

# Mainframe Topics

## Kapitel 4

# Virtualisierung auf dem Mainframe

# CMS – Conversational Monitor System

**Frank Heimes**

Senior IT Architect & Technical Lead for Ubuntu on z  
Canonical Ltd.

# Objectives

---

- List z/VM's base components and describe how they work together
- Describe CMS and the tasks it can accomplish
- State why CMS Pipelines are an important feature of z/VM
- Describe CMS Application Multitasking
- Describe the XEDIT environment and its purpose

# Introducing z/VM's Base Components

---

- Control Program (CP)
  - ▶ A component that manages the resources of a single system to make it appear that multiple computing systems exist
- Conversational Monitor System (CMS)
  - ▶ An end-user interface for running user programs
- REXX/VM
  - ▶ A programming language that allows you to write customized application programs and command procedures
- Group Control System (GCS)
  - ▶ A virtual machine supervisor that provides multitasking services

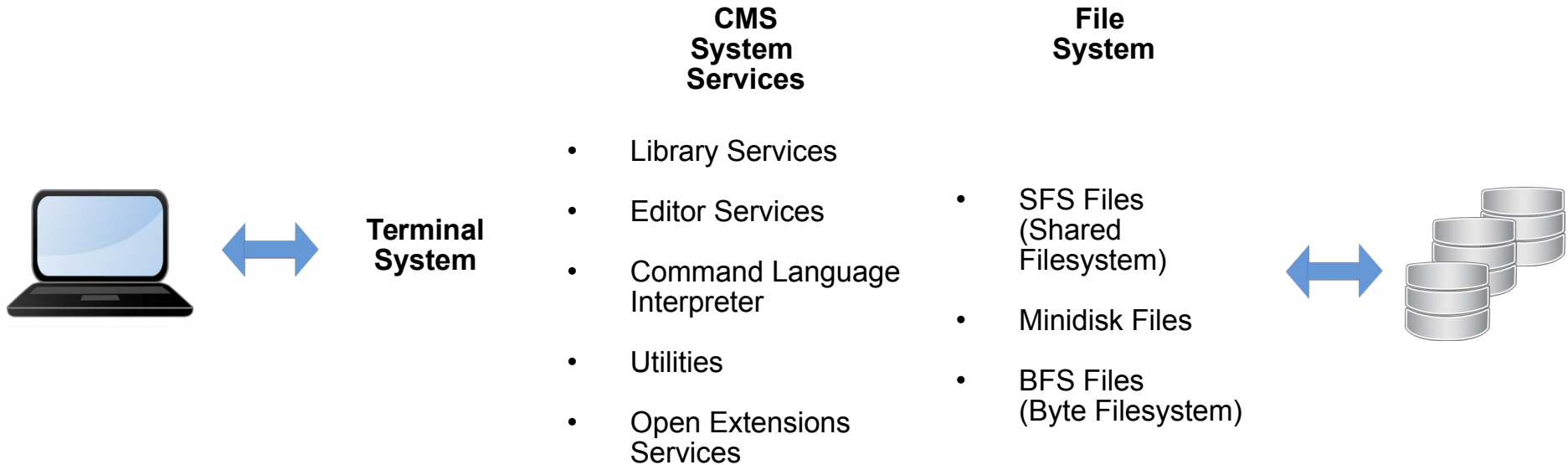
# What is CMS?

---

- Conversational Monitor System is an operating system environment itself
- Over time CMS became a part of VM
- CMS is a single user operating system, which makes it possible to:
  - ▶ Create and maintain files
  - ▶ Write and execute application programs
- CMS communicates with users through the console

# Structure of CMS

---



*Figure 1. CMS System Structure. This consists of the Terminal System, the File System and the CMS System Services.*

# CMS Pipelines

---

- Provide a rich and efficient set of functions that can be used to solve large problems by breaking them up into smaller programs
- Smaller programs are called stages
- Stages:
  - ▶ Read data
  - ▶ Filter and refine data
  - ▶ Combine multiple data items

# CMS Pipelines

---

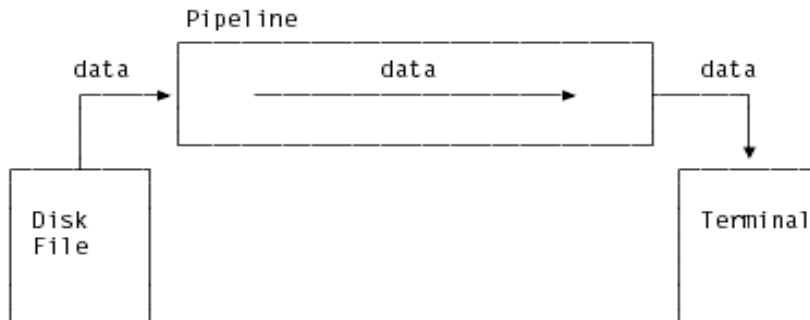


Figure 1. Data Flowing through a Pipeline

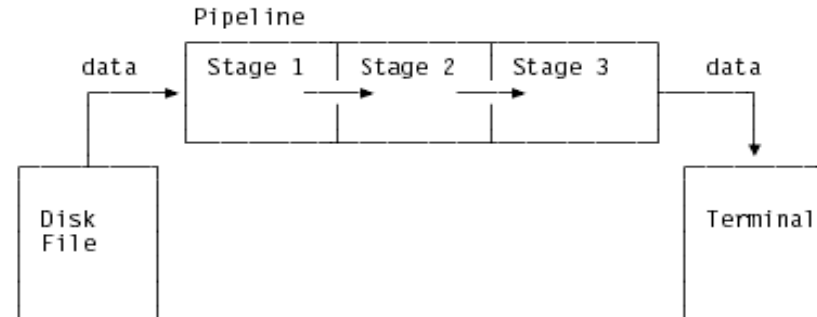


Figure 2. Stages within a Pipeline



Figure 3. Records Flowing through a Stage

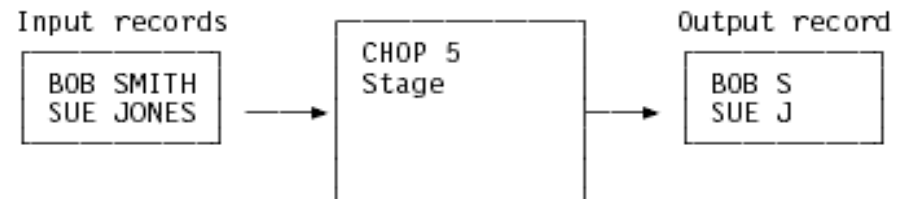


Figure 4. Records Flowing through a CHOP Stage

# Pipeline Example

---

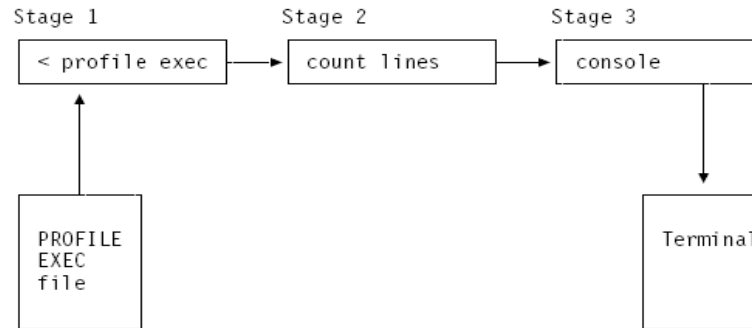


Figure 9. Map of Your First Pipeline

**pipe < profile exec | count lines | console**

- This command executes the number of lines contained within your PROFILE EXEC file.

# XEDIT

xedit dummy1 txt

The screenshot shows the XEDIT text editor interface. At the top, a status line displays file information: "DUMMY1 TXT A1 F 80 Trunc=80 Size=0 Line=0 Col=1 Alt=0". Below this, a message line says "Creating new file:". The main editing area is mostly blank. A prefix line contains four green equals signs. The current line (line 1) is marked with a green cursor and contains the text "=== \* \* \* Top of File \* \* \*". The scale (line 2) is marked with a green cursor and contains the text "|...+...1...+...2...+...3...+...4...+...5...+...6...+...7...". The status line at the bottom left shows "====> " followed by a green cursor. The status line at the bottom right shows "X E D I T 1 File" and "042/007".

```
DUMMY1  TXT  A1  F 80  Trunc=80 Size=0 Line=0 Col=1 Alt=0
Creating new file:

3

4
==== * * * Top of File * * *
|...+...1...+...2...+...3...+...4...+...5...+...6...+...7...
==== * * * End of File * * *

7
====>

X E D I T 1 File
042/007
```

- 1: File Identification Line
- 2: Message Line
- 3: File Area
- 4: Prefix Area
- 5: Current Line
- 6: Scale
- 7: Commandline
- 8: Status Area

# Basic xedit Commands

---

- Open file:  
xedit <filename> <type>
- Start editing:  
====> input  
(2 x Enter: leave input mode)
- Save file:  
====> file
- Quit xedit:  
====> quit
- Quit xedit without save:  
====> qquit
- 
- Delete single line:  
==d== <Enter>
- Delete block of lines:  
==dd= ...  
==dd= <Enter>
- Navigate to first line:  
====> top
- Navigate to last line:  
====> bottom
- Search & replace words:  
====> Change/old/new  
PF5,PF6

# The Batch Facility

---

- Can take over both short and long processing jobs for you
- Frees up your time to continue working at your terminal
- Two examples:
  - Have the facility format text and send it to the printer, instead of doing it yourself
  - Have large jobs run throughout the night to take advantage of lower computing costs

# z/VM Help Facility

---

- CMS Help Facility provide assistance for:
  - ▶ Tasks
  - ▶ Commands and options
  - ▶ Subcommands
  - ▶ REXX statements
  - ▶ Callable routines
  - ▶ Pipeline stages
  - ▶ Assembler language macros
  - ▶ Messages

# z/VM Help Facility

==> help query

```
COMMANDS QUERY                      All Help Information                      line 1 of 254
(c) Copyright IBM Corporation 1990, 2009

QUERY

>>--Query--operands--.-FIFO-.-----><
                        '-(---| Options |---.-'
                        '---'

Options:
      .-FIFO-.
|---.-STACK-+-----+-----|
|   '-LIFO-'   |
| -FIFO-----|
| -LIFO-----|

Authorization

General User

Purpose

Use the QUERY command to gather information about your CMS virtual machine.
You can get information about:

    o The operation of your virtual machine, such as:

        - The status of virtual machine characteristics that are controlled by
          the CMS SET command


        - The status of CMS/DOS functions

        - The search order for libraries (MACLIBS, TXTLIBS, CSLLIBS, DOSLIBS,
          and LOADLIBS)

        - Information about your saved segments.

    o The status of your files and file pool directories, such as:
PF1=          2= Top      3= Quit    4= Return    5= Clocate    6= ?
PF7= Backward  8= Forward  9= PFkeys 10=          11= Related   12= Cursor

====> █
```



Micro-read 1 File

042/007

# z/VM Help Facility



```
COMMANDS QUERY          Related Help Information          line 1 of 191
Related Information

You can use the various operands of QUERY to get information about
your virtual machine.  For HELP on a particular operand, place the
cursor under the operand and press ENTER or the PF1 key.

ABBREV - Displays status of abbreviation
indicator.
ACCESSED - Displays files and SFS directories
that you have accessed.
ACCESSM0 - Displays whether access of file
mode 0 files is enabled.
ACCESSOR - Displays information about users
currently accessing an SFS directory
with the directory control attribute.
ALIAS - Displays alias information for a
file or alias in an SFS directory.
APL - Displays the status of APL
character code conversion.
AUTHORITY - Displays file and directory
authorities.
AUTODUMP - Displays status of the SET AUTODUMP
command.
AUTOREAD - Displays status of the console
read.
BLIP - Blip facility is not supported by
CMS.
BLOCKS - Displays how much space an SFS
file is using, and how much the
system is using to store it.
BORDER - Displays status of the window
borders.
CHARMODE - Displays whether character or field
attributes are used.
CMSLEVEL - Displays licensed program
information.
PF1= Help    2= Top    3= Quit    4= Return    5= Clocate    6= ?
PF7= Backward 8= Forward 9= PFkeys 10= Morehelp 11=          12= Cursor

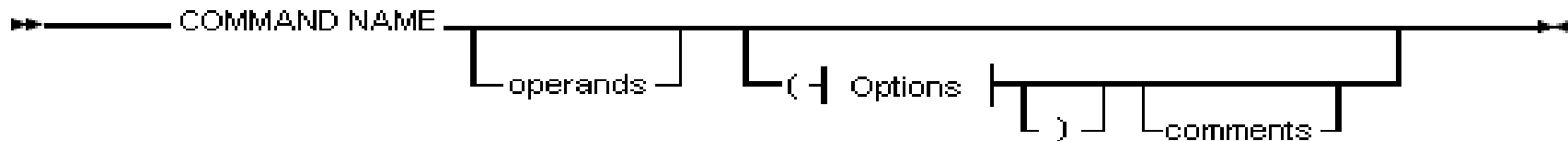
====> █

Macro-read 1 File
042/007
```

# Syntax Diagrams (basics)

---

- Read syntax diagrams from left to right and from top to bottom
- This side module shows you descriptions and examples



- Example:

```
query user42
USER42      -L0003
Ready; T=0.01/0.01 22:04:27
```

```
q user42 ext
USER42      -L0003 HOST TCPIP      FROM 10.31.2.2
Ready; T=0.01/0.01 22:04:31
```

---

# CMS Commands

# QUERY – help query

```
COMMANDS QUERY                      All Help Information                      line 1 of 254
(c) Copyright IBM Corporation 1990, 2011

QUERY

>>--Query--operands-----><
      |--! Options !--:--:--:
Options:
!-- --STACK-- --FIFO-- -----!
  |-- --LIFO-- --LIFO-- -----!
  |-- --FIFO-- --FIFO-- -----!
  |-- --LIFO-- --LIFO-- -----!

Authorization

General User

Purpose

Use the QUERY command to gather information about your CMS virtual machine.
You can get information about:

o The operation of your virtual machine, such as:

  - The status of virtual machine characteristics that are controlled by
    the CMS SET command

  - The status of CMS/DOS functions

  - The search order for libraries (MACLIBs, TXTLIBs, CSLLIBs, DOSLIBs,
    and LOADLIBs)

  - Information about your saved segments.

o The status of your files and file pool directories, such as:
PF1=          2= Top          3= Quit          4= Return          5= Clocate          6= ?
PF7= Backward 8= Forward    9= PFkeys       10=          11= Related         12= Cursor

====> _

Macro-read 1 File

MA A 42/007
```

# QUERY – Usage Notes

---

The following are the **operands** available with QUERY:

ABBREV	FILEPOOL	NAMEDeF	DOS	LIMITS	TRACECTL
<b>ACCESSED</b>	CONNeCT	NONDISP	DOSLIB	LINeND	TRANsLate
ACCESSM0	FILEPOOL CURRENT	OLDCMDS	DOSLNCNT	LOADAREA	TRAPMSG
ACCESSORs	FILEPOOL	OPTION	DOSPART	LOADLIB	TVICALL
ALIAS	DISable	OSTXTBUf	ENROLL	LOCATION	TXTLIB
APL	FILEPOOL PRIMARY	OUTPUT	ETRACE	LOCK	UPSI
AUTHority	FILESPACE	PROTECT	EXETrACe	LOGFILE	VSCREEN
AUTODUMP	FILEWAIT	RDYMSG	FILEATTR	MACLIB	WINDOW
AUTOREAD	FULLREAD	RECALL	FILEDeF	MACLsubs	WMPF
BLIP	FULLSCREen	REDTYPE	FILEDeF	MACLsubs	WMPF
BLOCKS	GEN370	RELPAge	FILEPOOL CONFLiCT	WORKSTation	<b>CMSLEVEL</b>
BORDER	GETMAIN	REMOTE			
CHARMODE	ESCAPE	RORESPECT			
CMSPF	IMPCP	ROUTE			
CMSREL	IMPEX	SEARCH			
CMSTYPE	INPUT	SEGMENT			
CMS370AC	INSTSEG	SERVER			
COMDIR	KEY	SHOW			
CSLLIB	KEYPROTeCT	STORECLR			
CURSOR	LABELDEF	SYNONYM			
DIRATTR	LANGLIST	SYSNAMES			
<b>DISK</b>	LANGUage	TAPECSL			
DISPLAY	LDRTBLS	TAPENEVR			
DLBL	LIBRARY	TEXT			

But what about the operands:

**PRIVCLASS** and **NAMES** and **CLEVEL** ?

The commands

**q privclass**

and

**q names**

and

**q clevel**

also work fine !

# ACCESS

```

      .-topdirectory as fm A-.
      |-0191 as fm A-----|
>>--ACcess--+-----+-----><
      |-| Path A |-----|
      '-| Path B |-----'

```

## Path A:

```
|--dirid--fm-- .----- .----- .----- .----- |
|          (1) |   |      (2)                                     |
|'-/ext-----'   |'-(------ .----- .----- .----- .'
|                                   '-NOPROF-'   | -FORCERO- |   '-'-) -'
|                                           '-FORCERW-'
```

## Notes on the Syntax:

(1) Do not specify a space between fm and /ext. (2) You can enter Options in any order between the parentheses.

Use either with a directory ID (dirid, SFS) or a device number (vdev, Minidisk).

# ACCESS

---

## Path B:

```
|--vdev--fm--.----->
|      (1)  .-*--*--*-----.|
|      '-/ext-----+-----+'
|              .-*--*-----.|
|              '---fn-----+-----+'
|                      .-*--. |
|                      '---ft-----+-----+'
|                              '-fm-'

>--.-----|
|      (2) |
|      '-(-----| Options B |--.---.-'
|                      '---)-'
```

## Options B:

```
| .----- .----- .----- .----- .----- |
| '-NOPROF-' | -ERASE---- | '-NODISK-' | '-MODE0-' |
|              | -SAVEONLY- |
|              '-NOSAVE---'
```

# ACCESS – Usage Notes

---

- Using the ACCESS command with a directory ID or a device number
  - With and without a file pool
- Using the ACCESS command with a directory ID
  - Determining status by ownership
- Using the ACCESS command with a virtual device number
  - Access can create a file directory in your virtual machine when one is not present

# RELEASE

```
>>--RELease--.vdev--.-----><
      |dirid|  '-(-----.-.-.-'
      '-fm----'      '-DETach-'  ')-'
```

- The RELEASE Command is used to free an accessed disk that was previously accessed with the ACCESS Command.
- Example:

```
▶ release 0293
```

# RELEASE

```
>>--RELease--.-vdev-.-.--.------><  
      | -dirid-|    '-(---.-----.-.-.-'-  
      '-fm----'          '-DETach-'     '-'-)
```

- If you want to release and detach the 498 disk that is accessed as your file mode b, then issue:

```
release 498 (det OR
```

```
▶ release b (det
```

- To just release the disk currently accessed as file mode c, issue:

```
▶ release c
```

# An example of daily work

Lia (owner)



MDISK 191 3390 3359 50 LX0001 MR

VDEV

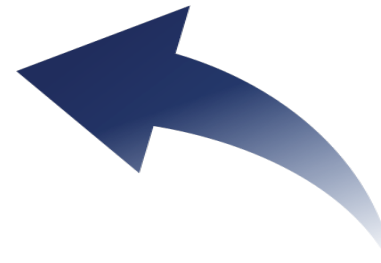
STARTNG  
CYLINDER

VOLID

MODE

DEVTYPE

NUMBER OF  
CYLINDERS



George (sharing)



LINK LIA 191 1191 RR

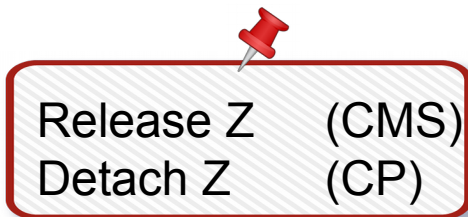
/\* make it available \*/

ACCESS 1191 Z

/\* to use it \*/

RELEASE Z (DETACH

/\* similar to umount \*/



# Conclusion

---

- CMS – Conversational Monitor System
  - ▶ Is a single user operating system that runs within z/VM
- CMS tasks include writing, testing, and debugging application programs to be used by CMS or z/VM guest systems
- CMS runs the full-screen editing facility called XEDIT
- The CMS help facility is a CMS Command that can be accessed when help is needed by entering:
  - ▶ `help cms`

# Backup

---

# Glossary

---

**Conversational Monitor System (CMS)**- A component of z/VM that runs in a virtual machine and provides both the interactive z/VM end-user interface and the general z/VM application programming interface. CMS runs only under the control of the z/VM Control Program (CP).

**CMS Pipelines**- CMS job control product for z/VM that enables complex tasks to be specified and executed. CMS Pipelines has three parts – a command parser, a library of built-in programs, and a dispatcher.

**Callable Services Library**-A package of CMS routines that can be stored as an entity and made available to a high-level language, REXX, or an assembler program.

# Glossary

---

**File pool** - A collection of minidisks managed by a file pool server. It contains user files and directories and associated control information. The files and directories for many users can be contained in a single file pool.

**Group Control System (GCS)**- A component of z/VM, consisting of a named saved system that the user can IPL and run in a virtual machine. It provides simulated MVS services and unique supervisor services to help support a native SNA network.

**Initial Program Load (IPL)**- The process of loading an operating system into a machine

**OpenExtensions Services**- The VM implementation of three POSIX standards

# Glossary

---

**File pool**-A collection of minidisks managed by a file pool server. It contains user files and directories and associated control information. The files and directories for many users can be contained in a single file pool.

**Group Control System (GCS)**- A component of z/VM, consisting of a named saved system that the user can IPL and run in a virtual machine. It provides simulated MVS services and unique supervisor services to help support a native SNA network.

**Initial Program Load (IPL)**- The process of loading an operating system into a machine

**OpenExtensions Services**- The VM implementation of three POSIX standards

# Glossary

---

**File pool**-A collection of minidisks managed by a file pool server. It contains user files and directories and associated control information. The files and directories for many users can be contained in a single file pool.

**Group Control System (GCS)**- A component of z/VM, consisting of a named saved system that the user can IPL and run in a virtual machine. It provides simulated MVS services and unique supervisor services to help support a native SNA network.

**Initial Program Load (IPL)**- The process of loading an operating system into a machine

**OpenExtensions Services**- The VM implementation of three POSIX standards

# Glossary

---

**OpenExtensions Shell and Utilities**- provides application development tools and an interactive environment in support of the POSIX application environment.

**REXX/VM**- (REstructured eXtended eXecutor programming language) processes English-like commands.

**XEDIT**- A full-screen editing facility that runs under CMS.