

SHARE

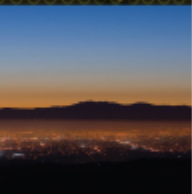
Technology • Connections • Results

z/VM Installation -- It's Installed, *NOW* What?

(Or: What we learned “on the job”)

Session 9120
16:30 Tuesday, August 12, 2008
SHARE – San Jose, CA

Contact: [Mike.Walter – Hewitt.com](mailto:Mike.Walter@Hewitt.com)
847.771.9212



Disclaimer

(you expected something else?)



- How *THIS* customer does it...
... after years of learning from others.
- **Your Mileage May Vary.**
- This session will probably re-appear with significant improvements at the next SHARE.
 - After other “old timers”, ***and YOU***, chip in ideas, tell me what I’ve done wrong (still), what I should do differently, and what else I should be doing!
 - “**SHARE**, it’s not an acronym – it’s what we **DO!**”

Mandatory “Brag Sheet”

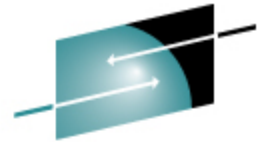
or: “Why should we listen to *him*?”



- Started as a “Teleprocessing Operator” in Feb. 1972
 - Thus, in Data Processing for over 37 years.
- VM-exclusive (mostly) since 1978 (30+ years of VM experience)
 - (VM 370 Release 5 Program Level Change 6)
- Installed VM at Hewitt Associates in early 1985
- Chaired CAVMEN for 6+ years
- Member of SHARE VM Technical Steering Committee since August 1997
- According to envelopes of various mailings: CIO; Partner-In-Charge; and Chief Cook, Manager of VoiceMail Systems

Agenda

(Your mind can't absorb what your butt can't endure
vs. Long Session Slots)



S H A R E
Technology • Connections • Results

- Where can I look for help?
- CP Directory
- CP_Owned DASD
- “SYSTEM CONFIG” file
- Backup and Restore
- What to document “just in case”
- Minidisks vs. Shared File System
- Best practices (CYA)
- Miscellany

Where do I look for (more) help?

- Obviously: IBMLink (leverage your support dollars):
 - <https://www-304.ibm.com/usrvvc/account/userservices/jsp/login.jsp?persistPage=true>
- IBM Publications (review them yourself back at the shop):
 - <http://www.vm.ibm.com/library/> (The VM library), which points to...
 - http://publibz.boulder.ibm.com/bookmgr_OS390/libraryserver/zvmv5r3/ (The z/VM Library Center)
 - http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/Shelves/hcsh2a90 (The z/VM Bookshelf)
 - <http://www-03.ibm.com/systems/z/os/zos/bkserv/zvm.html> (The z/VM Collection” bookshelf – Discs)
 - <http://www-03.ibm.com/systems/z/os/zos/bkserv/zvmpdf/> (The z/VM PDF Library)
 - Program Directories, data areas, control blocks, monitor records, tri-fold Installation Summaries, redbooks, whitepapers, reports, spec sheets, brochures, reference guides, and lots more
 - <http://www.elink.ibm.com/publications/servlet/pbi.wss?CTY=US> (the IBM Publications Center)
 - <http://www-03.ibm.com/systems/z/os/zos/bkserv/lookat/> (“LookAt” messages)
- Informal but VERY prompt, helpful, and flameless discussion lists:
 - Send email text to: LISTSERV@LISTSERV.UARK.EDU
 - **SUBSCRIBE IBMVM** firstname lastname
 - Send email text to: LISTSERV@VM.MARIST.EDU
 - **SUBSCRIBE CMS-PIPELINES** firstname lastname
 - Send email text to: LISTSERV@VM.MARIST.EDU
 - **SUBSCRIBE LINUX-390** firstname lastname

Where else can I look IMMEDIATELY?

You are in luck, you are attending ***the best interactive training value*** by being right here ...
at SHARE. Sessions still to come this week ...

- This page used to contain a list of 25 or so good SHARE sessions for those new to z/VM or Linux
- Good news/Bad news: When preparing the slide for this presentation the list accumulated to over 108 sessions.

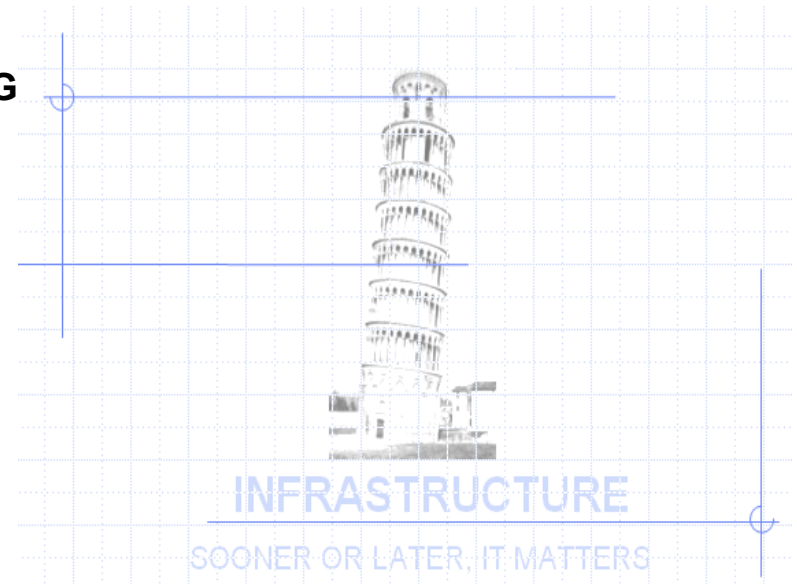
Summary: You're getting lots for your money.

CP Directory

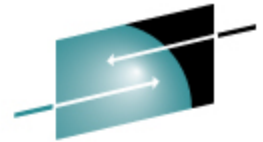
- The “**USER DIRECT**” file - where VM users (virtual machines of all types) are defined.
 - Documented in z/VM CP Planning and Administration manual.
 - After a z/VM installation, located on MAINT’s 2CC minidisk.

- Most basic statements:

```
USER jqpublic password 64M 2047M G  
ACCOUNT acntcode distcode  
ACIGROUP as_given  
MACHINE XA  
IPL CMS PARM AUTOCR  
CONSOLE 009 3215 C  
SPOOL 00C 2540 READER *  
SPOOL 00D 2540 PUNCH D  
SPOOL 00E 1403 P  
LINK MAINT 190 190 RR  
LINK MAINT 19D 19D RR  
LINK MAINT 19E 19E RR  
MDISK 0191 3390 beg_cyl num_cyls volser MR
```

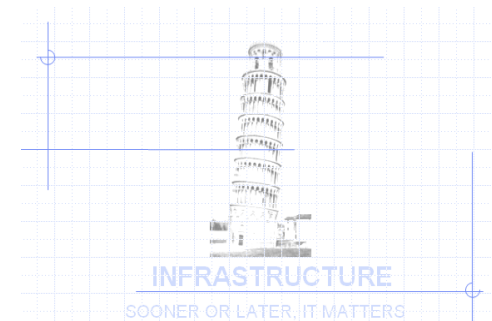


- After making any change, ALWAYS check for minidisk overlaps.
 - It is **very** easy to define more than one minidisk using all or some of the same cylinders used by another minidisk.
 - Results in “one-way” file encryption.
 - **Run the DISKMAP and/or DIRMAP commands, carefully checking the output for overlaps.**
 - DISKMAP and DIRMAP also report “Gaps”, which are free cylinders upon which you can allocate minidisks.
 - DIRECTXA (compiles the “USER DIRECT” file) does not report minidisk overlaps (since they may be valid).



CP_Owned DASD

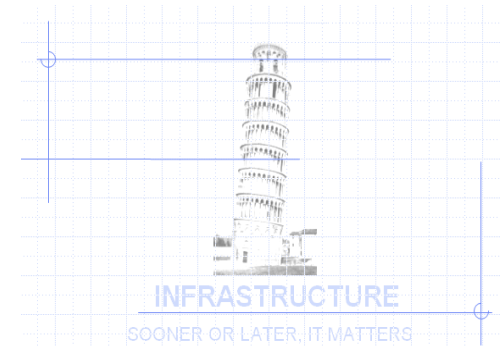
- All DASD attached to VM for VM's use must be **INIT**'ed by ICKDSF with parameters: "**CPVOLUME**", "**MODE(ESA)**" and "**NOFILLER**".
- The CPFMTXA EXEC is a nice front-end to the ICKDSF MODULE, defaulting to MODE(ESA) and NOFILLER, installing a "no room at the inn" dummy VTOC.
- Cylinder zero: home of the volume Allocation Bit Map
 - Tells CP how each cylinder is used, and in some cases if it is active.
 - Allocations: PERM, PARM, DRCT, PAGE, SPOL, TDSK
 - ALWAYS allocate cylinder zero as PERM to protect the Allocation Bit Map.



CP_Owned DASD

The “SYSTEM CONFIG” file defines which DASD are attached to the system when it is IPLed.

- Some DASD are critical ... pretty much all DASD with allocations other than PERM, and even some of those.
- SPOOL and PAGE DASD are *very* difficult to get offline on a running system.
 - You can prevent further allocation to them, but cannot cause SPOOL and PAGE cylinders to be brought back into storage to be moved to another DASD. Plan for system outages to remove SPOOL or PAGE DASD. Any type can be added on-the-fly, without an IPL.



The “SYSTEM CONFIG” file;
... and ...
Best practices;

- **Tip:** SPOOL volumes in first Slots
 - Any change in existing SPOOL volume **SLOT order will** result in a **loss of all SPOOL files on that and trailing SPOOL volumes.**
 - You can always add more SPOOL slots lower down (that’s why it is good to have RESERVED slots up top specifically for future SPOOL volume use.
 - Linux virtual machines will require more SPOOL due to DCSS usage – plan for it. (DCSS’ are stored in SPOOL)
 - You can remove completely-empty SPOOL volumes (difficult to get completely empty), as long as they do not affect the SLOT NUMBER order of the remaining SPOOL volumes.

The “SYSTEM CONFIG” file; ... and ... Best practices;



When faced with a file similar to the following, what's your first impulse?

```
CP_Owned Slot 1 VMPP01 OWN
CP_Owned Slot 2 VMR51A OWN
CP_Owned Slot 3 VMPP08 OWN
CP_Owned Slot 4 VMDUMP DUMP
CP_Owned Slot 5 VMTD01 OWN
CP_Owned Slot 6 VMPG05 OWN
CP_Owned Slot 7 VMPG01 OWN
CP_Owned Slot 8 VMPP06 OWN
CP_Owned Slot 9 VMPG07 OWN
CP_Owned Slot 10 VMPP07 OWN
CP_Owned Slot 11 VMR51B OWN
CP_Owned Slot 12 VMTD02 OWN
CP_Owned Slot 13 VMSP01 OWN
CP_Owned Slot 14 VMPG02 OWN
CP_Owned Slot 15 VMPP04 OWN
CP_Owned Slot 16 VMPP05 OWN
CP_Owned Slot 17 VMSP02 OWN
CP_Owned Slot 18 RESERVED
CP_Owned Slot 19 RESERVED
CP_Owned Slot 20 RESERVED
```

Why... sort it by volser, of course!

And then... change the SLOT numbers back into a nice, neat serially sequential list starting at 1.

BAD IDEA! At the next IPL you will lose all SPOOL files! Why? -- You changed the SLOT numbers of the SPOOL volumes (in this case fairly obvious VMSP01, VMSP02 – but maybe not always so obvious).

The "SYSTEM CONFIG" file; ... and ... Best practices;



SHARE

Technology • Connections • Results

Place SPOOL volumes up front...

CP_Owned	Slot	1	VMSP01	OWN	STRONGLY suggested -- spool volumes in FIRST Slots!
CP_Owned	Slot	2	VMSP02	OWN	STRONGLY suggested -- spool volumes in FIRST Slots!
CP_Owned	Slot	3	VMSP03	OWN	STRONGLY suggested -- spool volumes in FIRST Slots!
CP_Owned	Slot	4	VMSP04	OWN	STRONGLY suggested -- spool volumes in FIRST Slots!
CP_Owned	Slot	6	RESERVED		Reserved for another SPOOL volume (or anything)
CP_Owned	Slot	7	RESERVED		Reserved for another SPOOL volume (or anything)
CP_Owned	Slot	8	RESERVED		Reserved for another SPOOL volume (or anything)
CP_Owned	Slot	9	VMDUMP	DUMP	Still a SPOOL volume, convenient if you have enough DASD so that contiguous DUMP space is always found.
CP_Owned	Slot	10	VMR52A	OWN	Example of System Residence Product volumes
CP_Owned	Slot	11	VMR52B	OWN	
CP_Owned	Slot	12	VMTD01	OWN	Example of Temp-disk volumes
CP_Owned	Slot	13	VMTD02	OWN	
CP_Owned	Slot	14	RESERVED		
CP_Owned	Slot	15	RESERVED		
CP_Owned	Slot	16	RESERVED		
CP_Owned	Slot	17	RESERVED		
CP_Owned	Slot	18	RESERVED		
CP_Owned	Slot	19	RESERVED		
CP_Owned	Slot	20	VMPP01	OWN	Example of Program Product volumes
CP_Owned	Slot	21	VMPP02	OWN	
CP_Owned	Slot	22	VMPP03	OWN	
CP_Owned	Slot	23	VMPP04	OWN	
CP_Owned	Slot	24	VMPP05	OWN	
CP_Owned	Slot	25	VMPP06	OWN	
CP_Owned	Slot	26	VMPG01	OWN	Example of Paging volumes
CP_Owned	Slot	27	VMPG02	OWN	
. . .					

“SYSTEM CONFIG” file; Copies on MAINT’s CF1, CF2, and CF3 disks

- Backup (copy) the primary ‘PARM’ disk (MAINT’s CF1) to MAINT’s backup CF2 disk.
 - **Tip:** It is important to keep the CF1 and CF2 disk in synch every time you update the CF1 (or have any backout plan, see notes).
 - Should the CF1 become ‘damaged’, you can IPL using the CF2 disk.
 - **Tip:** RENAME files on the CF1 and CF2 before copying replacement files.
 - (e.g. RENAME CPLOAD MODULE x -1CPLOAD MODULE x)
 - Just after installation, I like to COPY CPLOAD MODULE x CPLDSO MODULE = (OLDDATE)
 - Consider backing up CF1 to CF3 (DDR copy) before changes, then keeping CF1 and CF2 “in sync” -- two ways to back out changes!
- **ALWAYS run “EXEC CPSYNTAX”** after every change to the “SYSTEM CONFIG” file.

Backup and Restore; What to back up?

- To steal Bit Bitner's standard performance answer:
"It Depends"
- For a **test** system – perhaps only things you are likely to change, esp. those that can prevent an IPL.
- For a **production** system:
 - For D.R. purposes: a CKD-format (e.g, DDR, VM:Backup Physical, FDR) backup of sysres DASD, program product DASD.
 - Normal file-level backups of everything.
 - Separate backup of the USER DIRECT file.
 - *Good idea to DIRECTXA USER DIRECT to DRCT cyls on second sysres DASD.*
 - Hardcopy reports from DIRMAP and/or DISKMAP utilities (on the S-disk).
 - Send hardcopy reports offsite, too!

What to document “just in case”; The worst happens: What’s Where?



- Know where your PARM cylinders are (so you can point to them with SALIPL).
- Know where your source directory cylinders are.
 - Pick a DASD volume and cylinder address you will remember without a hardcopy listing. You already ***KNOW*** that hardcopy will be **missing** in a real disaster. (E.g. VMPP01, cylinder, 1 may begin the source directory minidisk).
- Know where your backup catalog (e.g. VM:Backup 1B0) cylinders are.
 - Again, perhaps volser VMPP02, beginning at cylinder 1.
- Memorize the syntax:
 - **CP DEFINE MDISK AS vdev start count|END volser**

What to document “just in case”, Even worse happens!:: Stand-Alone IPL-able Utility



Tape(s)

- Create Stand-Alone IPL-able Utility Tape (ICKDSFSA, DDRXA, etc.)
 - May have been done by the UTILITY command when executing “Store a Backup Copy of the z/VM System to Tape” during your z/VM installation. **If not – create one!**
 - A single tape requires at least one IPL from the HMC to access each program.
 - There’s a pause (sometimes long, sometimes requiring “ENTER” to be pressed on the terminal) between each IPL.
 - You may want to create separate IPL-able tapes for each utility.
 - Advantage: a single IPL, making “HMC time” a little less frustrating.
 - You can test these tapes from a VM userid with a tape drive ATTACHed.
 - Just enter the command: CP IPL vdev CLEAR (where “vdev” is the *VIRTUAL* tape drive *address* where the tape is already mounted and ready).
- **Have a hardcopy manual available which explains each utility (and try them out ahead of time!).**

What to document “just in case”; How as DASD allocated?



- It has helped us more than once to have weekly hardcopy printouts of:
 - The Allocation Bit Map for every system-level DASD
 - Our user minidisks are all on DASD allocated as PERM space.
 - The CP Directory “USER DIRECT” file, or other maps.
 - Reports generated by VM:Secure, DIRMAINT, RACF/VM, or other External Security Managers (ESMs).
- Useful at Disaster Recovery sites, too.
 - Send a copy offsite!

Minidisks vs. Shared File System, for CMS-intensive customers.



- Minidisks:
 - Easy to allocate (except for the overlap potential).
 - Easy to back up.
 - Low administrative overhead.
 - Do not allow easy space expansion.
 - Need all users to release the old minidisks, allocate a new minidisk, format it, copy the files from the old minidisk, permit users to access the newly-sized minidisk, delete the old minidisk.
- Shared File System
 - Easy to allocate and increase or decrease user space.
 - More difficult to back up without a VM:Backup-like product.
 - Requires much more Administrative effort, especially without the SafeSFS product (from Safe Software, Inc.)

Best practices; Change Identification/Backout



- Learn “What Mother Never Told You About VM Service”:
 - **Rule Number 1: “NEVER CHANGE ANYTHING IBM SENDS YOU”!**
 - Instead **change a copy on another disk**, or copy the IBM version to another ‘instant backup’ name on the same disk and change the original file.
 - **Rule Number 2: “Keep your stuff separate from IBM’s”.**
 - **See:** “What Mother Never Told You About VM Service” (1983).
 - <http://pucc.princeton.edu/~melinda/tutorial.pdf>
 - A bit outdated, but valuable insight.
 - Basics of Control File, Aux File, Update Files remain the same.

Best practices; Change Identification/Backout



S H A R E

Technology • Connections • Results

- You always want a means (which does not require a tape mount!) to see what IBM sent originally.
 - Your problem may keep you from mounting or reading a tape, which can prevent restore of that changed file.

Best practices; CMS file dates



- When using the **COPYFILE** command, **religiously** use the **OLDDATE** option, unless there is a significant reason to use the current date.
 - Not wise to set COPYFILE default to OLDDATE since that would break many existing (IBM, ISV, and local) procedures.
- Undocumented module DMSPLU will let you change the date and time of a file.
 - From: DMSPLU ASSEMBLE on MAINT's disks

```
PARMS:  FN  FT  FM  DATE  TIME  (ALL REQUIRED, POSITIONAL)
        DATE = MM/DD/YY OR MM/DD/YYYY
        TIME  = HH:MM:SS
```



S H A R E

Technology • Connections • Results

Best practices; When to IPL (religion)

- Keeping VM up without an IPL for years does not necessarily make it “better” than some “Other System”. We already **know** VM will stay up indefinitely!
- It’s certainly not better when:
 - Not everything comes back up after “n+” months (or years) of “incidental” changes that have crept in and become active only after an IPL.
Possibility: many problems *all at once!*
 - Your system crashes due to “core cancer” (perhaps lots of System Data Files with ancient users, or badly behaved applications).
 - Applications keep using obsolete, ACCESSEd minidisk files or System Data Files, wasting more storage and increasing paging..
 - Your operators (or you) can’t remember how to IPL.
 - Self-preservation: Something changed on some Other System.
- Try to schedule regular IPLs when your business permits, but still giving you time to diagnose and repair problems before prime time.

Best practices; Stand Alone Program Loader (SALIPL)



- Create a more meaningful Stand Alone Program Loader (SALIPL).
 - Our IPL from the HMC with “LOADPARAM console_rdev” brings up:

```
STAND ALONE PROGRAM LOADER: z/VM VERSION 5 RELEASE 2.0
DEVICE NUMBER:      0910      MINIDISK OFFSET:    00000000      EXTENT:    1
MODULE NAME:        CPLOAD      LOAD ORIGIN:        1000
```

-----IPL PARAMETERS-----

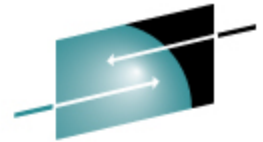
-----COMMENTS-----

Some possible IPL PARAMETERS section (above) entries:

```
CONS=ccuu          PROMPT          <--- Used most often
NOEXITS            FN=system        FT=config
```

- See: `MKSALIPL EXEC` at the end to make one that helps your site.

Best practices; Stand Alone Program Loader (SALIPL)



SHARE
Technology • Connections • Results

- Pressing F9=FILELIST, then F4=SORT(TYPE) results here with a display like:

```
STAND ALONE PROGRAM LOADER: VM/ESA VERSION 2 RELEASE 4.0
FILENAME FILETYPE FORMAT LRECL   RECORDS   BLOCKS   DATE       TIME
-1SYSTEM CONFIG   D1 F       80        721        15 10/10/03   8:14:36
-2SYSTEM CONFIG   D1 F       80        721        15  7/27/03  22:29:00
-3SYSTEM CONFIG   D1 F       80        709        14  7/21/03   9:24:53
-4SYSTEM CONFIG   D1 F       80        709        14  4/25/03  10:48:12
-5SYSTEM CONFIG   D1 F       80        709        14  4/23/03  14:34:52
-6SYSTEM CONFIG   D1 F       80        695        14  4/21/03  12:04:34
-7SYSTEM CONFIG   D1 F       80        671        14  4/15/03  13:36:07
-8SYSTEM CONFIG   D1 F       80        680        14  4/10/03  15:31:15
-9SYSTEM CONFIG   D1 F       80        675        14  4/08/03  16:28:56
SYSTEM   CONFIG   D1 F       80        721        15 12/01/03   8:13:31
DISASTER IMBED    D1 F       80         1         1  7/18/02  10:01:41
HALINVA1 IMBED    D1 F       80         1         1  7/18/02  10:01:51
HALINX1  IMBED    D1 F       80         1         1  7/18/02  10:01:48
RECOVERY IMBED    D1 F       80         1         1  7/18/02  10:01:45
SYSTEM   LOGMSG   D1 F       80         1         1  5/31/95   8:34:39
DEFAULT LOGO      D1 F       78        16         1  3/17/99  10:59:00
various LOGO's    D1 F       78        16         1  3/17/99  11:15:06
SNA      LOGO      D1 F       78        16         1  3/17/99  15:25:39
SNA      LOGOINFO  D1 F       78         6         1  8/03/99  11:01:05
-1CPLOAD MODULE   D1 V     65535      78       1211  8/30/02  15:14:09
-2CPLOAD MODULE   D1 V     65535      77       1207  4/06/01  13:11:26
CPLOAD   MODULE   D1 V     65535      78       1211 11/07/02  14:24:09
CPLDB4VX MODULE   D1 V     65535      77       1199  5/23/00  12:15:23
ICKSADSF MODULE   D2 V     65535      15        218  8/05/99  16:12:07
HAXACO   TRS010HP D1 F       80         86         2  8/31/00  16:30:09
HCPAUS   TRS080HP D1 F       80         19         1  8/31/00  16:29:04
HCPQLN   TRS090HP D1 F       80         21         1  1/19/01  11:11:25
HCPQON   TRS090HP D1 F       80         21         1  3/08/01  10:45:22
HCPQAI   TRS110HP D1 F       80         27         1  4/25/03  10:37:14
3=QUIT  4=SORT(TYPE) 5=SORT(DATE) 6=SORT(NAME) 7=BACK 8=FORWARD 11=SELECT
```

**Best practices;
LOGO files on CONFIG disks
(On MAINT's CF1, CF2 and CF3 disks)**



- Logo files and other samples can be found on MAINT's 2C2 "Samples Disk" (don't miss "DRAWLOGO SAMPEXEC").
- Our "SNA LOGOINFO" contains easily-forgotten instructions as comments:

```
/* Just a reminder that once the SNA logo has been updated, and the */  
/* command: CPACCess MAINT CF1 A SR */  
/* has been entered, you won't see any update on the logo until after*/  
/* command: CP REFRESH LOGOinfo LOGO CONFIG */  
/* and: CP REFRESH LOGOVSM VSCS */  
/* mrw 03/17/99 */
```

- **KEY POINT: *place comments where you're likely to trip over them.***

Best practices; Building a new TEST system.



- Right after any new system has been tested, and before placing it into production)
- Where will you test changes to the new system once it goes into production?
 - DDR copy the new sysres volumes to *test* sysres' (remember to change USER DIRECT, SYSTEM CONFIG, CA product 'DIRECT vdev volser' **records**).
 - **“CP DEFINE MDISK”** *is your friend* - memorize it! Practice beforehand.
 - Understand that DEFINE MDISK always gives R/W access – you can shoot your own foot.
 - If you don't need to write, for DEFINED MDISKS always: ACCESS vdev fm/fm
 - *“fm/fm”* accesses the vdev as a R/O extension of itself.
 - Be sure the test system works *before* migration.
 - It can help you back out errors in production.
 - It can be IPLed on the bare metal (if you did it right).

Best practices; Building a new TEST system.



- Define a new 1st Level userid
 - In this presentation called “VM520”, use VM530 or whatever fits.
 - See examples at the end of handouts.
 - Privclasses “**B,F,G**” (we did not find a need for C, D, or E) VM userid.
 - B = System Resource Operator.
 - *Needed only if VM520 cannot issue site-defined TAPE MOUNT commands.*
 - C = System Programmer.
 - *You are one; but potentially dangerous and not needed for an installation.*
 - D = Spooling Operator (system-wide, ALL spool files).
 - *Not needed and DANGEROUS!*
 - Define a 191 mdisk for local files (PROFILE EXEC, etc.)

Best practices; Building a new TEST system.

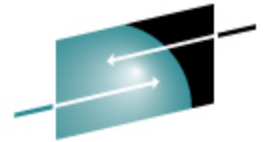


- Have AUTOLOG1 (or 2) bring up your VM520 system 2nd level for user testing every time you IPL production.
- See at the end of this handout:
 - Sample directory entry for userid **VM520**

Miscellany:
Transferring files between Level 1 and Level 2
Level 1 = Production, Level 2 = test system)



- Many ways (SENDFILE, FTP, SFS, etc.) – one easy example:
 - Define userid 'XFER' (or some other easy-to-identify ID) on Level 1 and Level 2 with a minidisk sharing the same extents, on both systems, on the same DASD.
 - “Your gun, your foot” ... What me worry?
 - Normally “one way encryption” (both system can write to the disk at the same time).
 - In this case, you're **only copying** data, you still have the original files on the source system minidisk.



Miscellany:

LOGON/LOGOFF FORCE PENDING

Hung users – getting doc, and saying goodbye.

- On rare occasions a userid will get “hung” in a “Logoff/Force Pending” state, being unable to complete logon or logoff. This is usually due to an incomplete I/O, but not always.
- Over the years, compliments of VM sysprogs on VMSHARE and the VMESA-L discussion lists, the following list was put together (some commands require more than privclass “G”):
 - If you have the time, simply waiting 15 minutes for CP to perform housecleaning chores might free the userid, completing the LOGOFF or FORCE.
 - Use the public domain utility "TRACK" to determine if the userid is awaiting completion of an I/O to a particular unresponsive device.
 - *Use the commands:*
 - `TRACK hungid DEV CLASS * IO PENDING`
 - `TRACK hungid DEV CLASS * IO ACTIVE`
 -
 - *Nota bene: As of 23 Feb 2006 TRACK can be obtained from:*
 - <http://vm.marist.edu/track/>

Miscellany: LOGON/LOGOFF FORCE PENDING Hung users – getting doc, and saying goodbye.

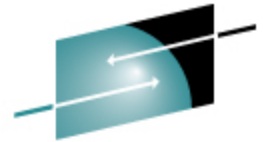


- **GETTING DOC *first*:** Before attempting anything that actually changes the hung userid, if you can (consider communication time-outs which may occur that could affect other users) before muddying the waters, get a current system dump for IBM to diagnose later. From a privclass "A" user: From a privclass "A" user:
 - *CP QUERY DUMP* (then ensure that it is going to disk)
 - *Then: CP WNG ALL* This system may be non-responsive for a few minutes while diagnostic information is obtained.
 - *Then: CP SNAPDUMP*
- Sometimes a simple message frees up the "hungid" without further ado. From a privclass A, B, or C userid, issue:
 - *CP WNG hungid Hello*
- If the ID was awaiting I/O to a terminal, simply connecting from a working terminal may free the ID. From a free terminal, issue:
 - *CP LOGON hungid **HERE***

Miscellany:

LOGON/LOGOFF FORCE PENDING

Hung users – getting doc, and saying goodbye.



S H A R E
Technology • Connections • Results

- For users logged on via TELNET terminals, issue:
 - *NETSTAT TELNET*
 - *Find the matching tn3270 connection, and issue:*
 - *NETSTAT DROP conn_num*
- "CPHX" is reported to cancel pending CP commands: ATTACH, LOCATE, LOCATEVM, and VARY ONLINE|OFFLINE (see HELP for more detail).
 - *From a privclass "A" userid, issue:*
 - *CP CPHX hungid*

Miscellany: LOGON/LOGOFF FORCE PENDING Hung users – getting doc, and saying goodbye.



- If TRACK (above) showed an active I/O which cannot be remedied (e.g. by making a tape drive "Ready"), the I/O may be able to be cancelled. From a privclass "A" userid, issue:
 - *CP HALT rdev*
- Due to queued I/Os or recalcitrant devices, HALT may need to be issued repeatedly until a message similar to the following is received:
 - *Halt was not initiated to tape nnnn because the device as not active*
- If nothing freed the hung user, open a Problem Management Report (PMR) with IBM, and provide the SNAPDUMP for analysis.

Miscellany: (and maybe, Best practices) CMS Pipelines Runtime Library (RTL)



- Not installed as part of IBM's CMS distribution (which comes with "Endicott Pipes").
 - Read: Another little task for you (but well worth the effort).
- Contains everything supplied with "Endicott Pipes", plus many added stages and other enhancements.
 - PIPERLD - CMS Pipelines Runtime Library
 - <http://vm.marist.edu/%7Epipeline/>
- "PICKPIPE EXEC" (on the IBM VM Download page) lets you swap back and forth between Pipes versions.
 - <http://www.vm.ibm.com/download/packages/>

Miscellany: Re-labeling DASD

- **Requires understanding of the “USER DIRECT” file.**
 - See: “z/VM CP Planning and Administration”
- **Requires updates to:**
 - DASD label (volser) -- use the CPFMTXA EXEC
 - “USER DIRECT” (DIRECTORY statement)
 - “SYSTEM CONFIG” on MAINT CF1, CF2.
 - C.A. “product CONFIG” file “DIRECT” records
 - e.g. “DIRECT vdev volser”
- **A DASD naming convention can make your job simpler.**
 - We were required to begin VM DASD labels with “VM” (change to meet your needs), hence:
 - **VMR52A, VMR52B etc. (In place of IBM’s 520RES)**
 - **VMSPnn – spool packs (In place of IBM’s 520SPL)**
 - **VMDUMP – if you are “DASD rich”, reminder to keep DUMP space contiguous and ensures there is enough.**
 - **VMPGnn – paging volumes (best to keep paging and spool alone on DASD due to seldom-ending channel programs).**
(In place of IBM’s 520PAG)
 - **VMPPnn – Program Product MDISKs and SFS spaces. Makes D.R. easier: first restore VMRnna and VMPPnn, while formatting VMSPnn,**
VMPGnn, and Allocating VMTDnn. Then IPL *your* system to restore everything all “non-system” DASD.
 - **VMTDnn – keeps TDISK separate from other MDISKs, but can result in heavy I/O.**
 - **VMUunn – “user” files**

Miscellany: Re-labeling DASD

- **Tip:** For each DASD volume, create a userid to permit a CP LINK to the volume as needed.

```
USER -volser- NOLOG 64M 64M  
MDISK A00 3390 0000 0001 volser R  
MDISK F00 3390 0000 3339 volser R
```

- **Be careful to NOT format cylinder 0 (zero);** it contains the “Allocation Bit Mask” describing the use of each cylinder (such as PARM, DRCT, WARM, CKPT, SPOL, PAGE, etc.), and ... the IPL TEXT!
 - You’ll know you did if you attempt to IPL and receive message:
HCPVMI232E IPL UNIT ERROR; IRB 00404017 00020A78 00200018 00800000
 - **Good Rule of Thumb:** Don’t allocate minidisks on cylinder zero, there are lots of other cylinders to use, why go looking for problems?

Miscellany: “HELPful” help files.

- Ancient HELP files have a way of accumulating. A tip from Roger Deschner at the University of Illinois Chicago; when copying groups of non-VMSES/E help files to the 19D disk.
(VMSES/E files get to MAINT’s 19D via VMFCOPY):
 - For non-VMSES/E help files going to the 19D disk, create a “product LISTFILE x1” to go on the 19D disk, too.
E.g. for VM:Backup help:
 - *“LISTFILE * * W (EXEC LABEL”*
 - *“COPY CMS EXEC A VMBACKUP LISTFILE x1 (OLDDATE”*
- When using the HELP command, only filemode ‘x2’ files are accessed. Using “filename LISTFILE x1” keeps them out of sight of users (less clutter), saving memory. All accessed files use up memory for pointers in the Active File Table (AFT).

Best practices: REXX (religion)

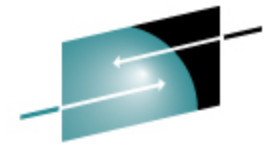
- REXX syntax can inspire near-religious wars!
- Be extravagant with comments when writing code. They will help you when you get old, gray-haired, and forget the code after a “long time”.
 - If you’re like me, they’ll help you next week.
 - There **is** such a thing as **too many comments**:

```
'ERASE SOME FILE A' /* Erase the file */
```
- Try to place comments where they explain something that’s **not obvious**, could have multiple meanings, is based upon the results of command which may return different results in different circumstances, etc.

Best practices: REXX (religion)



- I like Boolean variables (those TRUE (1) or FALSE (0) flags) to begin with '?' -- E.g. "If ?lost then ..."
 - Helps next reader understand it can only be true or false.
 - Mild problem with typical Pipe "ENDCHAR" and VAR stage.
- Avoid use of the 'INTERPRET' statement
 - (very powerful, very confusing).
- Avoid use of the "Logical Not" symbol " \neg ", instead use "<>", or "\?true"
 - Other operating systems which support REXX of some flavor (e.g. Regina REXX), but do not always understand "Logical Not".



Best practices: REXX (religion)

- In REXX code:
- I tend to line up 'SELECT/DO/END' clauses, helping find missing or extra END statements in XEDIT (what a **GREAT** editor!):

```
  If  ?AlmostDone  then
      Do until ?AllDone
          many statements
          many statements
      End /* until ?AllDone */
```

```
ALL / SELECT / | / DO / | / END /
```

```
      Do until ?AllDone
----- 2 line(s) not displayed
-----
      End /* until ?AllDone */
----- 7 line(s) not displayed
-----
```

Best practices: REXX (religion)



SHARE

Technology • Connections • Results

QDI, John Hartmann's XEDIT Macro for Coloring Program Syntax Elements

- <http://vm.marist.edu/~pipeline/#QDI>

```
B - JQPUBLIC - [43 x 80]
File Edit View Communication Actions Window Help

/* Prolog; See Epilog for additional information *****/
/* Exec Name      -          EHEC                      */
/* Unit Support   -          -                          */
/* Status        - Version n, Release n.n             */
*****/

address 'COMMAND'
parse source xos xct xfn xft xfm xcnd xenvir .
parse upper arg parms 1 operands '()' options ')' parmrest

Signal ON Syntax
Signal ON NoValue
/* Signal ON ERROR */

Do until wordpos(reply,'QUIT QUI QU Q QQ QUIT END STOP HALT HK')>0
  parse upper pull reply .
  Select
    When abbrev(reply,'SHUTDOWN',4) then ?shutdown=1
    When abbrev(reply,'TEST', 1) then ?test=1
    Otherwise
      Do
        say 'Invalid reply: "'reply'".'
        say 'RIP!'
      End
  End /* Select */
End

say 'Good job! You almost got this rexx statement right!'

Call Exit rc

*****/
/* Sub-Routines below this point */
*****/

Exit:
  parse arg exitrc .
  If verify(exitrc,'-0123456789')=0 then Exit exitrc
  else Exit 99999

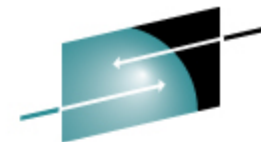
HE b 43/080
```

The “SYSTEM CONFIG” file; “Record Qualifiers” ...



- We'll go through “Record Qualifiers” quickly as it could be considered an “advanced” topic.
- But they are:
 - an important capability
 - is not obvious
 - and can be difficult to understand without an example...

The “SYSTEM CONFIG” file; “Record Qualifiers” ...



SHARE

Technology • Connections • Results

```
System_Identifier 2094 %%7777 yournode GATEWAY gw_name
```

- **%%** are wildcards, very important when running in an LPAR
- **yournode** is returned by the command: CP QUERY USERID
 - E.g. JQPUBLIC AT **BIGBLUE**
 - parse value diag(08,'QUERY USERID') with self . **ConfigSysID** . '15'x
 - Not to be confused by CMS command “IDENTIFY”, which comes from the ‘nodeid’ defined in the “SYSTEM NETID S” file.
- **If you have other CPUs AT YOUR SITE upon which you might recover.**

```
System_Identifier 2094 %%0666 RECOVERY GATEWAY gw_name
```

- So your system EXECs can act differently at a Disaster Recovery site (and users can tell, too).

```
System_Identifier_Default DISASTER GATEWAY gw_name
```

```
yournode: Timezone_Definition CST West 06.00.00
```

```
yournode: Timezone_Definition CDT West 05.00.00
```

```
RECOVERY: Timezone_Definition CST West 06.00.00
```

```
RECOVERY: Timezone_Definition CDT West 05.00.00
```

```
DISASTER: Timezone_Definition EST West 05.00.00 /* East-coast D.R. */
```

```
DISASTER: Timezone_Definition EDT West 04.00.00 /* East-coast D.R. */
```

The “SYSTEM CONFIG” file; “Record Qualifiers” ...



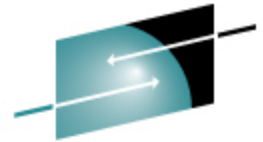
SHARE

Technology • Connections • Results

```
yournode: RECOVERY: , /* Devices "at home" and on other boxes "at home" */
  Rdevice 440-47F Type UNSUPPORTED DEVclass 3270_Display
DISASTER: , /* Same hardware, different address at D.R. provider */
  Rdevice 540-57F Type UNSUPPORTED DEVclass 3270_Display
...
yournode: RECOVERY: ,
  Operator_Consoles ,
    06A0 , Esysop */
    06A2 , Esysop2 */
    06A1 , PerfMon */
  SYSTEM_3270 , /* Integrated 3270 console on the HMC */
  SYSTEM_CONSOLE /* HMC *and* 2nd Level Autolog support */

DISASTER: ,
  Operator_Consoles ,
    0009 , /* Level 2 Testsys */
    00C0 00C1 00C2 00C3 00C4 00C5 00C6 00C7 , /* Disaster Recovery */
    00C8 00C9 00CA 00CB 00CC 00CD 00CE
    00D0 00D1 00D2 00D3 00D4 00D5 00D6 00D7 ,
  ...
  SYSTEM_3270 , /* Integrated 3270 console on the HMC */
  SYSTEM_CONSOLE /* HMC *and* 2nd Level Autolog support */
```

Miscellany: "SYSTEM NETID S2" file



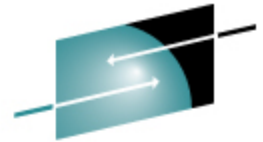
SHARE
Technology • Connections • Results

• The Effect of LPARs on the CPUID (note 2nd digit)

```
*CPUID NODEID NETID (Note: SECOND digit is the LPAR number)
* IBM z800 2066-0B1, in LPAR 1 (Same as "Basic Mode" when on SN x17777)
017777 yournode RSCS
117777 yournode RSCS
217777 yournode RSCS
317777 yournode RSCS
...
F17777 yournode RSCS
* IBM z800 2066-0B1, in LPAR 1
017777 yournode RSCS
117777 yournode RSCS
217777 yournode RSCS
317777 yournode RSCS
...
F17777 yournode RSCS
* IBM z800 2066-0B1, in LPAR 2
027777 yournode RSCS
127777 yournode RSCS
227777 yournode RSCS
327777 yournode RSCS
...
F27777 yournode RSCS
```

Arrggghhh!

Q: How Am I Supposed To Remember All This *NEW STUFF*?



SHARE
Technology • Connections • Results

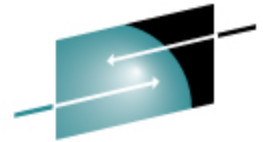
Don't try to remember everything –
just know where to look.

- Create “help me” files with often-used but hard-to-remember commands, and other details.
- Save the files on your own disk, or make them public.
- For example, since our VM system seldom has hung users, I forget the TRACK command to look for pending I/O on a given user.
- The file “TRACK HELPME” contains those commands.

```
B-M2Walker - [43x80]
File Edit View Communication Actions Window Help

TRACK:  HELPME  Z1  F 80  Trunc=80 Size=17 Line=0 Col=1 Alt=0
====>
.....1.....2.....3.....4.....5.....6.....7...
==== * * * Top of File * * *
==== OK, so I'm getting more forgetful...
====
==== T user CMS XEDIT
==== T user CMS ALL
==== T user CONSOLE
====
==== T user CMS STACK
==== T user DEV CLASS * IO PENDING
==== T user DEV CLASS * IO ACTIVE
==== T user CMS DISKS SFS
====
==== FORMAT VMDBK address HCPOM1 MACLIB
====
==== Nota bene: As of 23 Feb 2006 TRACK can be obtained from:
==== http://vm.marist.edu/track/code.html
====
==== * * * End of File * * *
```


A: Help yourself!



SHARE

Technology · Connections · Results

- **Just entering: *HELP ME TRACK* yields the clues I usually need.**
- **Build your own “HELPME” library as-needed.**
- **All that Linux for zSeries “stuff” is new to me, too.**
 - *I expect to write lot of new HELPME files.*

```
IBM-M2Walter - [43x80]
File Edit View Communication Actions Window Help

ME TRACK                               All Help Information           line 1 of 16
OK, so I'm getting more forgetful...

T user  CMS XEDIT
T user  CMS ALL
T user  CONSole

T user  CMS STACK
T user  DEV CLASS * IO PENDING
T user  DEV CLASS * IO ACTIVE
T user  CMS DISKS SFS

FORMAT VMDBK address HCPOM1 MACLIB

Nota bene: As of 23 Feb 2006 TRACK can be obtained from:
           http://vm.marist.edu/track/code.html

* * * End of File * * *

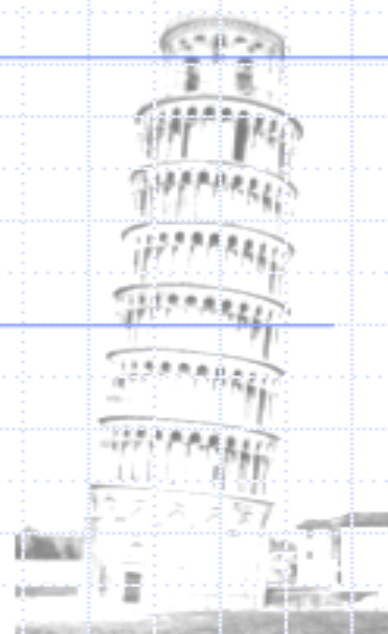
PF1=          2= Top      3= Quit    4= Return    5= Clocate   6= ?
PF7= Backward 8= Forward  9= PFkeys 10=         11=         12= Cursor

====>
Macro-read 1 File
```



SHARE

Technology • Connections • Results



INFRASTRUCTURE

SOONER OR LATER, IT MATTERS

- Questions?
 - Later...about session materials, or in general:
Mike Walter
Hewitt Associates LLC
847.771.9212
Mike.Walter -- hewitt.com
- ***Please***, fill in the **bubbles** on your Session Evaluation Form **COMPLETELY** in **DARK BLUE** or **BLACK** ink.
- **Help make this session better!** – ***write your comments on the back of your form.***

```

/* Prolog; See Epilog for additional information *****
* Exec Name      - MKSALIPL EXEC                               *
* Unit Support   - OSS/VM                                     *
* Status         - Version 1, Release 1.0                     *
*****/

address 'COMMAND'
parse source xos xct xfn xft xfm xcmd xenvir .
parse upper arg parms 1 operands '(' options ')' parmrest

Signal ON Syntax
Signal ON NoValue
/* Signal ON ERROR */

hi='1DE8'x          /* 3270 Hilite Char      */
lo='1D60'x          /* 3270 Default Char    */
parse var operands writevdev minivol volid errs
If writevdev='?' then Signal Explain
If errs<>' ' then
  Do
    say xfn'; Invalid operands found: "'errs'".'
    Call Exit 20
  End

If writevdev='' then
  writevdev='0F00' /* Where to write SALIPL          */
  extent=1         /* Which CP-fmt'd "PARM" extent    */
  If minivol='' then /* Mdisk label at "extent" (e.g. vdev=CF1, */
    minivol='NOVERIFY' /* label="MNTCF1") should be checked first. */
    /* the "CPLOAD MODULE" should be loaded */
  module='CPLOAD' /* Default=CPLOAD, CP nucleus module to load*/
  offset=0        /* Default=0, must be 0 if "extent" is given*/
  origin=1000     /* Default=1000, address at which SALIPL */
    /* should load the "module". */
  If volid='' then /* Mdisk or CP label to be checked before */
    volid='NOVERIFY' /* write. */

iplparms='' /* Typically entered at IPL time */
comments='?'

```

```

/* Exactly 4 lines of comments, truncated at 80 bytes          */
c.1='Some possible IPL PARAMETERS section (above) entries:'
c.2='CONS=ccuu          PROMPT          <--- Used most often'
c.3='NOEXITS          FN=system          FT=config'
c.4='PDNUM=ParmDiskNum  PDVOL=ParmDiskVolser  PDOFF=ParmDiskcyloffset'
c.0=4
cmd='SALIPL' writevdev '(EXTENT' extent 'MINIVOL' minivol ,
    'MODULE' module 'ORIGIN' origin 'VOLID' valid ,
    'COMMENTS' comments
Do forever
  say hi
  say 'The following command is about to be executed' lo
  say cmd
  say 'stacked Comments:'
'PIPE STEM c. | CONS'
  say
  If minivol='NOVERIFY'
    then say 'Minidisk extent' extent '(e.g. MAINTs CF1) WILL' ,
            'NOT have its volser verified.'
    else say 'Minidisk extent' extent '(e.g. MAINTs CF1) WILL' ,
            'be verified as:' minivol
  If valid='NOVERIFY'
    then say 'The virtual device at "'writevdev'" WILL' ,
            'NOT be verified.'
    else say 'The virtual device at "'writevdev'" WILL' ,
            'be verified as:' valid

  say
  say hi
  say 'Continue? Enter Yes or No' lo
  parse upper pull ans .
  If abbrev('YES',ans,1) then Leave /* Forever */
  If ans='NO' | ans='N' then
    Do
      say 'Aborting before writing SALIPL per reply:' ans
      Call Exit 32
    End
  say
  say '+++ Invalid reply:' ans
End /* forever */
'PIPE STEM c. | STACK'
cmd
Call Exit rc

```

Explain:

```
say xfn 'is used to write SALIPL to the a given address (usually a'
say 'full-pack extent minidisk.'
say
say '          +-0F00+----NOVERIFY++  +-NOVERIFY++          '
say '>-MKSALIPL---vdev+----minivol---+----volid-----><'
say
say 'Example: MKSALIPL 123 MNTCF1 510RES'
say
Call Exit 4
```

```
/*-----*/
/*          Sub-Routines below this point          */
/*-----*/
```

Exit:

```
  parse arg exitrc .
  If verify(exitrc,'-0123456789')=0 then Exit exitrc
  else Exit 999999
```

Syntax:

```
  say '+++ Syntax error routine entered in:' xfn xft xfm
  say '+++ from line:' sigl', which reads:'
  say '+++'sourceline(sigl)
Call Exit 20
```

NoValue:

```
  say '+++ NoValue error routine entered in:' xfn xft xfm
  say '+++ from line:' sigl', which reads:'
  say '+++'sourceline(sigl)
Call Exit 24
```

```
/* Epilog *****/
* Function          - See 'EXPLAIN' subrtn above.          *
* Component of     - VM Sysprogs toolbox                  *
* Command format   - See 'Explain; subrtn above.          *
* Called by        - VM sysprogs                          *
* Program Lang.    - CMS REXX                              *
* Date Written     - 20001117                              *
* Author           - Michael R. Walter                     *
* Acknowledgments - Based upon code the to the VMESA-L listserv *
*                 Thu, 31 Aug 2000 13:35:14 -0500        *
*                 "Frank M. Ramaekers Jr." <FrankRam@Earthlink.Net>*
* Changed | By | Description of Change                    *
* -----+-----+-----*
* 20060101 mrw - Add 'volid' operands, improve 'Explain'. *
* -----+-----+-----*
```


VM520 directory entry:

(A sample 1st level userid used to install and run 2nd level system)

USER VM520 xxxxxxxx 64M 2047M G 60
ACCOUNT OVERHEAD 93S0
MACH XA
AUTOLOG VM520IPL

<===== Permit VM520IPL to XAUTOLOG this userid (VM520)

IPL CMS PARM AUTO CR
OPTION MAXCONN 2048 TODENABLE
OPTION SVM QUICKDSP
OPTION SVC76VM
IUCV ANY

CONSOLE 6A0 3215 C VM520IPL
SPOOL 00C 2540 READER *
SPOOL 00D 2540 PUNCH B
SPOOL 00E 3211 P

<===== Permit VM520IPL to enter commands for this userid
Requires "SYSTEM_CONSOLE" included in "SYSTEM CONFIG" statement
"Operator_Consoles"

*
LINK MAINT 190 190 RR
LINK MAINT 19D 19D RR
LINK MAINT 19E 19E RR
LINK -VMT52A- F00 280 MR
LINK -VMT52B- F00 288 MR
LINK -VMR52A- F00 28C RR
LINK -VMR52B- F00 28D RR

* R/O link to PP volumes

LINK -VMPP01- F00 301 RR
...
LINK -VMPP06- F00 306 RR

* TEST VM system volumes

LINK -VMPPX1- F00 307 MR
LINK -VMPPX2- F00 308 MR
LINK -VMUDX1- F00 309 MR
LINK -VMUDX2- F00 310 MR
LINK -VMUDX3- F00 311 MR
*LINK -VMSPX1- F00 310 MR
*LINK -VMTDX1- F00 311 MR
*LINK -VMPGX1- F00 312 MR

VM520 directory entry, continued:

(A sample 1st level userid used to install and run 2nd level system)

*** TEST VM User volumes**

***LINK -VMUT01- F00 321 MR**

...

***LINK -VMUT05- F00 32A MR**

*** R/O link to 1st Level User volumes**

LINK -VMU200- F00 1200 RR

LINK -VMU201- F00 1201 RR

...

*** Something to DIAL to that won't try to come up as a VM:Operator console**

SPECIAL 610 3270

SPECIAL 611 3270

...

SPECIAL 62F 3270

***SPECIAL 6A0 3270 defined as CONSOLE above**

SPECIAL 6A1 3270

...

SPECIAL 6CF 3270

MDISK 0191 3390-3 1812 2 VMPP0 4 MR