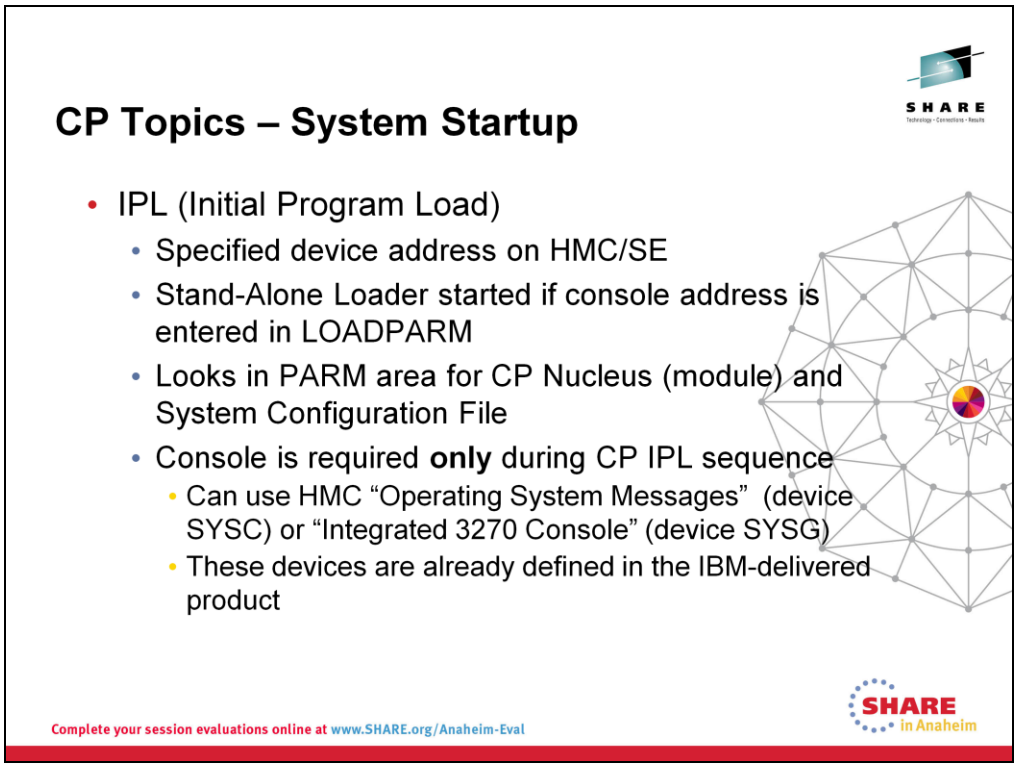
Complete your session evaluations online at www.SHARE.org/Anaheim-Eval



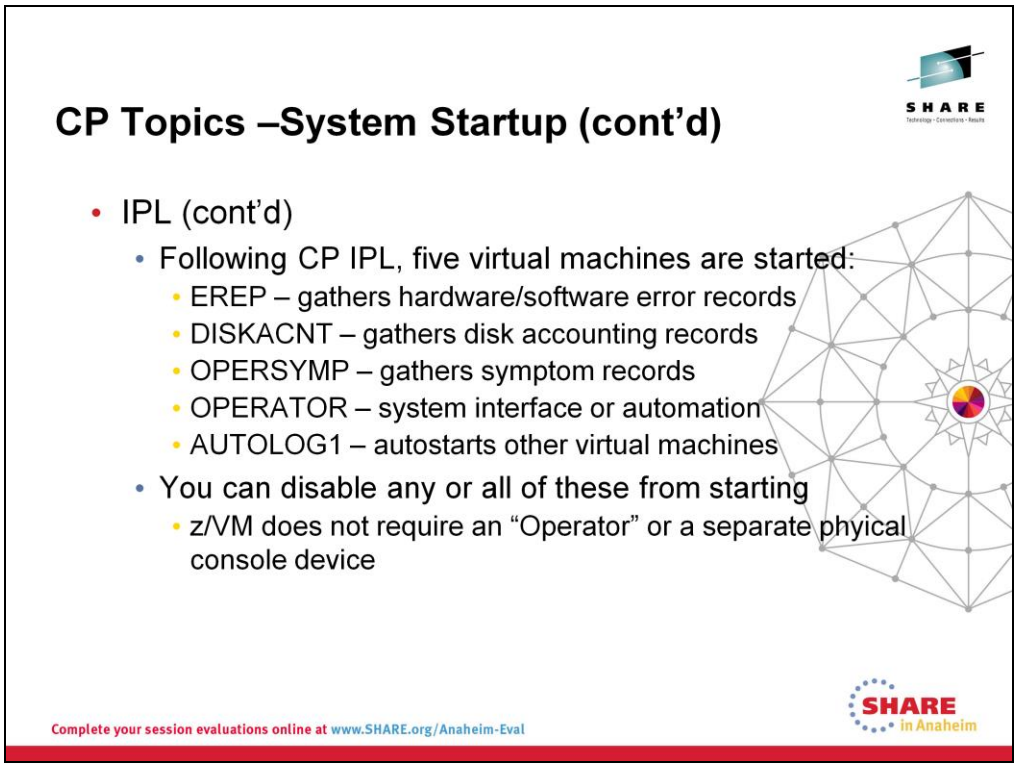
A Linux kernel can be made into an NSS, saving up to 1.5MB of private virtual storage per guest, although . This should be considered when you have many Linux instances.



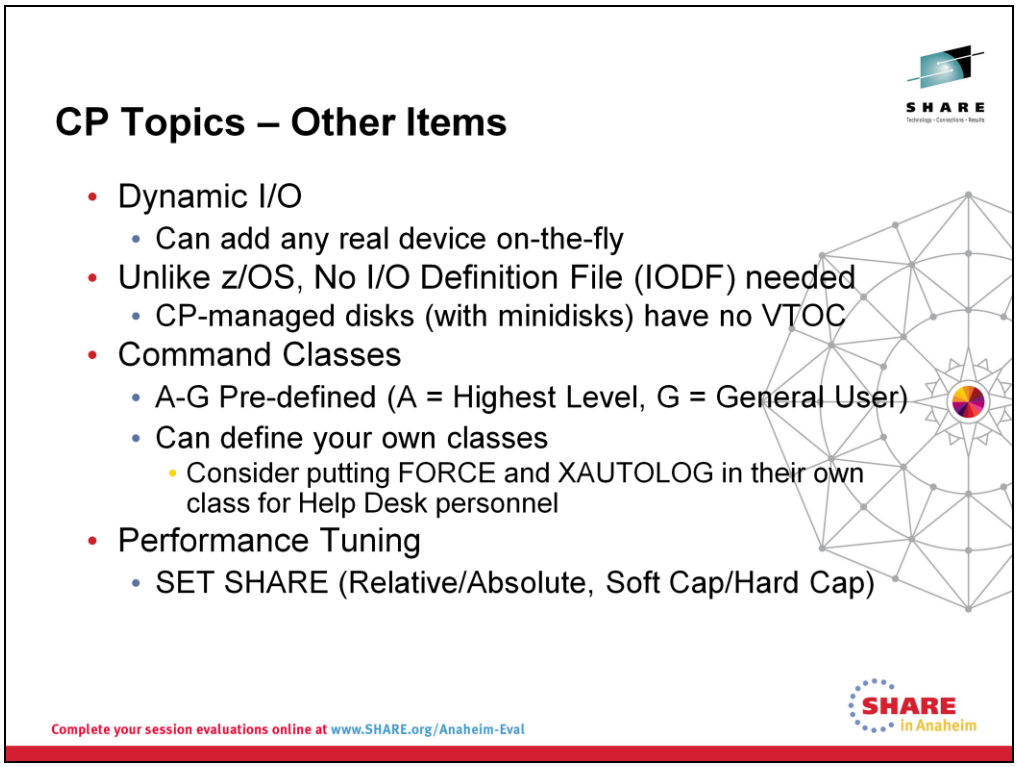
If you are planning to use full volumes for z/VM page/spool space, you must understand the role of the allocation byte map or strange and apparently random bad things may happen to your running system



The Allocation Byte Map is read by CP for all CP-OWNED volumes as listed in the System Configuration file. CP Dump Space is allocated in Spool based on the amount of real storage that CP sees at IPL time



You must plan for log management of EREP and DISKACNT space; most shops use the VMUTIL virtual machine for this via the WAKEUP service (similar to CRON)



IODF is an option starting with z/VM 4.4; it is **not** required