

z/VM's Control Program (CP)

Useful Things to Know

Lauren Maietti
z/VM CP Core Product Owner & Developer
lmaiett@us.ibm.com

Caroline Hart
z/VM Hardware Emulation Developer
caroline.hart@ibm.com



Disclaimer

© Copyright IBM Corporation 2025

IBM and the IBM logo are trademarks of IBM Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on ibm.com/trademark.

This document is current as of the initial date of publication and may be changed by IBM at any time. Statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

Examples presented are illustrative only. Actual results will vary based on client configurations and conditions and, therefore, generally expected results cannot be provided.

Not all offerings are available in every country in which IBM operates.

It is the user's responsibility to verify the operation of any non-IBM products or programs with IBM products and programs. IBM is not responsible for non-IBM products and programs.

The client is responsible for ensuring compliance with all applicable laws and regulations. IBM does not provide legal advice nor represent or warrant that its services or products will ensure that the client is compliant with any law or regulation.

Special thanks...

- John Franciscovich – Original Author
- Daniel Shekhtman – Content Refresh & Modernization

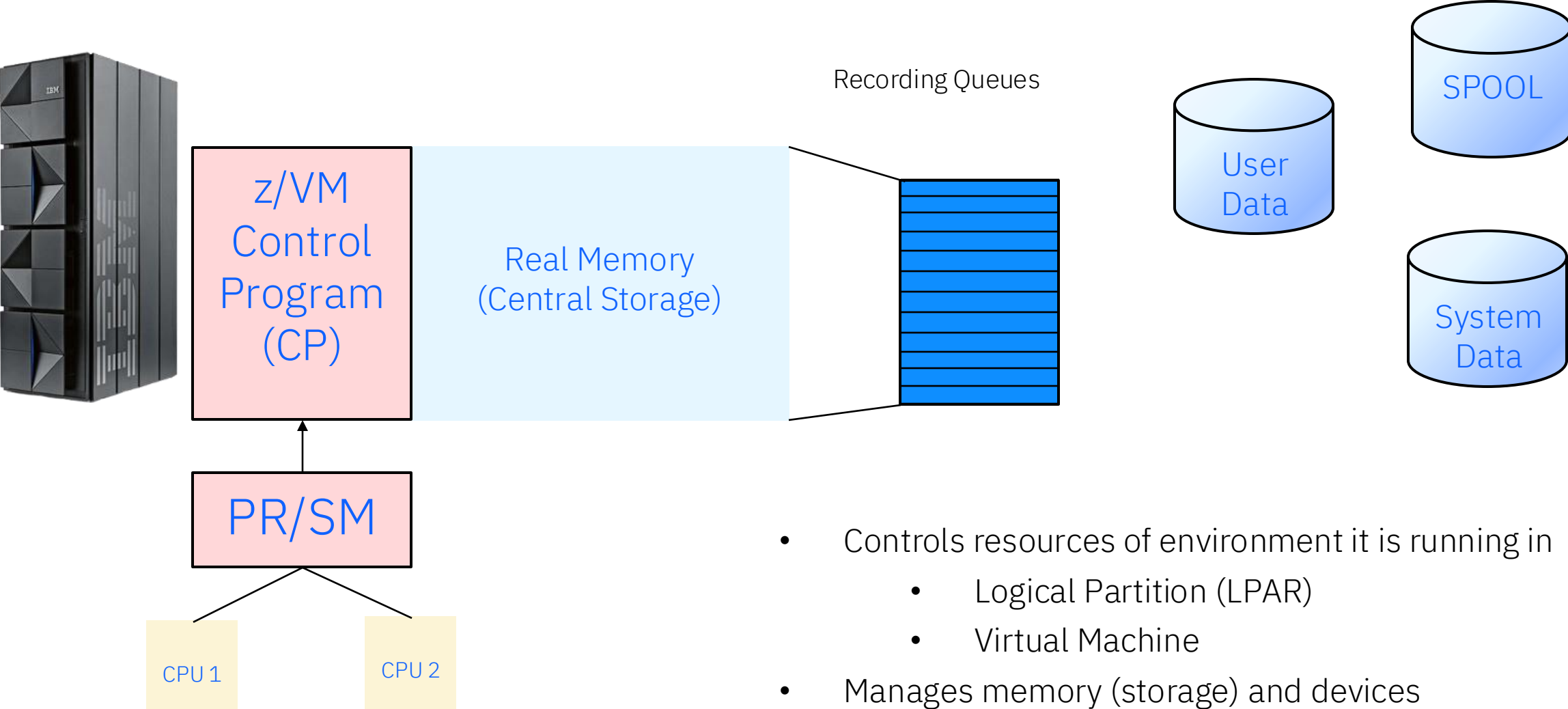


Topics

- Overview of z/VM's CP Facilities & Functions
- Booting (aka IPLing) CP
 - What you need
 - Saving and Restoring Information
- Defining & Creating Virtual Machines
- Virtual Machine Connectivity and Networking
 - Virtual Machine Communication
 - Virtual Networking
- Interacting with CP
- Diagnostic Data
- Summary

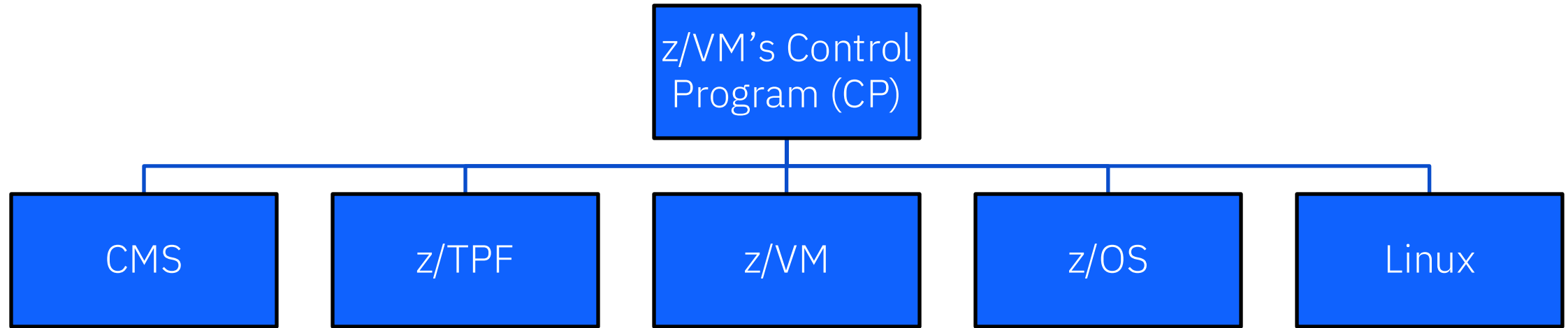
Overview of CP

CP – z/VM’s System Control Program



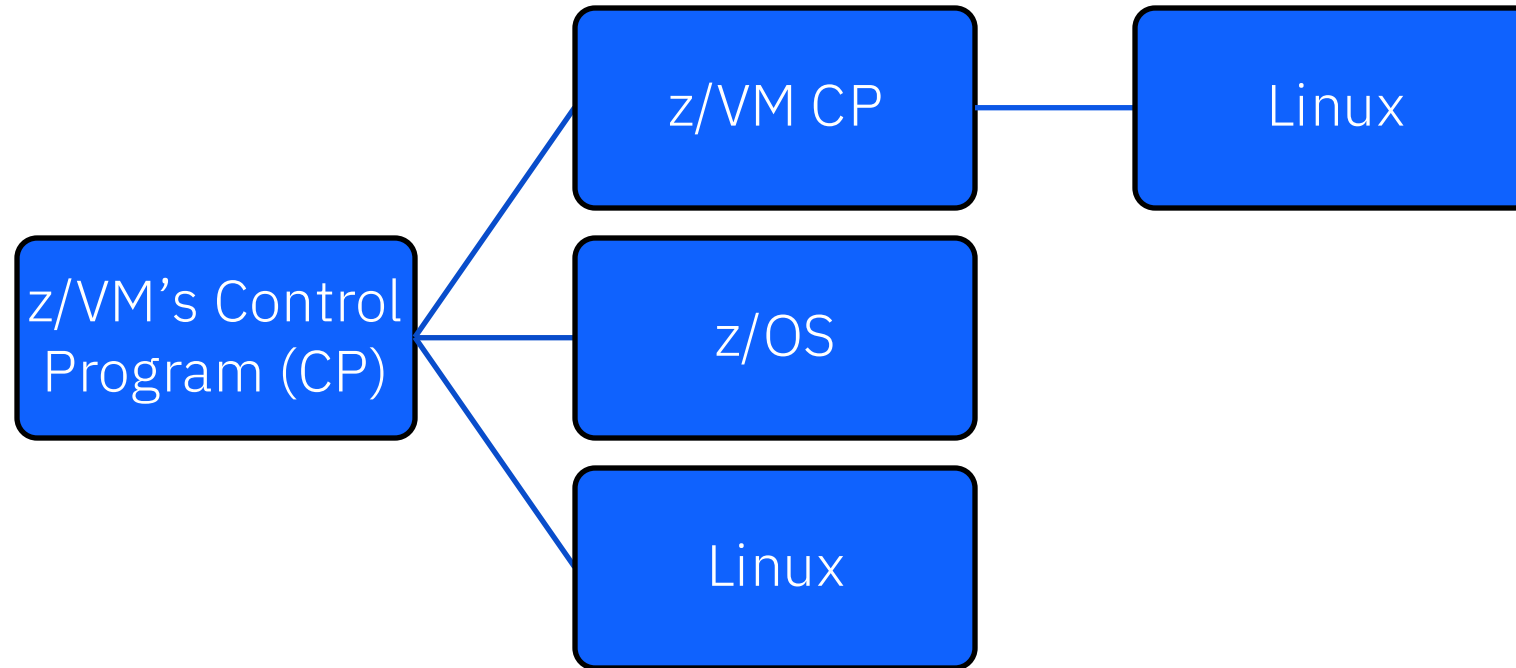
- Controls resources of environment it is running in
 - Logical Partition (LPAR)
 - Virtual Machine
- Manages memory (storage) and devices
- Records usage and system event data
- Provides error recovery facilities

CP – z/VM's System Control Program



- **Manages Virtual Machines**
 - ESA/390 and z/Architecture
 - Guest operating systems
 - Interactive users
 - Conversational Monitor System (CMS) is a special single user OS that is part of the z/VM product
- **Shares real resources among virtual machines**
- **Supports connectivity among virtual machines**
 - Virtual networking
 - Data sharing and exchanging information

CP – z/VM's System Control Program



- Supports multiple layers of virtualization (commonly referred to as n-level)
 - z/VM can run as a guest in a virtual machine
 - Guest z/VM system may host its own guest operating systems
 - With nested virtualization, performance degrades the further you get from the hardware.

CP Device Support

Real Devices (RDEVs)

- Sensed by CP at IPL time
 - Can also be defined to CP in system config file or dynamically
- Attached or dedicated to a single virtual machine for its exclusive use
- Virtualized and shared among several virtual machines
- Used by CP for system functions

Virtual Devices (VDEVs)

- Appear to virtual machine as a real device
- Defined
 - In virtual machine's directory
 - Dynamically after virtual machine is active
- Either virtualized or simulated
 - Virtualized - presents an image of a real device to virtual machines
 - Simulated - defined to virtual machine without an associated real device

CP Disk Space ("CP Owned")

PARM

- CMS Minidisk containing system configuration files
- Usually contains CP nucleus

DRCT

- User directory (created with DIRECTXA utility)

PERM

- Checkpoint & Warmstart Areas
- User Minidisks
- Could contain CP nucleus

PAGE

- System paging

SPOL

- Spool files, including DUMP files and other System Data files

TDSK

- Temporary disk space available to users

CP Disk Space (“CP Owned”)

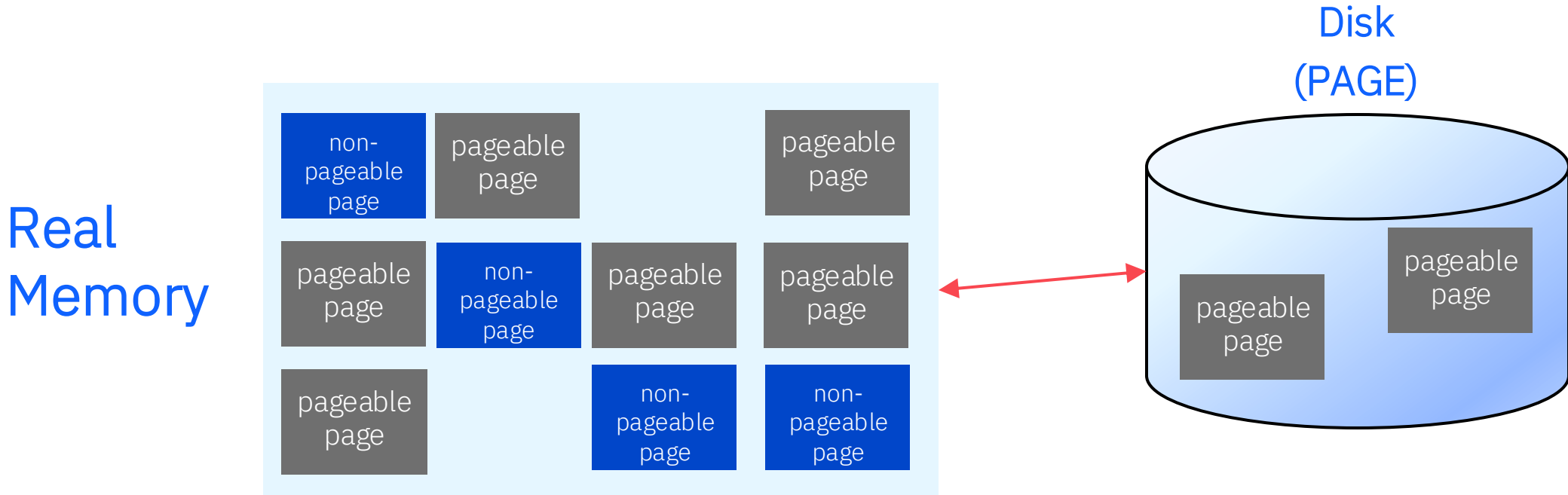
- CP disk space is defined in CP_Owned SYSTEM CONFIG file statement.

```
CP_Owned Slot 001 esares          /* System Residence volume */
CP_Owned Slot 002 essys1         /* Spool and T-disk space */
CP_Owned Slot 003 essys2         /* Spool space */
CP_Owned Slot 004 sysdmp Dump     /* Dump space */
CP_Owned Slot 005 spool1 RDEV 3BC0 /* More spool space */
CP_Owned Slot 006 Reserved       /* Leave some slots open in */
CP_Owned Slot 007 Reserved       /* case we need to add some */
CP_Owned Slot 008 Reserved       /* extra spool space later. */
```

- May be added dynamically to a running system using DEFINE CPOWNED
- CPFMTXA Utility formats & allocates types of CP disk space
- QUERY CPOWNED command shows list of CP owned disk volumes

```
q cowned
Slot  Vol-ID  Rdev  Type   Status          SSI0wner  Sys0wner
  1  RSE701  D700  Dump   Online and attached  -----
  2  SPE701  D509  Own    Online and attached  -----
  3  SPE702  D611  Own    Online and attached  -----
  4  SPE703  D502  Own    Online and attached  -----
  5  SPE704  D474  Own    Online and attached  -----
```

Managing Real Memory Among Virtual Machines



CP optimizes use of real memory for virtual machines

- Virtual machine memory is pageable
 - **Demand paged** - only paged out when necessary
- Paged to
 - Disk (CP-Owned PAGE area)

CP Spooling

Simulates real unit record devices

- Virtual unit record devices defined for each virtual machine
 - Reader
 - Printer
 - Punch
- Reads input (reader) files and makes data available
- Writes data into output (punch or printer) files
- Files may be sent to (or read from) associated real devices
- SPOOL is an acronym for Simultaneous Peripheral Operations On-Line

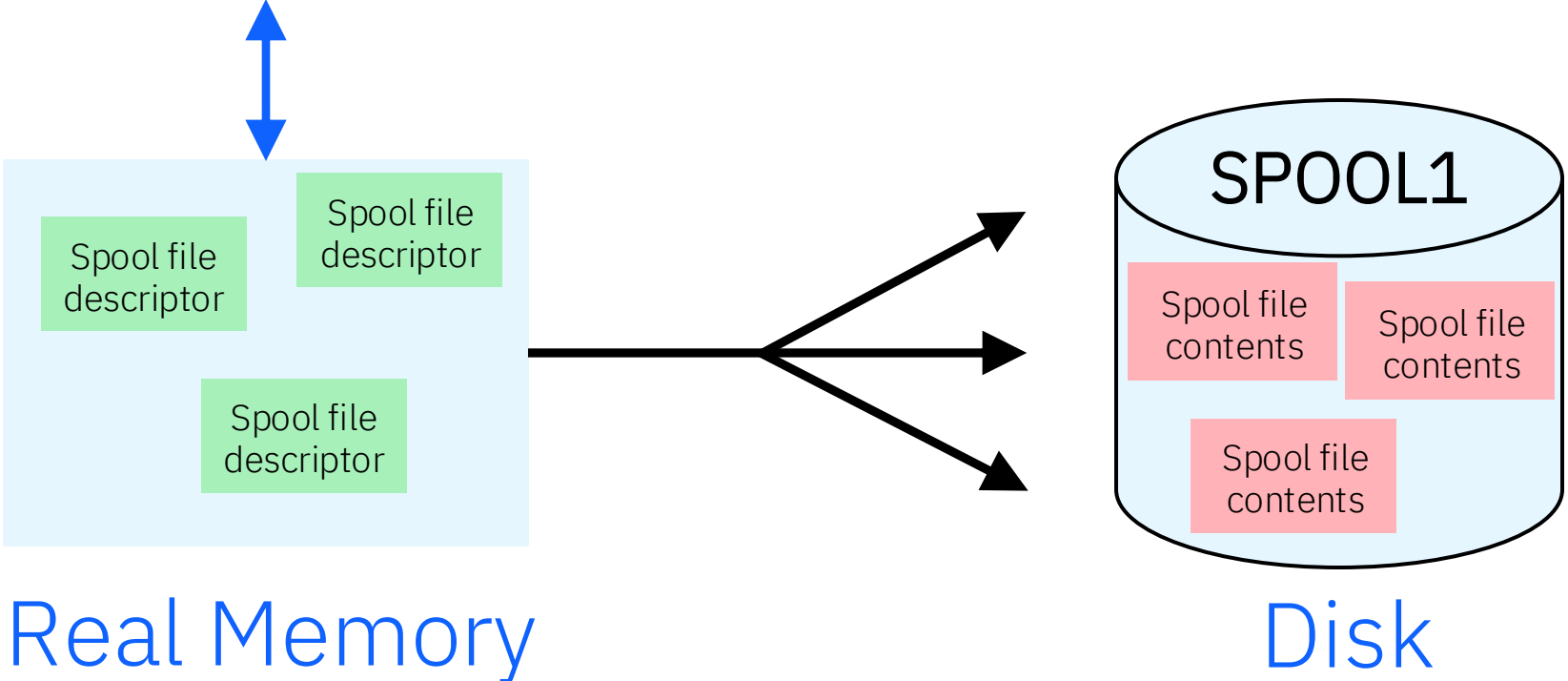
SPOOL files are used for

- Email
- Transferring information between virtual machines and systems
- Sending/Receiving information from associated real devices
- Saving console output
- System and virtual machine dumps
- Specific system functions

CP Spooling

```

q rdr all
ORIGINID FILE CLASS RECORDS CPY HOLD DATE TIME NAME TYPE DIST
DMSHEKHT 0216 A PUN 00000049 001 NONE 01/25 15:07:05 PROFILE EXEC 35H:0253
OPERATOR 1903 A RDR 00000006 001 NONE 01/24 12:07:18 OPERATOR
USER1 0046 A PUN 00000086 001 NONE 07/16 10:22:32 TEXTFILE XEDIT USER1
    
```



(CP-Owned SPOOL area)

CP SPOOLing – System Data Files

NLS (National Language Support)

- Message repositories for translated z/VM messages

TRF (System Trace Files)

- System trace data generated by a virtual machine or TRSOURCE and TRSAVE commands

NSS (Named Saved System)

- Named copy of an operating system

DCSS (DisContiguous Saved Segment)

- Saved copy of data and/or code

IMG (Image Library)

- Definitions such as spacing and character sets used by printers

UCR (User Class Restructure)*

- Customized privilege class information for commands and diagnose codes

*UCR files are no longer supported as of z/VM 6.3 (2013). Use MODIFY CMD/PRIV_CLASSES instead!

Named Saved Systems & Saved Segments

NSS

(Named Saved System)

- Code is saved in a segment
- Can be IPL'd by name (e.g. CMS)
- Single copy on system for N virtual machine instead of N copies.

DCSS

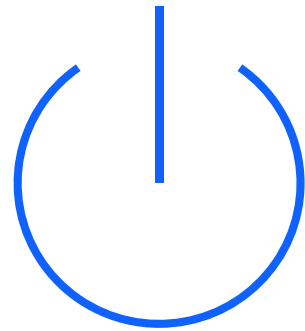
(DisContiguous Saved Segment)

- Used to contain shared Data or Code
- Single copy is shared amongst all virtual machines

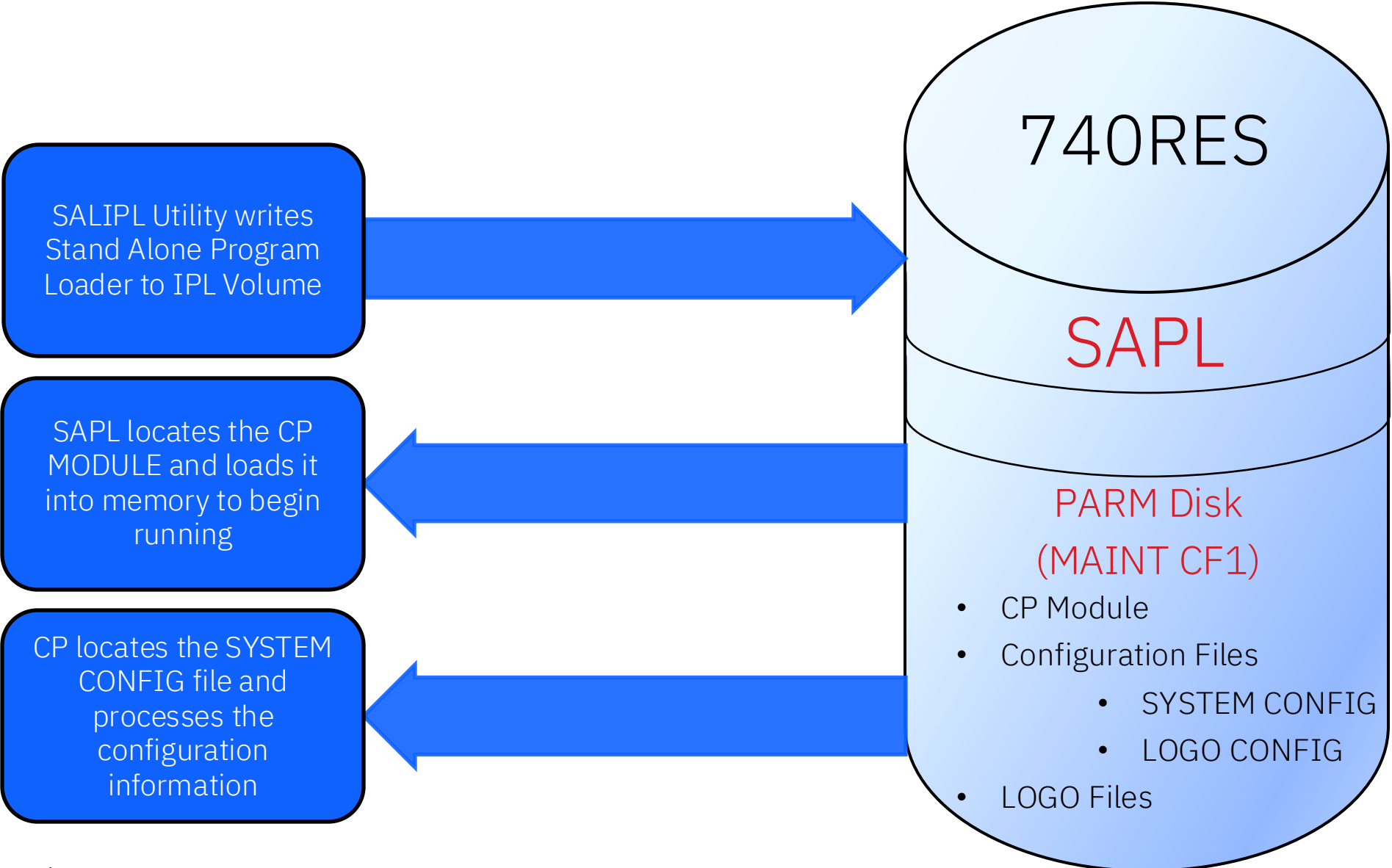
Created with
DEFSYS and
DEFSEG
commands

- Skeleton Files
- Must be saved with SAVESYS and SAVESEG commands before they can be used by virtual machines

Booting (aka IPLing¹) CP



What you need to IPL CP



Restoring Information during Boot (aka IPL)

CP saves system environment and data during SHUTDOWN, including:

- Accounting, EREP¹, and Symptom records
- Unit record device status
- System log messages
- Spool files
- System Data files

Type of IPL determines how much saved system information is restored

- WARM
 - Restores all information saved during SHUTDOWN
- FORCE
 - Restores as much saved information as possible
- COLD
 - Only restores system data files
- CLEAN
 - Does not restore any saved information

Restoring System Data..

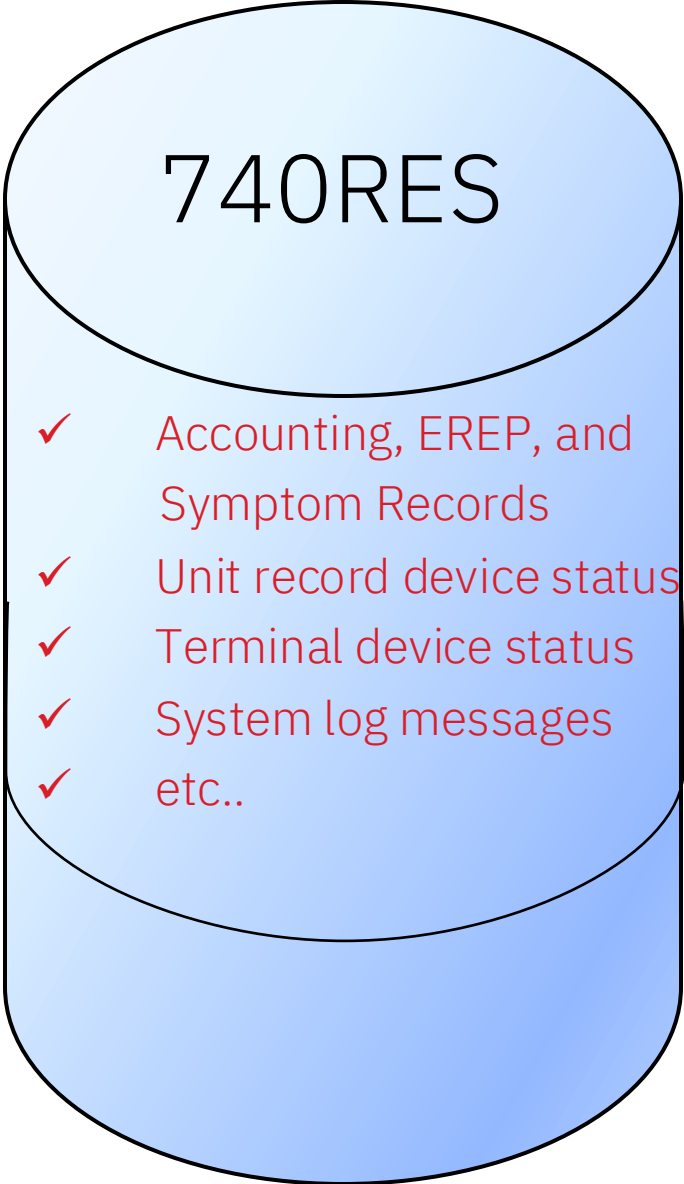
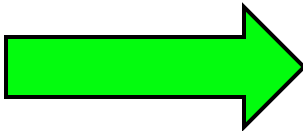
Checkpoint Area

System Data to be restored during an IPL (WARM or FORCE)

- Located on a CP-owned volume
- Not Necessarily the IPL volume

```
System_Residence,  
Checkpoint Valid 740RES From CYL 21 For 9 ,  
Warmstart Valid 740RES From CYL 30 For 9
```

CYL 21-29



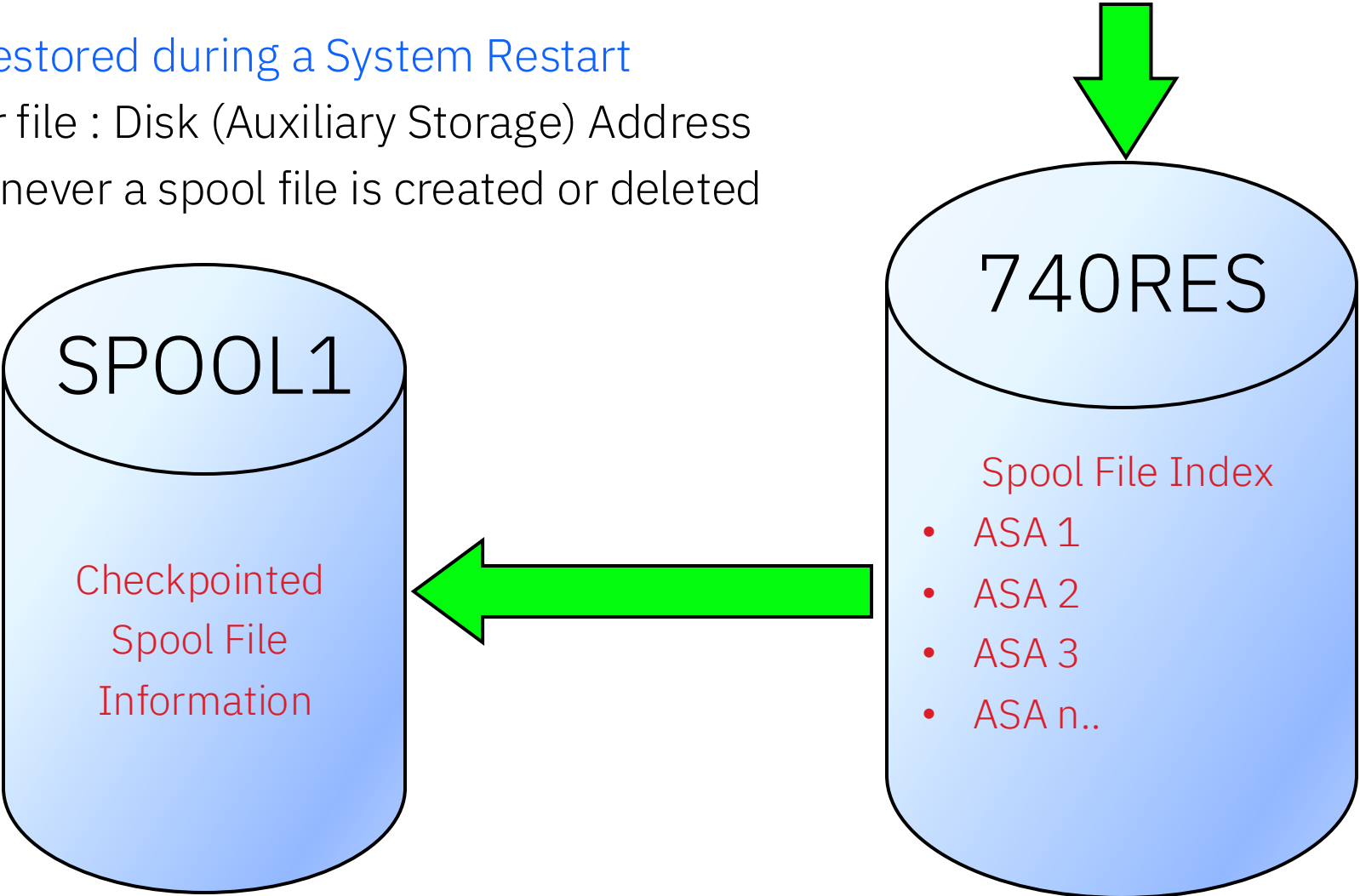
Restoring System Data..

Warmstart Area

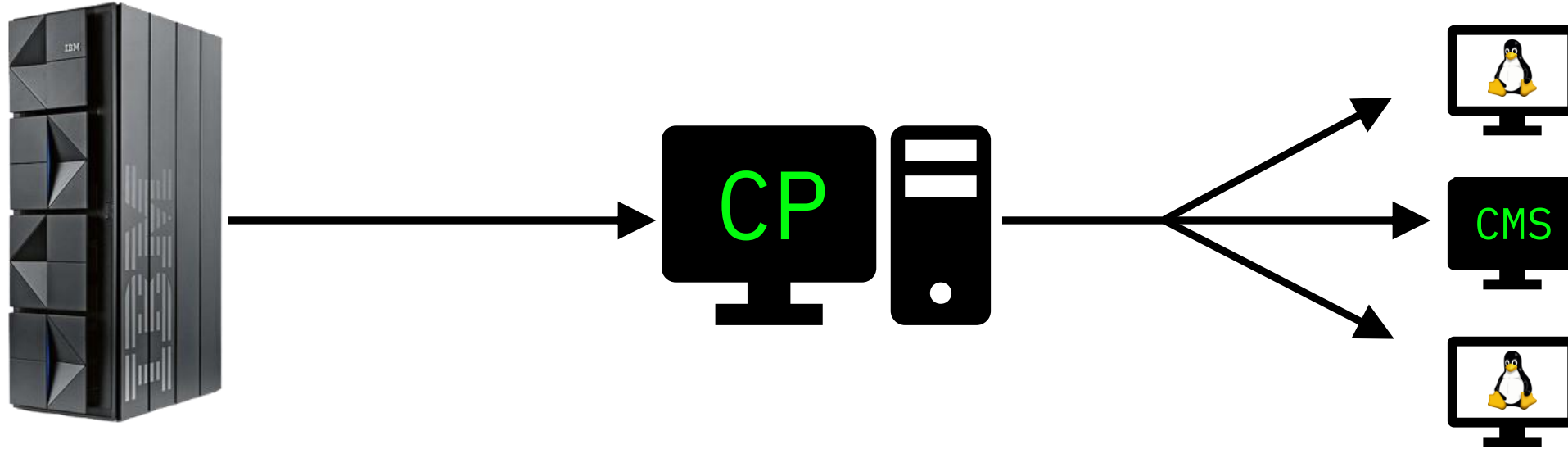
Spool Files to be restored during a System Restart

- One entry per file : Disk (Auxiliary Storage) Address
- Updated whenever a spool file is created or deleted

```
System_Residence,  
Checkpoint Valid 740RES From CYL 21 For 9 ,  
Warmstart Valid 740RES From CYL 30 For 9
```

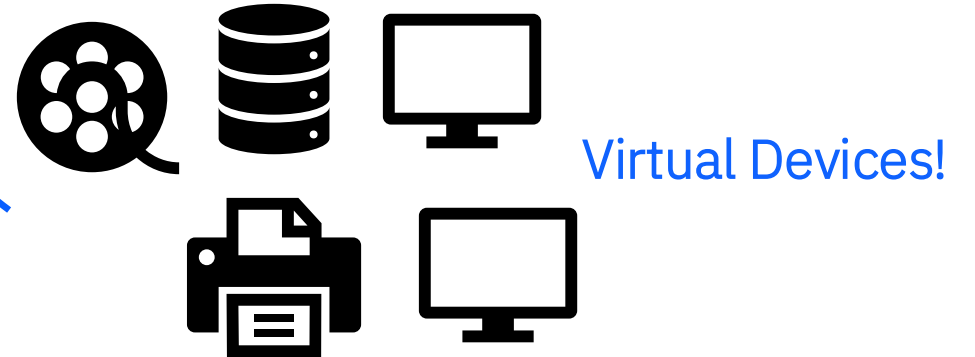
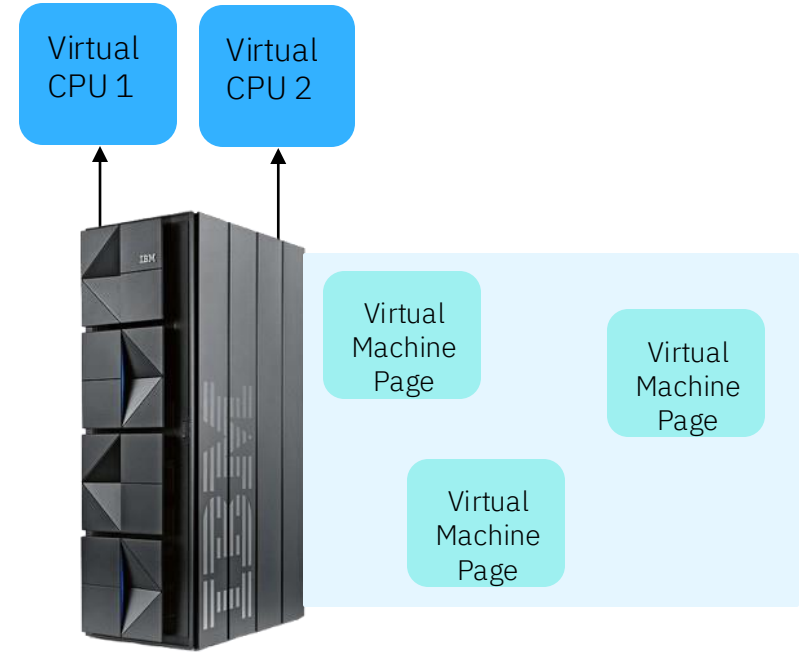
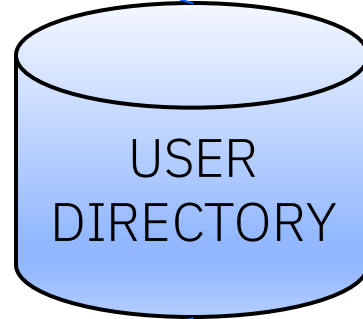


Virtual Machines



Defining a Virtual Machine

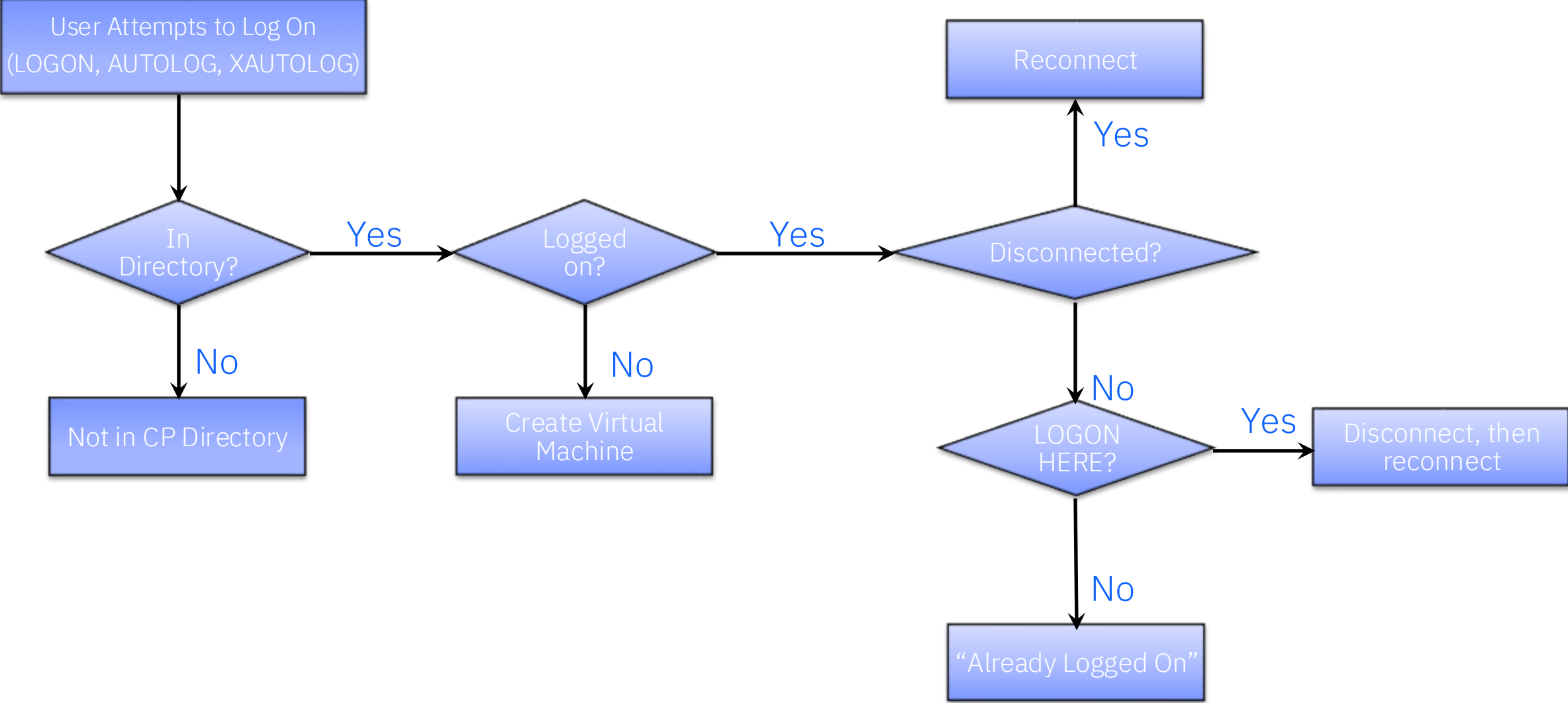
```
USER U1    U1PW 1024M 32G G
IPL 190 PARM AUTOOCR
MACHINE ESA 2
COMMAND ATTACH 555 * 555
CONSOLE 0009 3215 T MAINT
SPOOL 000C 2540 READER *
SPOOL 000D 2540 PUNCH A
SPOOL 000E 1403 A
MDISK 191 3390 000 009 DAN191 MR
MDISK 193 3390 000 017 DAN193 RR
LINK MAINT 190 190 RR
LINK MAINT 19D 19D RR
```



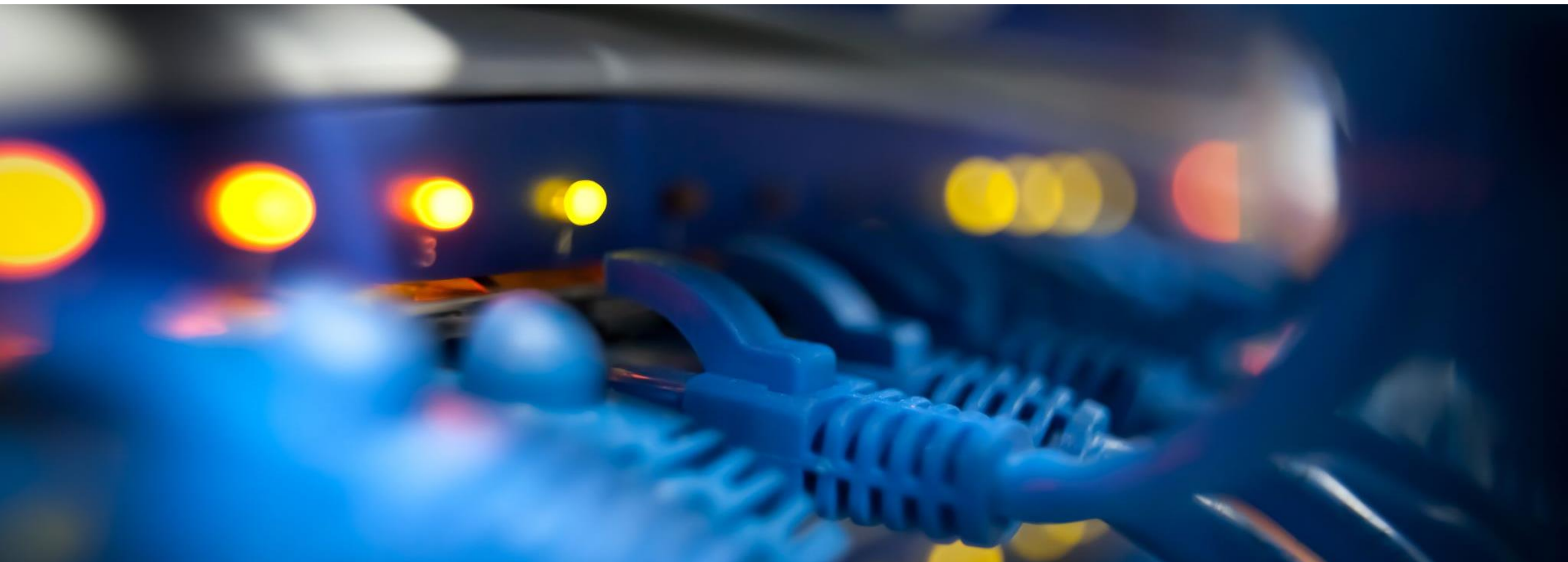
Created when a user logs on

- Real hardware and features are virtualized
 - Processors, Devices, Memory
- Aggregate of virtual objects and memory may be greater than available real resources

Logging onto z/VM (creating a virtual machine)



Virtual Machine Connectivity & Networking



IUCV

Inter-User
Communication
Vehicle

Provides an efficient data transfer protocol unique to the z/VM platform

Communication
between Virtual
Machines

Virtual CTCA

Simulates existence of real Channel-to-Channel devices for each virtual machine

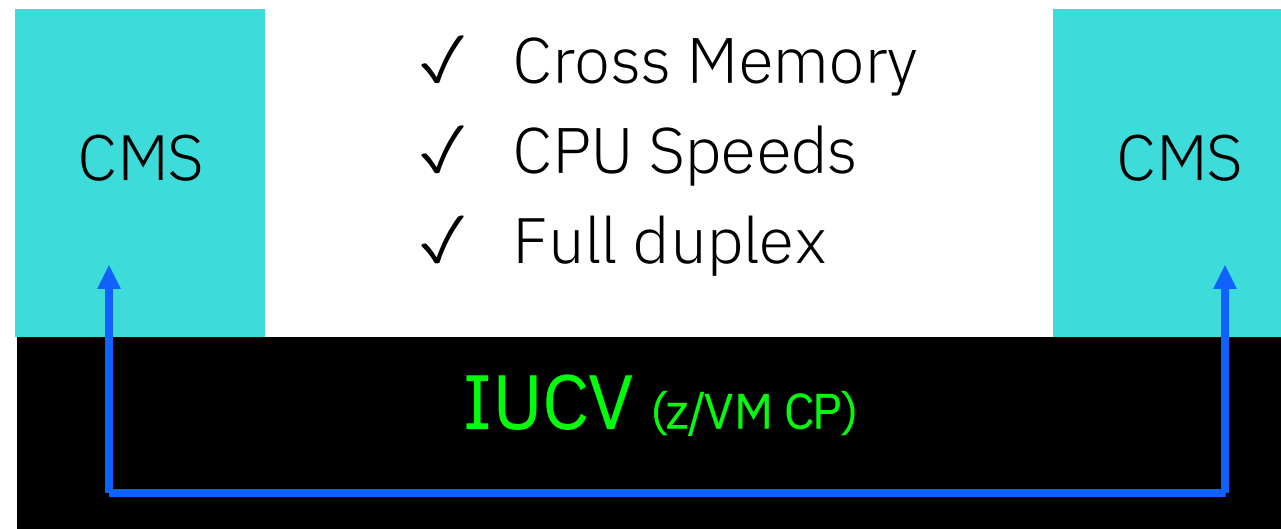
Virtual NIC

Simulates existence of real Network Interface Cards for each Virtual Machine

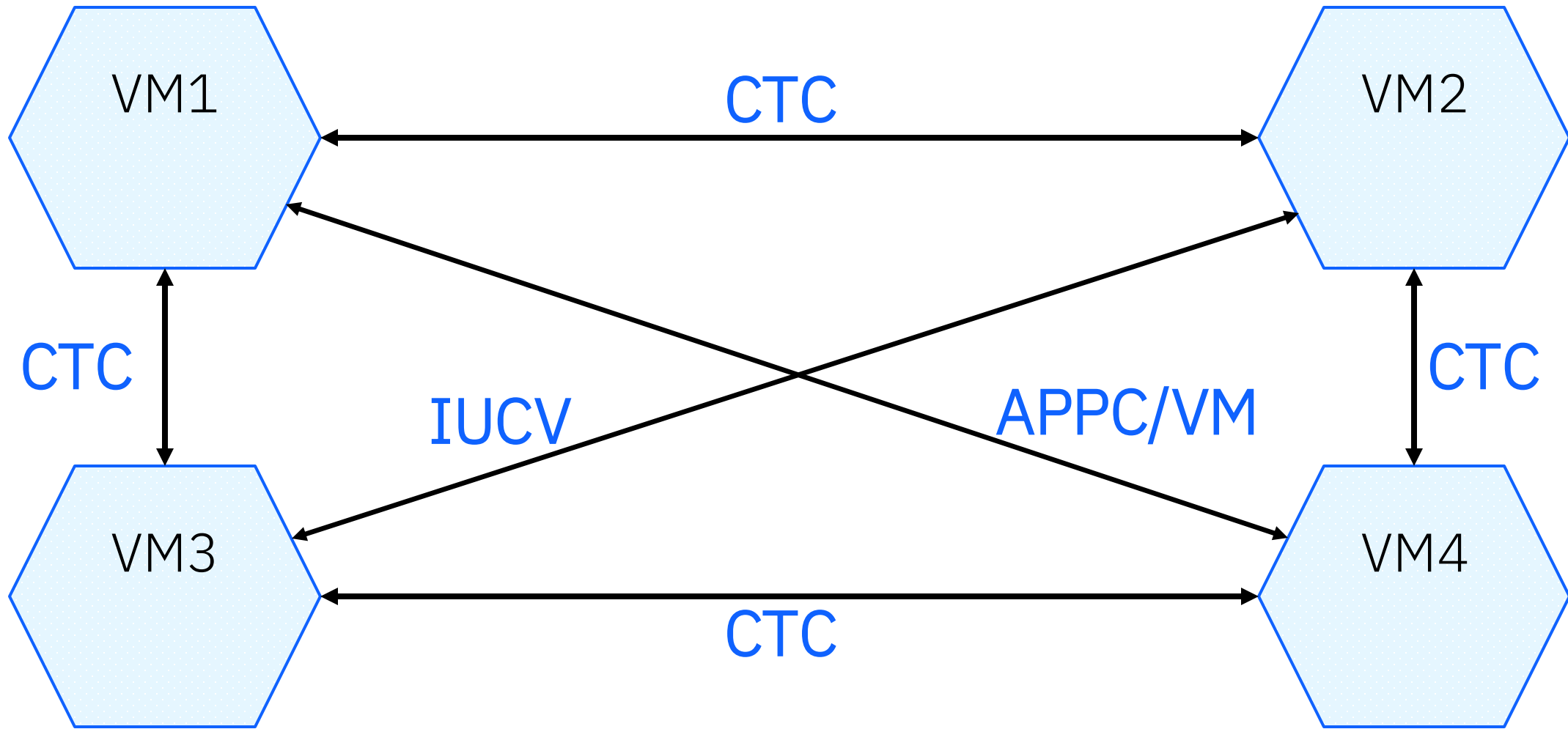
IUCV Communication

Inter-User Communication Vehicle

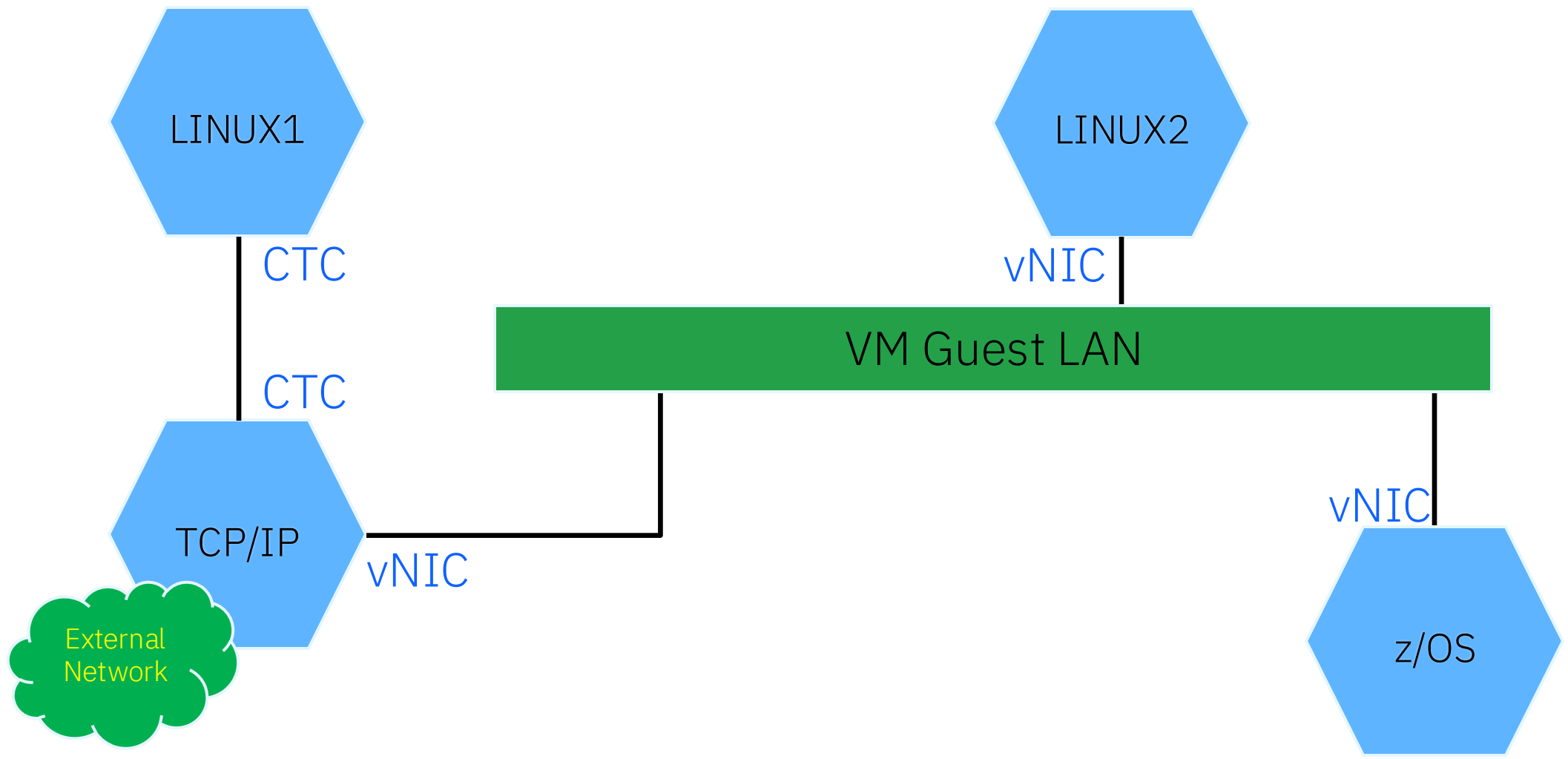
- Allows communication between an application and other virtual machines or CP system services
- Simultaneous communication over multiple connections allowed for each virtual machine
- Transparent communication between virtual machines on different systems via ISFC (Inter-System Facility for Communications)
- Point-to-Point networking between Linux and z/VM TCP/IP



A VM ISFC Collection



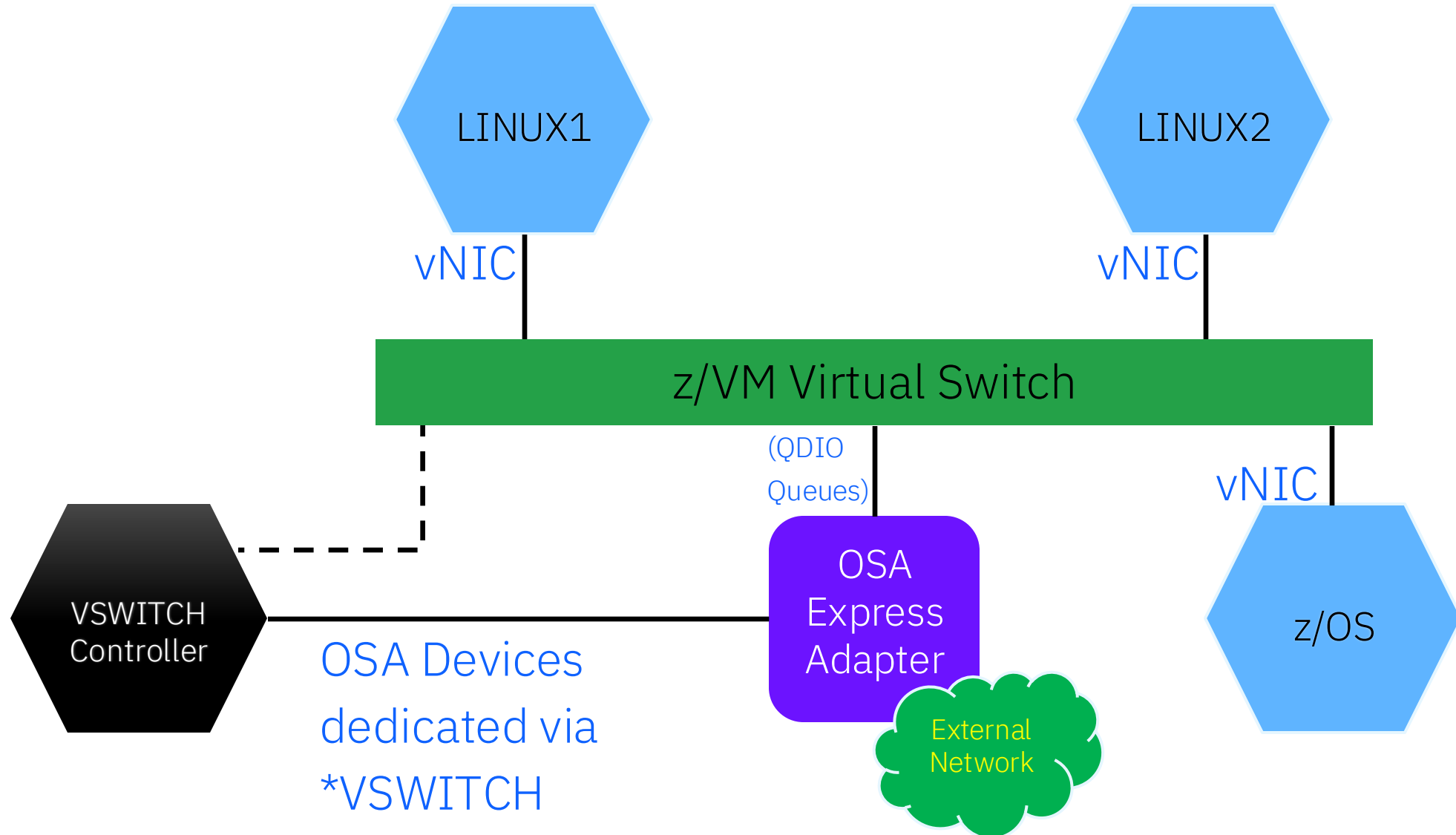
A Virtual Network (Guest LAN)



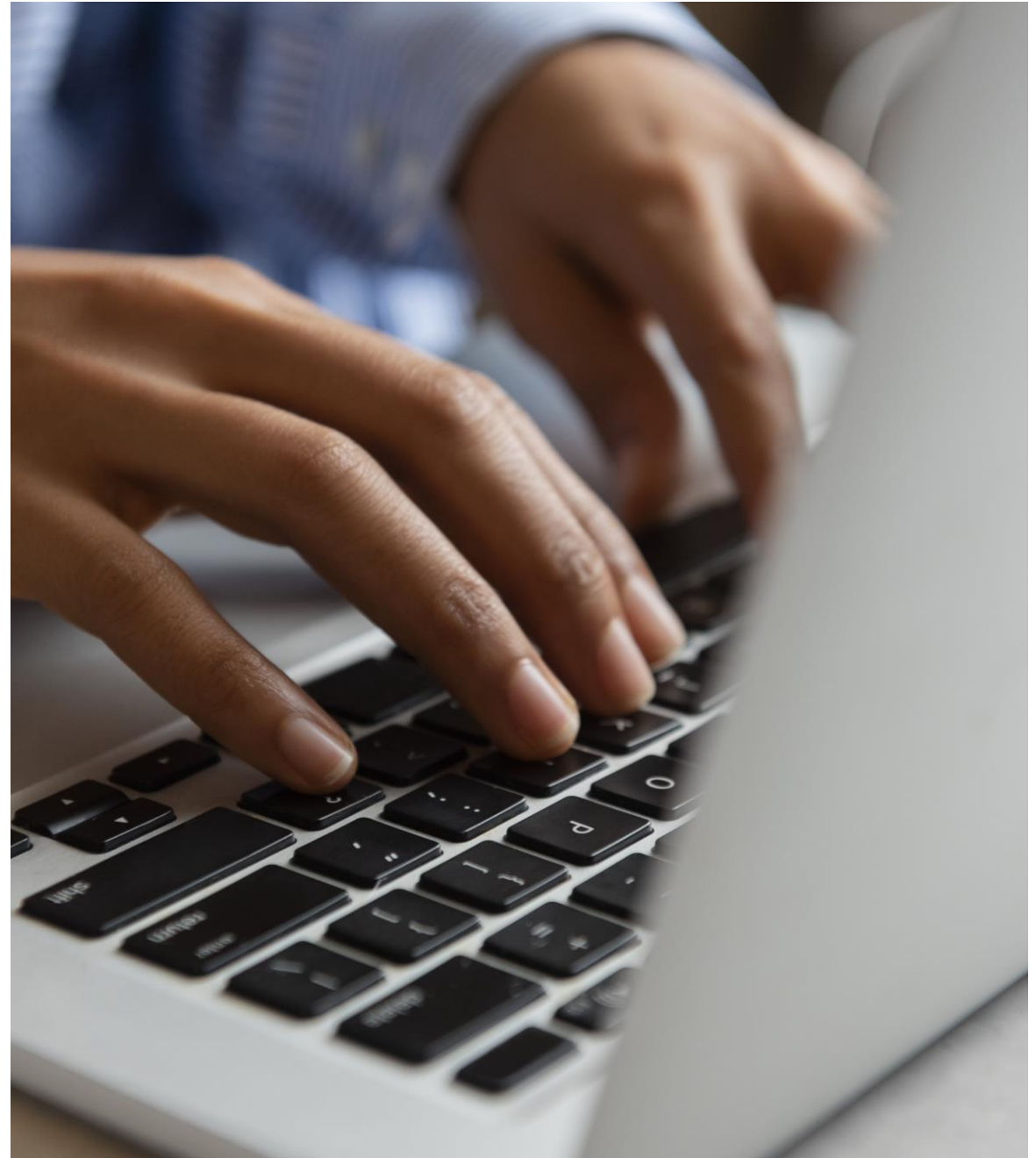
All connections within the z/VM host are virtual

- No physical connections are necessary

A Virtual Network (z/VM Virtual Switch)



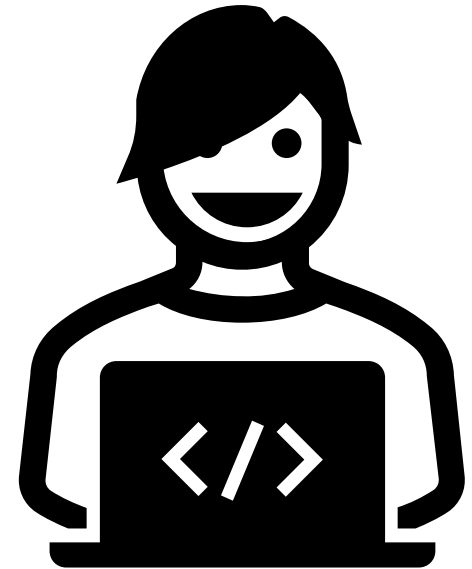
Interacting with CP



CP Commands

Used for a variety of purposes, including:

- System Operator functions
- DEFINE/SET/QUERY
 - System and Virtual Machine characteristics
 - Real/virtual device settings
 - System and user data.
- Assigning & Releasing system resources
- Moving data and files between users
- Communication between virtual machines

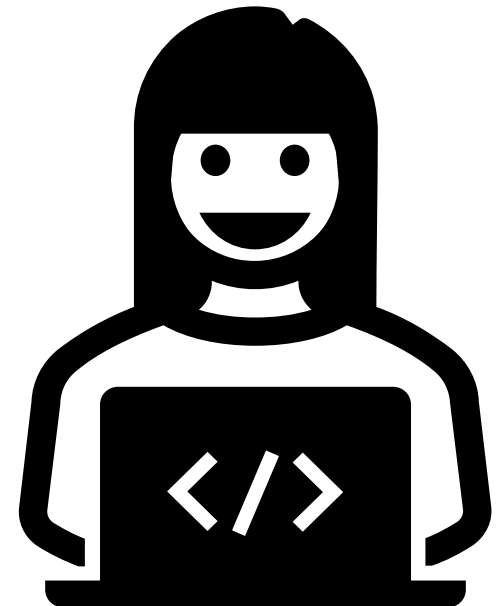


CP Commands...

Used for a variety of purposes, including:

- COMMAND directory statement
- FOR command
 - Allows a user to issue CP commands on behalf of another user
 - Issuer must have SECUSER authority or Class C privileges
 - Target must have appropriate authorization for subject command
- Command responses send to issuer
 - No indication to target user
- Example: from user OPERATOR

```
FOR u1 CMD QUERY VIRTUAL STORAGE
Ready;
11:59:21 U1 : STORAGE = 32M
11:59:21 U1 : HCPFOR069I Command Complete. CP return code = 0000
```



Privilege Classes

Each virtual machine/user has one or more privilege classes

- Most are only class G

Identifies:

- CP commands the user is allowed to issue
- Scope of influence of commands

Privilege classes may be modified

- User
- Command

Class	User and Function
A	System Operator: responsible for the availability of the z/VM system and its resources. In addition, the system operator controls system accounting, broadcast messages, virtual machine performance options, and other options that affect the overall performance of z/VM.
B	System Resource Operator: controls all the real resources of the z/VM system, except those controlled by the system operator and the spooling operator.
C	System Programmer: updates or changes system-wide parameters of the z/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 z/VM storage areas.
F	Service Representative: obtains, and examines in detail, data about input and output devices connected to the z/VM system. This privilege class is reserved for IBM® use only.
G	General User: controls functions associated with a particular virtual machine.
Any	Commands belonging to class “Any” are available to any user, regardless of the user's privilege class. These commands are primarily those used to gain access to, or relinquish access from, the z/VM system.

CP Programming Interfaces

Provide application programs with access to:

- CP Services
- Data created by CP to be processed by applications
- Certain CP Data areas

Types of programming interfaces

- Diagnose Codes
- CP System Services
- IUCV and APPC/VM macros
- Address space macros (ESA/XC virtual machines)



Customizing CP

CP Exit Support

- Allows non-disruptive additions and deletions of customized CP routines
 - CP Commands
 - Diagnose Codes
 - Message Repositories
 - Exit Routines (user modifications to CP)
- No need to shutdown and IPL to apply user code
 - Modifications applied with commands and configuration file statements
- Minimizes rework to user code due to IBM source code changes

Diagnostic Data



Diagnostic Data

Several types of data created by CP can help diagnose problems!

- Console messages and logs
- Dumps
 - System (CP Dumps)
 - Virtual Machine (VMDUMPs)
- TRACE Data
 - TRSOURCE
- Performance Data
 - MONITOR data
 - Reports from Performance Tools (PerfKit, OMEGAMON XE, Velocity, etc.)
 - Live performance data from z/VM Performance Data Pump
 - INDICATE commands

Diagnostic Data continued..

Commands may be used to gather additional information

- QUERY (some examples below)
 - **QUERY CPLEVEL** to display current maintenance level of CP
 - **QUERY COLLECT** to display information about nodes and routes within an ISFC collection
 - **QUERY DASD DETAILS <rdev>** to display detailed information about a real DASD device
- LOCATE
 - Host Logical and/or Host Real address, depending on resource being located
- DISPLAY
 - Specify Host Logical or Host Real addresses to be displayed

Console Messages and Logs

Most applications & system functions write messages to the virtual machine console

- System messages are displayed on the OPERATOR console

Console information can be easily saved for review!

- SPOOL CONSOLE START * command
 - Begin collecting console data
 - Direct console file to desired virtual machine (in this case, to ourselves)
- SPOOL CONSOLE STOP/CLOSE command
 - Stop collecting console data
 - Close the file so it may be saved and reviewed
 - Console file is sent to the virtual reader of the user you are spooling the console to
- RECEIVE file to disk or PEEK it in RDR

CP Dumps

Hard Abend (various codes)

- Contains all CP-owned memory

Soft Abend (various codes)

- Does not cause system termination
- Contains
 - VMDBK of the active virtual machine at time of abend
 - CP Trace table for processor where error occurred

SNAPDUMP (Code SNP001)

- Identical to a Hard Abend dump, but no PGMBKs or FRMTBL unless requested
- Non-disruptive but will quiesce your system for time it takes to dump, which can vary.

Restart Dump (Code SVC002)

- Triggered from HMC via PSW RESTART function. Dumps then restarts the LPAR.
- Identical to Hard Abend dump. Ideal for hung user or hung system situations.



More Dumps!

VMDUMP (Virtual Machine Dump)

- Created with VMDUMP command
 - Unformatted dump
 - 4K Pages of virtual machines memory
 - Placed in virtual reader
 - DUMPLD2 command used to load into CMS file
 - The maximum address that VMDUMP can dump is X'7FFFEFFF' for ESA/390 and ESA/XC virtual machines, and X'7FFFFFFFFF' (512G) for z/Architecture[®] and z/XC virtual machines.

Stand-Alone Dump (SAD)

- Same format as CP abend dump
 - Writes dump of all main memory
- Created when you IPL the Stand-Alone Dump Utility volume from HMC
 - Utility created by SDINST EXEC which formats and initializes the IPL and DUMP DASD
- You need to have media prepared **BEFORE** you have a problem.
- See Chapter 12 of CP Planning & Admin for preparation steps.
- ABEND code SAD001 or SAD002

Processing CP Dumps

CP Dumps are generally sent to OPERATNS reader (RDR)

- DUMPLD2 command processes dumps from RDR to disk

The VM Dump tool (VMDUMTPL) is used to analyze dumps

- CP Abend, SNAPDUMP, Restart, Virtual Machine, or Stand-Alone dumps.
- Issue VMDUMPTL command on MAINT 193 disk.

```
VM Dump Tool - z/VM 7.4.0 - HTT001 - 2U0F144A MDMP0001 M1

z/VM Version 7 Release 3.0, service level 0000 (CP 64-BIT)
Generated at 07/03/24 19:44:54.000000, IPLd at 07/03/24 19:55:25.874550
Date 07/03/24 Time 21:46:16.840471

CPUID = 0008E198 91758000

CPU address is 0000      Prefix register is 00132000      (failing)
CPU address is 0002      Prefix register is 0105E000
CPU address is 0004      Prefix register is 01062000
CPU address is 0006      Prefix register is 01066000
7F440040 21:46:16  HTT001 Hard Abend svc 00 at HCPHTT+E2E opsw 04040001_0041042E svcilc 0002

Summary of CP exits
    1 Pre-defined exit found
    0 Dynamic exits found
    0 Diagnose exits found

HTT001 CP fault in primary space caused by an inappropriate reference to an address in storage.
This abend is usually the result of a translation exception.

>>> dumptype
Dump type is 64-bit Hard Abend Format (CPDUMP) with FRMTBL NO PGMBKS NONE

>>> describe
HTT001 CP fault in primary space caused by an inappropriate reference to an address in storage.
This abend is usually the result of a translation exception.
```

Tracing

General CP Tracing

- CP builds trace tables for each CPU during initialization
- All occurrences of traceable system events are recorded.

VMDUMPTL Display of CP Trace Table

```
>>> trace cpu 0 for 100 one
07E00380 00 14:57:50.077529 HTT001 Hard Abend svc 00 at HCPHTT+A82 opsw 04040001_0034A0B2 svcilc 0002
07E00340 00 14:57:50.077529 Pg trans except at HCPVLC+AC2 instr LLGT R4,X'10' (,R10) xlt-adr 00000000_62E6E800
07E00320 00 14:57:50.077528 Call fr HCPIQG+A30 to HCPVLCDX cpebk 0254C600 parm 01A73068 iac Primary
07E00300 00 14:57:50.077528 Rtrn to HCPIQG+A1C fr HCPSWU+22EA cpebk 21FDD600 cc1 rc>FFF iac Primary
07E002E0 00 14:57:50.077528 Rtrn to HCPSWU+22E4 fr HCPSWR+436 cpebk 0254C600 cc1 rc>FFF iac Primary
07E002C0 00 14:57:50.077528 Rtrn to HCPSWR+430 fr HCPSWC+168E cpebk 02449E00 cc1 rc=0 iac Primary
07E002A0 00 14:57:50.077528 Rtrn to HCPSWC+13CA fr HCPSWI+B76 cpebk 0249DC00 cc1 rc=0 iac Primary
07E00280 00 14:57:50.077527 Rtrn to HCPSWI+3BC fr HCPIQG+120C cpebk 02445200 cc1 rc=0 iac Primary
07E00260 00 14:57:50.077527 Rtrn to HCPIQG+DAA fr HCPSWU+1C9A cpebk 02442600 cc1 rc>FFF iac Primary
07E00240 00 14:57:50.077527 Rtrn to HCPSWU+1C90 fr _ERMES+8EC cpebk 02450800 cc1 rc=0 iac Primary
07E00220 00 14:57:50.077527 Release 76 dw (???) at 489EE2B0 by _ERMES+8E2 vmdbk 00100000 SYSTEM
07E00200 00 14:57:50.077527 Rtrn to _ERMES+866 fr _ERMBM+900 cpebk 65F18C00 cc1 rc=0 iac Primary
07E001E0 00 14:57:50.077527 Release 115 dw (???) at 08973808 by _ERMBM+8F6 vmdbk 00100000 SYSTEM
07E001C0 00 14:57:50.077527 Release 9 dw (GSD) at 0125F248 by _ERMBM+8A8 vmdbk 00100000 SYSTEM
07E001A0 00 14:57:50.077527 Rtrn to _ERMBM+778 fr HCPCFR+BE4 cpebk 6F011600 cc0 rc=0 iac Primary
07E00180 00 14:57:50.077527 Rtrn to HCPCFR+9FC fr HCPQCO+9EA cpebk 02443C00 cc1 rc=0 iac Primary
07E00160 00 14:57:50.077527 Stack Call to HCPGRFIC CPE 02449A00 HCPQCO+8D8 vmdbk 065B4000 OPERATOR vmdstate 42
...
```

Tracing...

TRACE Command

- Monitors events in virtual machines
 - Execution of instructions
 - Memory Alteration
 - Register Alteration
 - I/O Activity

Data, I/O, and Guest Tracing

- TRSOURCE and TRSAVE commands
- Data written to System Trace File (TRF)

```
CP TRSOURCE ID TRAP1 SET TRSAMPLE TYPE DATA LOC HCPSPX + C42 41200074
CP TRSOURCE ID TRAP1 SET TRSAMPLE TYPE DATA DL G0:15=REGS
CP TRSOURCE ID TRAP1 SET TRSAMPLE TYPE DATA DL G5.D0=SPFBK
CP TRSAVE FOR ID TRAP1 DASD TO * SIZE 256 KEEP 4
CP TRSOURCE ENABLE SET TRSAMPLE
CP TRSOURCE DISABLE SET TRSAMPLE
QUERY TRF ALL
TRACERED x x x x CMS TRSDATA OUTPUT A
```

Performance Data – MONITOR

What is Monitor data?

- Binary data describing system configuration & activity
- Consists of monitor records, each tied to a specific system event or metric.
 - There are two primary types:
 - Sample - periodic summaries of user/device activity
 - Event - specific actions as they occur.

Who creates it?

- Control Program (CP) – system-level behavior
- Guests – their own workloads
- The CP privileged command, MONITOR, controls monitoring, including the type, amount, and nature of data to be collected.

How do I collect it?

- Collected using IBM's MONWRITE utility
- For simple instructions & best practices on collecting monitor data via the MONWRITE utility, see <https://www.vm.ibm.com/devpages/bkw/monsimp.html>

Summary



Summary

VM's Control Program (CP):

- Efficiently manages the environment it is running in
 - LPAR
 - Virtual Machine
- Manages processors, memory, and devices among virtual machines
 - Efficiently shares available resources to meet virtual machine requirements
 - Virtualizes resources for use by virtual machines
- Preserves and restores data across system IPLs
- Provides virtual networking and connectivity capability
- Records Diagnostic Information
 - Several types of data
 - Many ways to collect it

Additional References

z/VM Library

- <https://www.vm.ibm.com/library/>

IBMVM Listserver

- <https://listserv.uark.edu/scripts/wa-UARKEDU.exe?A0=IBMVM>

z/VM Support

- <https://www.vm.ibm.com/support/index.html>

z/VM Education

- <https://www.vm.ibm.com/education/>

Virtualization Cookbook for IBM Z (z/VM 7.2)

- <https://www.redbooks.ibm.com/abstracts/sg248147.html>



