



Refer to Figure 20

Our Marketing Division consists of System Engineers (SE) and Marketing Representatives. The Marketing Representative is responsible for sales and the SE assists customers in installation planning and procedures.

PSRs and CEs work in the Field Engineering Division. PSRs provide software support and the CE is responsible for hardware performance.

In addition we have added the IBM Support Centers. This is an expedited software support service comprised of over 3,000 people across the country with a commitment to provide quality service to each of IBM's many customers. The development of remote support has proved to be an outstanding achievement for IBM. I sincerely hope the information I have provided will assist you in making the Support Center work for you!

SHARE SESSION REPORT

SHARE NO.	SESSION NO.	SESSION TITLE	ATTENDANCE
61	B549	Ok, VM is Installed, Now What?	150
CMS		Larry Grazioplene	BAM
PROJECT		SESSION CHAIRMAN	INST. CODE
Bank of America, 1455 Market St., San Francisco, CA 94103 (415) 622-1881			
SESSION CHAIRMAN'S COMPANY, ADDRESS, AND PHONE NUMBER			

OK, VM IS INSTALLED,

NOW WHAT?

Tips and Techniques for Running a VM System, and Changing CP's Page, Spool, T-Disk, and Free Storage Areas

August 23, 1983

SHARE Session B549

Ben Pettersson
Boston M&D Office
IBM Corporation
610 Lincoln Street
Waltham, MA 02254

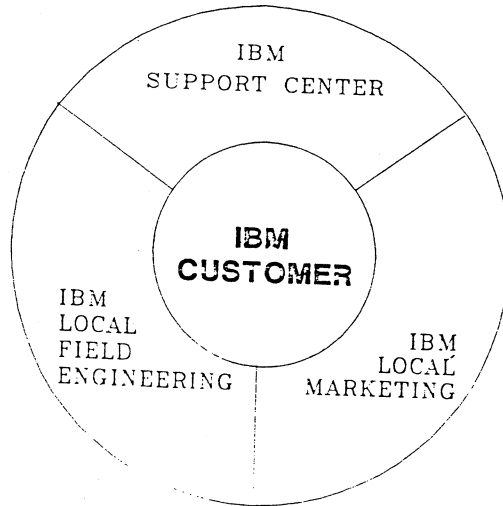
VM New Users Project

Permission is granted to SHARE, Inc. to publish this presentation paper in the SHARE proceeding in SHARE 60. IBM retains the right to distribute copies of this presentation to whomever it chooses.

4/C/rcg/3

237

Figure 20



INTRODUCTION

This presentation is primarily aimed at VM/SP customers who have installed the VM/SP System Installation Productivity Option-Extended (SIPOE), and who are running VM "as is" with little or no tailoring. I will be covering the following topics today:

MISCELLANEOUS TOPICS

- Tailoring the AUTOLOG1 Machine's PROFILE
- Tailoring VMUTIL's WAKEUP PARMS File
- Virtual Control Unit Interference
- Making Batch Spooling More Efficient
- Putting a Test DOS or VS1 SYSRES on a Minidisk
- Why Multi-Write Update is a Problem in CMS

DMKSYS CONSIDERATIONS

- New Missing Interrupt Handling Function - SYSMIH
- Adding CP-Owned Volumes to SYSOWN
- If You Want To Change VM's SYSRES Address of "123"
- Automatic T-Disk Clearing
- Paging/Spooling on Two Different Types of Disks
- Free Storage Considerations - What it is and why CP EXTENDS may make you change it.

PAGING, SPOOLING, AND T-DISK CONSIDERATIONS

- How CP Handles CP-Owned Volumes
- T-Disk Areas
- Page/Spool Areas
- Separating Paging from Spooling
- Adding a New TEMP Disk
- Looking at CP SYSRES's Allocation Information

- Reducing the S12E of TEMP and T-DISK Areas

TAILORING AUTOLOG1'S PROFILE EXEC

In a SIPOE system, when VM is IPLed, the AUTOLOG1 machine is automatically logged on, and the following functions occur:

- The virtual machine is put to sleep for ten seconds to allow the VM Operator to force off the AUTOLOG1 machine if something is seriously wrong with its PROFILE EXEC.
- Autolog various SIPOE virtual machines such as CMSBATCH, ROUTER, and RSCS.
- Enable all local 3270s (and remote ASCII teletype and start/stop terminals).

You might want to consider the following changes if you have not made them already:

- Reduce the sleep interval to 2 to 3 seconds, or eliminate sleep.
- Put an asterisk (*) in front of the AUTOLOG statements for machines that you are not currently using (changes the statement to a comment).
- If you want your guest DOS/VSE or OS batch machine to own most of the terminals, replace the CP ENABLE ALL statement with a

```
CP ENABLE CUU CUU CUU ...
```

to put the VM LOGO on only a few specific 3270s.

If you don't know what the address of a particular terminal is, issue CP ENABLE ALL, then press ENTER at the terminal to clear the VM LOGO and, without logging on, issue

```
MSG * HI
```

to send a message to yourself. CP will display on your screen:

```
MSG FROM LOGONXXX HI
```

where XXX is your terminal address.

To find the RESID (resource ID) address of a remote 3270 terminal, issue CP NETWORK ENABLE ALL and send a message to yourself. CP will display

```
MSG FROM LOGNxxxx HI
```

where XXXX is your terminal RESID address.

Other things you might want to add to AUTOLOG1'S PROFILE EXEC are:

If the VM Real Time Monitor (SMART) is installed:

add near the beginning of the PROFILE:

```
CP AUTOLOG SMART password CP SET RUN ON
```

Insert near the end:

```
EXEC SMINIT SMART
```

to set up SMART's threshold values.

If VM Performance/Monitor Analysis Program (VMAP) is installed:

and you want to run Monitor twenty-four hours a day, add the following statements:

```
CP MONITOR ENABLE PERFORM USER DASTAP ...
```

```
CP MONITOR START SPOOL
```

Then, regardless of what time of day VM is IPLed, Monitor will be enabled.

If VM is at VM PUT 8207 or higher, and you want to turn off VM's new Missing Interrupt Handler function which was made available in PUT 8207, add:

```
CP SET MITIME OFF
```

VMUTIL'S WAKEUP PARM FILE

If your VM system operator would like to eliminate the

```
"THE TIME IS NOW ..."
```

message from appearing on OPERATOR'S console every two hours, edit WAKEUP PARMS and globally erase the

```
"MSG01"
```

operand from the file by doing the following:

```
C /MSG01// **
```

If you want to run MONITOR twenty-four hours a day, during the week, but wish to get VMAP output by shift for example,

```
1st - Midnight to 9 a.m.
```

```
2nd - 9 a.m. to 5 p.m.
```

```
3rd - 5 p.m to Midnight
```

add the following commands to the WAKEUP file:

```
M-F 00:02:00 02/24/83 CP MONITOR ENABLE PERFORM USER DASTAP
```

```
M-F 00:02:00 02/24/83 CP MONITOR START SPOOL
```

```
M-F 09:00:00 02/24/83 CP MONITOR CLOSE
```

```
M-F 17:00:00 02/24/83 CP MONITOR CLOSE
```

```
M-F 23:58:00 02/24/83 CP MONITOR STOP
```

Note: CLOSE tells CP to send all accumulated Monitor data to VMAP, but continue collecting Monitor data. STOP stops collection of Monitor data.

VIRTUAL CONTROL UNIT INTERFERENCE

Does the console of your batch virtual machine mysteriously hang and then free up fairly often?

One of the common causes of this condition is VIRTUAL CONTROL UNIT INTERFERENCE, which is caused by having incompatible devices logically attached to the same virtual control unit, that is, some devices require SHARED Unit Control Words (UCWs) while other devices require UNSHARED UCWs.

I have seen this situation occur many times with many customers when the guest batch machine has its console in full screen (DOC) mode on a 3270 at virtual address 01F, and one or more of the following virtual devices within the address range:

010 TO 01F

010 - 3705
01C - Diskette Reader
01D - CP Spooled Punch
01E - CP Spooled Printer

My usual recommendation is to change the batch machine's virtual console from 01F to something else.

Another not-so-obvious example of virtual control unit interference is:

Disks on 358 (uses SHARED UCWs)
Tapes on 350 (uses UNSHARED UCWs)

The disks will hang until tape I/O operations are finished.

MAKING BATCH MACHINE SPOOLING MORE EFFICIENT

If you dedicate your system printer to a guest batch virtual machine, make sure your spool blocks are as large as possible, otherwise, when your virtual machine's time slice ends, your printer channel program may terminate before your virtual machine is dispatched again, slowing down the effective print speed.

For example, in DOS/VSE POWER, POWER spool blocking defaults to 800-1000 bytes, depending on the type of disk being used, whereas the maximum allowable blocking is 2008 bytes. By specifying a value as close to 2008 as possible, the printer's channel program will continue for a longer period of time after time-slice end, driving the printer at closer to its rated speed.

To Double Spool or Not to Double Spool

If two or more batch machines as well as CMS users need to share the same printer, rather than attaching and detaching the printer for specific virtual machines, it might be better to have CP own the printer. You might think that double spooling would not be very efficient, but in fact it is a very good performer. The alternative of not using the batch machine's spooling mechanism but rather assigning separate virtual readers, punches, and printers to each partition or region would cause 7 TO 8 TIMES the number of Start I/Os (SIOs) that CP would have to handle since every 80-character reader and punch record and every 120- to 133-character print line would cause an SIO. Such a large number of SIOs would substantially increase the CP overhead, since SIOs are one of the most significant causes of overhead in a VM system.

PUTTING A TEST DOS OR VS1 SYSRES ON A MINIDISK

Minidisk Restrictions

A DOS, OS, or CMS minidisk cannot start in real cylinder zero of a CP-owned pack (that is, a pack with CP paging, spooling, and T-disk or VM directory areas on it), since CP's Volume Table of Contents (VTOC) is incompatible with those of DOS/VSE and OS. A major difference is CP's Allocation Bit Map which is located on cylinder zero, track 0, record 4. (How to eliminate this incompatibility is beyond the scope of this session.)

VS1 Minidisk Considerations

VS1 minidisks must start in real cylinder zero of each disk if a multi-track VTOC is used because the VS1 Locate Catalog Management function uses a "Search ID equal or high" channel command word (CCW) to find the end of the VTOC. Therefore, you cannot put CP-owned functions such as paging, spooling, or T-disk space on this pack.

If the VS1 minidisk started on other than real cylinder zero, the search ID equal or high would always occur after only the first track of the VTOC is searched. If you have only a one-track VTOC, the VS1 minidisk can be placed anywhere.

The Device Support Facility (DSF) is required to initialize a minidisk on 3375 and 3380 disks. For all other disks, you may optionally use the IBCDASDI program. In either case, you should use the "Cylinder" option to define the number of cylinders to be allocated for the minidisk.

DOS Minidisk Considerations

DOS, DOS/VS, or DOS/VSE minidisks can be located anywhere (except on cylinder zero of CP-owned volumes) and can be initialized by DSF with the number of cylinders or blocks option specified. Optionally, the DOS Initialize Disk Program (INITDISK) could be used to initialize minidisks on all DASD types except 3370, 3375, and 3380s. If INITDISK is used, you must use the QUICK DASDI function:

```
// UID IS
```

to put the correct number of cylinders in bytes 19 and 20 of the VTOC.

WARNING: Do NOT use the IQ initialization function of INITDISK, or IBCDASDI to initialize DOS minidisks, since both place an incorrect value for "number of cylinders" in bytes 19 and 20 of the VTOC, making the minidisk unusable for DOS SYSRES System or Private Library functions. If you try to use a DOS function to COPY or RE-

STORE onto an incorrectly initialized minidisk, you will get a vague error message stating that the function did not work.

WHY MULTI-WRITE UPDATE IS A PROBLEM IN CMS

Since CMS does not provide any form of multi-update support such as RESERVE/RELEASE or ENQ/DEQ, if two or more CMS users have multi-write links and accesses to the same minidisk, bad things are likely to happen to that minidisk. You take the risk of losing one or more files and perhaps the whole minidisk if double updating occurs.

When two users LINK and ACCESS the same minidisk, CMS gives each user a STATIC view of the contents of the disk, including a bit map that shows which blocks of the minidisk are used or available. A file erased by one individual is still seen, and can be accessed by the other user. A file added by one user is not seen by the other. If one user updates an existing file, the other user will see the OLD copy of the file. The first person to write his file back on disk MAY have that file overlaid when the second user writes a file on the disk. Whether or not both users write on exactly the same area depends on if a user ERASED an old file prior to creating a new file. Depending on where the erased file's blocks were located, the newly created file may re-use some or all of the erased file's blocks. Even if neither user erased a file before creating a new one, it is very unlikely that both people wrote exactly the same size file, so CMS's pointers to available areas on the minidisk are now permanently messed up. If the first user tries to write out another file, CMS will detect that the pointers are bad and will ABEND. The user can IPL CMS again, but he can't write on the file again.

If you do not have a current tape backup of the files on the minidisk, you can define a T-Disk and use FLIST/FILELIST to copy over all UNCHANGED files. (Any attempt to copy a changed file that has messed-up pointers will cause CMS to ABEND again.) When you are done copying, you must reformat your minidisk with the CMS FORMAT command to clean up the pointers. Then copy the files back from the T-Disk to your reformatted minidisk.

Many customers encounter this double update problem when they try to update the 191 minidisk of a disconnected virtual machine that is also periodically updating its own disk; for example, VMUTIL's WAKEUP PARMS file. Therefore, if you want to update a file such as WAKEUP PARMS, first do a QUERY NAMES. If VMUTIL is running, force it off, update the file, and then AUTOLOG VMUTIL back on again.

To avoid this double update problem, edit your VM directory and change all non-essential multi-write (MW) modes to Multi-Read (MR) and eliminate the associated "MULTI" passwords.

DMKSYS CONSIDERATIONS

New Missing Interrupt Handling Function - SYSMIH

With PUT 8207 or later, a new Missing Interrupt Handling function has been added to DMKSYS. If you apply such a PUT tape and reassemble DMKSYS without adding the new SYSMIH macro statement, you will get a mildly reproachful MNOTE saying that the default values for SYSMIH will be assumed. The TEXT deck created is OK, though.

With the new Missing Interrupt Handling support in VM, if CP detects a missing interrupt condition for some I/O device attached to the guest virtual machine, CP passes an INTERFACE CONTROL CHECK condition back to the virtual machine. Since DOS and VS1 do not handle interface control checks very well, you might want to turn the new function off by adding to AUTOLOG1's PROFILE the following statement:

CP SET MITIME OFF

Adding New CP-Owned Volumes to SYSOWN

If you want to spread paging and/or spooling space over more disks, you must use CP Format/Allocate to format and allocate the new packs. This procedure is described in detail later in this session. The new VOLIDs (assumed to be VMPK02 and VMPK03 on 3350s) should be added to the END of the SYSOWN statement. Reassemble DMKSYS and generate a new CP nucleus. When VM is shutdown and reIPLed, the VM operator can then specify a WARM start, so no existing spool files will be lost.

Old SYSOWN Statement: SYSOWN VMSRES,VMPK01

Updated SYSOWN Statement: SYSOWN VMSRES,VMPK01,VMPK02,VMPK03

Changing VMSRES's Virtual Address from '123'

In a SIPOE system, the CP System Residence Volume (VMSRES) has a virtual address of 123. There is no need for you to change VM's virtual SYSRES address, but if for some reason you want to, you must also change at least two other places that refer to the virtual address '123.'

In DMKSYS, SYSRES=123

In the VM Directory (in two places)

DIRECTORY 123 3350 VMSRES

USER MAINT password
MDISK 123 3350 000 555 VMSRES MW ...

Since IPF has the virtual address "123" hardcoded in many maintenance EXECs, if you do not find and change all occurrences of "123" to your real SYSRES address, you will not be able to use IPF for those maintenance functions. Also, unless you move the modified IPF EXECs to a different disk, the next PUT tape for IPF that is installed on your system will wipe out your changes.

After you reassemble DMKSYS, but before you can generate a new CP nucleus, you must redefine "123" to your new address (cuu):

CP DEFINE 123 cuu

otherwise, your CP nucleus build and load process, which is looking for your new SYSRES address, will fail with an obscure "I/O error" because it can't find the new virtual address.

Automatic T-Disk Clearing - A Security Feature

SIPOE's DMKSYS file has a parameter in the SYSRES macro called SYSCLR

SYSRES SYSCLR=YES,

This SYSCLR operand tells CP during VM Initialization to clear, in small groups of cylinders, ALL T-Disk space on ALL CP-owned volumes. A CMS user must still invoke the CMS FORMAT program before using the T-disk area. When the user detaches the T-Disk space or logs off, the area is cleared by CP again. If you are not that concerned about privacy and security of temporary data, and you want to eliminate some CP overhead, you might wish to change SYSCLR to "NO."

SYSRES SYSCLR=NO,

NOTE: Do not just eliminate the SYSCLR operand, because the default is YES, not NO.

FREE STORAGE CONSIDERATIONS

Paging/Spooling on Two Different Types of Disks

If you are paging and spooling on 3350S on Channel 1, but would like to also use a different type of DASD such as 3370s on Channel 2, there are several ways to accomplish this using the SYSORD macro.

The default SYSORD that comes with the system has no operands specified, but, for 3350s, it defaults to

```
SYSORD SYSTEMP=(3350)
```

If the CP Format/Allocate program was used to define PAGE space on 3370s and to define TEMP space on 3350s, then SYSORD should be:

```
SYSORD SYSMH=(3370),SYSTEMP=(3350)
```

ALL PAGING will be done on 3370s, on Channel 2, and

ALL SPOOLING will be done on the 3350s on Channel 1.

If both paging and spooling have high activity, this is a reasonable choice.

If, however, both 3350s and 3370s were allocated as TEMP and SYSORD is

```
SYSORD SYSTEMP=(3350,3370)
```

the results are not at all what you might expect.

CP will do ALL paging and spooling on the DASD type specified first (the 3350s). Only when all TEMP space on 3350s are FULL will CP start to use the second type of disks, the 3370s. If what you really want to do is spread paging and spooling activity -- round-robin fashion -- EQUALLY across both DASD types and both channels, you must specify TWO SETS OF PARENTHESSES as shown below:

```
SYSORD SYSTEMP=((3350, 3370))
```

If you wanted to round-robin on 3350s and 3370s, and use a third type of disk such as 3375s only when the others are full, specify

```
SYSORD SYSTEMP=((3350, 3370),3375)
```

We will look at paging/spooling considerations in more detail a bit later.

CP builds control blocks and tables for each virtual machine running under VM in an area called FREE STORAGE. The VM Planning and System Generation Guide estimates that a non-SIPOE CMS user requires approximately 2.3K of free storage. Since on a SIPOE system, the average size of a CMS virtual machine is bigger (512K VS 320K, has more disks accessed (6 or 7 VS 3), and the saved discontinuous shared segments (DCSSs) are positioned higher in virtual storage (starting at 2 meg VS 1 meg), the actual Free Storage use per virtual machine is 1.2K to 1.4K bigger.

CP uses a large amount of free storage for another unexpected purpose - TERMINAL I/O AREAS - for ALL active terminals, local and remote, that are connected to the VM computer regardless of whether the terminals are being used by CMS or by the teleprocessing (TP) programs in guest batch virtual machines. The size of each terminal I/O area is dependent on the type and model of terminal used:

3278/3279 TERMINAL TYPE	SIZE I/O AREA (APPROX.)
2	2K
3	3K
4	4K
5	5K

In addition, if you use GDDM graphics on 3279 terminals, a screen-load of data can exceed 40K bytes.

If CP momentarily runs out of Free Storage, CP takes a page away from the DYNAMIC PAGING AREA (also called Pageable Pages Area) where virtual machines and the pageable part of the CP nucleus contend for available pages. This process is called EXTENDING. When CP finds that it is done with the control information or terminal I/O area on that extended page, CP gives the page back to the Dynamic Paging Area, a process called DISEXTENDING (or UNEXTENDING). While CP is extending or disextending, the VM system grinds to a halt, so performance degradation can become noticeable if the extend/disextend rate becomes somewhat high, even if the total NUMBER of extended pages remains relatively small.

To avoid this "YOYO" effect of extend, disextend, extend and disextend, there are several things you can do:

- Specify a larger amount of free storage for your VM system.
- Request your CMS users to LOGOFF rather than DISCONNECT when their CMS machines will be idle for long periods of time. Each disconnected machine continues to tie up 2-3K of free storage even when nothing is running in those machines.

- Move non-critical batch production off-shift, if possible.
- Reduce the number of partitions or regions in your batch virtual machines.
- Reduce the size of your CP nucleus by using the SMALL CP option, or by hand-tailoring the CP nucleus loadlist to eliminate unused CP routines.
- Apply CP and CMS PTFs to fix any known problems that cause CP to waste or not release free storage.

Since a larger free storage area will cause a corresponding decrease in the Dynamic Paging Area, if your VM system already is storage-constrained and paging to some degree, making free storage larger will cause paging to go up even more. If your computer is not already at its maximum storage size, now may be a good time to add more storage. If you are paging heavily, but you are already at maximum storage, you probably have to consider the next larger CPU that permits more main storage.

Assuming for the moment that you have sufficient real main storage on your computer, if you have not specified in DMKSYS how much free storage you want, CP assumes the following default free storage sizes based on the real storage size you specified on the SYSCOR statement.

REAL STORAGE SIZE IN MEGS	NUMBER OF FREE STORAGE PAGES
1	15
2	31
3	47
4	63
8	127
16	255

NOTE: For AP/MP systems, add 25% plus ten pages for each relevant storage size.

The following is an example of the storage layout of a 4 meg CPU:

CP RESIDENT NUCLEUS	DYNAMIC PAGING AREA	TRACE TABLE	FREE STORAGE
300-400K	3,380-3,480K	64K	252K
75-100 PAGES	845-870 PAGES	16 PAGES	63 PAGES

4 MEG 4341 VM SYSTEM LAYOUT

Determining How Much Free Storage You Need

There is no easy way to determine how much free storage CP is actually using, unless, of course, CP is starting to extend. A rather loose rule-of-thumb that I use is 5K per CMS virtual machine plus 10-15K per guest batch virtual machine. Most customers I know use the default value for free storage unless there is no CMS usage and just a few guest batch machines.

There are a number of ways that you can tell how serious your extend problem is. If you have SMART (the VM Real Time Monitor) installed and you have established threshold values via SMINIT EXEC for both TOTAL extended pages and the extend/disextend RATE, SMART will issue the following messages:

SYSTEM EXTENDING nnnn PAGES MINUTE

which indicates that the specified extend/disextend "YOYO" rate per minute (normally 60-120) has been reached. This figure is not particularly helpful in determining how much more free storage is needed.

CP FREE STORAGE EXTENDED nnn PAGES

which indicates that NNN new extended pages have been taken from the Dynamic Paging Area since VM was IPLed, or since the last time this warning message was issued. If you add up the total of nnn pages for all occurrences of this message, you will end up with a MINIMUM value by which free storage should be raised.

If you have the VM Performance/Monitor Analysis Program (VMAP) installed, several fields will indicate the seriousness of your extend problem.

On VMAP's System Summary page, the two fields to look at are:

	AVE.	MAX.
	----	----
FREXTND	16.83	88.00
FREUNXTND	16.47	88.00

which show the average and maximum rate of extends and disextends per interval (normally 15 minutes). The "max" values show the greatest activity rate during any 15 minute period that Monitor is collecting data.

To see how extend activity varies during the day, look at the following sample listing from VMAP:

SUMMARY OF SYSTEM MAIN STORAGE AND PAGING LOG

	ALLOCATION	FREE STORAGE	
	PAGE	XTND	UNXTND
TIME	ABLE	/INT	/INT
10.00	850	0	0
10.15	845	15	15
10.30	840	11	10
10.45	839	4	3
11.00	831	20	17

The right-hand two columns again show the actual rate of extends and disextends during each monitored 15 minute period, but I prefer to look at the PAGEABLE column to determine the total number of extended pages taken.

You must adjust upward the pageable values shown in this table by the number of pages newly-reserved by the CP LOCK command and by net new shared pages used. Then the result of subtracting the lowest value from the highest value for Pageable Pages (i.e., the Dynamic Paging Area) will give you the minimum amount by which you should increase free storage.

Finding Extend/Disextend Information Manually

If you don't have VMAP or SMART, you can use the CP command Display Real CP Storage (DCP) to manually look in real CP storage at the two 4-byte fields that contain respectively, in hex, the total number of extends and disextends since VM was last IPLed.

When you have located the minidisk (normally OPERATNS' 193 or MAINT's 194) that contains your current CP nucleus map (normally called CPNUC MAP), edit or browse the file looking for the label DMKFREN:

/DMKFREN

When you get a match, write down the hex address associated with that label. Then, using any virtual machine such as MAINT or OPERATOR that has CP privilege class C or E, issue the command

CP DCP HEXLOC.8

which will display the following 8 bytes of information:

NO. OF	NO. OF
EXTENDS	DISEXTENDS
4 BYTES	4 BYTES

If you subtract the number of disextends from the number of extends, convert the result to decimal, and add a few more, for good luck, to the total, you should bump up the FREE operand in SYSCOR by that amount.

For example, if you had a 4 meg 4341 with the default free storage of 63 pages, and the difference between the number of extends and disextends was 15 decimal, you should probably add twenty pages and specify:

SYSCOR RMSIZE=4M,FREE=83

PAGING, SPOOLING, AND T-DISK CONSIDERATIONS

How CP Handles CP-Owned Volumes

"CP-Owned" volumes, by definition, are disks whose VOLIDs are defined in the DMKSYS SYSOWN macro that contain any or all of the following:

- Paging
- Spooling
- T-Disk
- VM Directory
- Dump Area

Other disks not specified in the SYSOWN list may also contain any of the above "CP-owned" areas, but are ignored by CP. This is one of the reasons you can run VM under VM.

CP establishes, during VM Initialization, where all these CP-owned areas are. Any changes you make while VM is running are UNKNOWN to VM until VM is shutdown and reIPLed again. This means you can add, format and allocate any NEW area while VM is running without worrying about its impact on VM.

You can also invoke the ALLOCATION portion of the CP Format- /Allocate program while VM is running to cut down on CP's existing paging, spooling, and T-disk areas. Since CP may happen to use those cut-down areas until VM is shutdown, you should NOT attempt to use those areas for anything else until VM is reIPLed. For example, if CP tries to write a 4K page or a printer spool block into an area that has just been reformatted into a 1K or 2K block CMS minidisk, CP will ABEND.

Of course, if you want to play it safe, you can shutdown the VM system and IPL the standalone version of the CP Format/Allocate program from a tape drive. To ensure that the CP Format program is at the latest level, don't use the one at the beginning of the PID starter tape - build your own IPLable tape copy by using the following procedure:

Attach a tape to your virtual machine.

```
ATTACH CUU * 181
```

```
FILEDEF INMOVE DISK IPL FMT S2  
FILEDEF OUTMOVE TAPI (DEN 1600 (OR 6250)  
MOVE
```

This procedure moves the IPL FMT program from the CMS 190 system disk to the tape.

NOTE: If you also want to put a copy of DDR and Directory (DIR) onto the same tape, just issue:

```
FILEDEF INMOVE DISK IPL DDR S2  
MOVE  
FILEDEF INMOVE DISK IPL DIR S2  
MOVE
```

NOTE: Since the OUTPUT statement did not change, it does not have to be keyed in again.

How CP Handles T-Disk Areas

If you have a lot of T-disk usage, what would you think is better?

One big T-Disk Area

Two medium-sized T-disk areas on two disks or

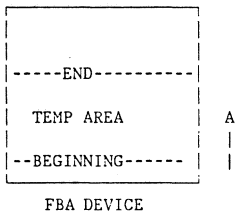
Three smaller T-disk areas on three disks.

The answer is 3 T-disk areas, but not for the reason you may think. Unlike paging or spooling activity, CP does not attempt to spread T-disk activity evenly across all available T-disk packs. CP will use up the first T-disk area before starting to use the next T-disk area. If you make the T-disk areas too big, you might have an imbalance of T-disk activity on that pack. If your T-disk areas are heavily utilized, either place those areas near the paging/spooling areas, or even better, move the T-disk areas to other packs.

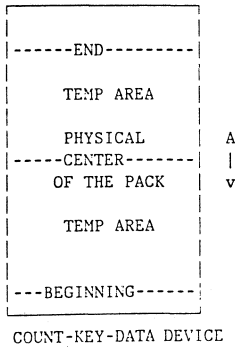
Page/Spool Areas

On VM SIPOE systems, paging and spooling areas are interspersed in TEMP areas on the CP-owned packs. How CP uses the TEMP areas depends on whether the disk type is Fixed Block Architecture (FBA) or Count-Key-Data (CKD). For CKD devices, it also depends on exactly where the TEMP area is physically located on the disk.

For FBA devices such as the 3370, CP starts using space from the BEGINNING of the TEMP area. If you want to make the TEMP area bigger or smaller, make the changes to the HIGH END of the TEMP space.

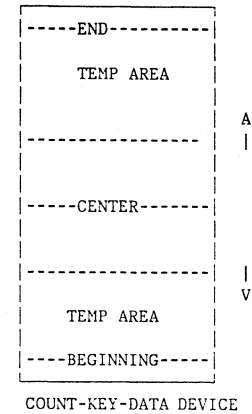


For Count-Key-Data disks such as the 3350, 3375, or 3380, CP starts to use TEMP space from as close to the physical center of the pack as it can, and then progressively alternates between the next lower and next higher cylinders to minimize arm movement. The ideal TEMP area location, therefore, is exactly in the center of the pack.



A poorer choice for location is to have the TEMP area completely WITHIN the top half or bottom half of the pack since average access arm movement will increase CP can select new TEMP cylinders in only one direction.

The worst choice of all would be to have two TEMP areas at opposite extremes of the pack, since CP TEMP activity on both areas will cause extremely long average arm movements.



Assuming that your existing TEMP space straddles the center of your pack, if you want to make the area bigger or smaller, make the changes EQUALLY to both ends of the TEMP area. If you change TEMP space as I have suggested on FBA and/or CKD devices, you may want to back up your existing spool files with the CP SPTAPE command. If you forget to save the spool files, you can issue:

FORCE start

during VM initialization so CP can recover as many spool files as it can.

247

Even if your paging/spooling areas are defined as TEMP, CP does not intermix both types of records in the same cylinder. When CP gets its first request to write a spool block, CP assigns a whole cylinder or a group of FBA blocks to the spool function, and tends to write a bunch of consecutive spool blocks in that area.

On the other hand, when CP gets a bunch of pages to write out, CP tends to distribute the pages equally in a round-robin fashion, among all TEMP (or PAGE) devices in an attempt to balance out the paging load, if possible, across channels and devices. Therefore, it is desirable, if the paging rate is high, to spread CP paging over multiple channels and multiple devices to minimize page and I/O wait time and also reduce device, control unit, and channel contention.

Separating Paging From Spooling

If the volume of both paging and spooling activity is moderate, or if EITHER activity is high, there probably is no compelling reason to separate CP paging from spooling. If paging and spooling rates are both high, they tend to interfere with each other, particularly if printer spool files are not printed immediately when they are created.

If you have two different types of disks available, it is recommended that you put paging on the higher performance disks, unless its channel or control unit is already overloaded. When VM is IPLed, if CP finds PAGE and TEMP areas are defined, CP will do ALL paging in PAGE areas until the PAGE areas are full and then will start to use TEMP space. On the other hand, if CP runs out of TEMP space for spooling records, CP will NOT use PAGE space, so make sure you have enough TEMP space in your VM system.

Adding Paging or TEMP Space on a New Disk

If you want to create additional paging and/or spooling areas, and maybe a few CMS minidisks on a disk that is not currently CP-owned, you must do the following activities:

1. Using CP Format/Allocate, first FORMAT and then ALLOCATE the new pack as required.
2. Add the new VOLID to the SYSOWN list in DMKSYS, reassemble DMKSYS, and generate a new CP nucleus.

NOTE: In addition to allocating the desired areas as TEMP or PAGE, you must allocate ALL the rest of the pack as PERM, since any area NOT allocated defaults to TEMP not PERM.

3. At some convenient time, shut down the VM system and reIPL VM. If this is the first time you have defined PAGE space in your system, you should do a COLD start. If you are just adding more TEMP space and you added the new VOLID to the end of the SYSOWN list, the VM system operator can WARM start since all existing spool pointers are still valid.

Using CP Format/Allocate in a Virtual Machine

An IPLable copy of CP Format/Allocate (called IPL FMT) resides on the CMS 190 disk. To use that version, you must spool the IPL FMT program to your virtual reader and then IPL the reader address; for example:

1. SPOOL D to *
2. PUNCH IPL FMT S2 (NOH)
3. IPL C

The "NOH" parameter tells CMS not to put a

:READ

statement in front of the IPL deck, otherwise the IPL will fail.

Details

Assume that we are creating a 40-cylinder TEMP area in the center of a 3350 to be labeled VMPK02 at real address 142. Since a 3350 has 555 cylinders, the center of the pack is $555/2=277$. If we place 20 of the cylinders on each side of cylinder 277, the TEMP space will be well-positioned.

ATTACH the 3350 disk to your virtual machine.

ATTACH 142 *

If the disk already has the correct VOLID, it doesn't matter whether ALLOCATE or FORMAT is done first as long as both are done, but in this case, the disk is new or has DOS or OS label information which is incompatible with CP-owned volumes, so FORMAT must be done first.

ENTER FORMAT or ALLOCATE:	FORMAT
ENTER DEVICE ADDRESS:	142
ENTER DEVICE TYPE:	3350
ENTER START CYLINDER OR "LABEL":	000 0 258
ENTER END CYLINDER:	554 0 297
ENTER DEVICE LABEL:	VMPK02

You must format at least cylinder zero to create a valid CP-owned VTOC format and then re-invoke FORMAT again to format cylinders 258 to 297 into 4K pages. To check on the condition of the whole pack, it is recommended that you format all 555 cylinders. When formatting is completed, invoke the Allocate function.

```
ENTER FORMAT OR ALLOCATE:    ALLOCATE
ENTER DEVICE ADDRESS:       142
ENTER DEVICE TYPE:          3350
ENTER DEVICE LABEL:         VMPK02
ENTER ALLOCATION DATA:      TYPE CYL CYL
                             ....
```

If we wanted to look at the existing allocations, we would specify

END

and the following information would be displayed:

TEMP 000 554

which shows us the default status of this pack. We would then have to re-invoke ALLOCATE and repeat the above responses to enter our changes. We would key in:

```
PERM 000 257
TEMP 258 297
PERM 298 554
END
```

The next time VM is IPLed, this new TEMP area will be recognized and used by CP.

Since VM does not automatically keep a record of new CP-owned areas for use by minidisk mapping programs such as DIRMAP, remember to add the following statement in the VM Directory in USER \$TEMP\$ which is a fake userid entry for CP TEMP space:

MDISK D01 3350 258 040 VMPK02 R

Formatting TEMP Space on a 3370 FBA Disk

If we want to put 20,000 4K blocks of TEMP space on a 3370 starting at block 14,459, we would format blocks 0 to 2, and blocks 14,459 to 24,458.

Our allocations would be:

```
PERM 00000 14458
TEMP 14459 24458
PERM 24459 69749
```

Since, for FBA devices, the VM directory entries are in 512 byte blocks, we must multiply 14,459 X 8 to get 115,672. We must also multiply the 20,000 blocks by 8 to get 160,000.

Add the following entry in the VM directory to USER \$TEMP\$:

MDISK E01 FB-512 115672 160000 VMPK03 R

Looking at CP SYSRES's Allocation Information

The VM/SP SIPOE Planning Guide contains charts that show, for each supported DASD type, what areas of each disk is allocated for PERM, DRCT, TEMP, PAGE, and TDSK.

As you saw earlier, the VM Directory is also distributed with NOLOG entries for each occurrence of the above. This is so that you can issue the IPF DIRMAP command to see the layout of both CMS mdisks and CP system areas on each VM pack to avoid inadvertent overlaps.

Whenever you add or change any of the above allocation areas, you should manually update the VM directory source and invoke DIRMAP to verify the correctness of the change. Then issue the DIRECT (or DIRMAINT) command.

If you have a suspicion that there is an inconsistency between the allocation bit map and the DIRMAP listing, log onto MAINT (since MAINT owns the pack) and use the ALLOCATE function described previously to verify the minidisk mapping information.

Assume VMRES is at virtual 123.

Invoke the CP Format/Allocate program:

```
ENTER FORMAT or ALLOCATE:  ALLOCATE
ENTER DEVICE ADDRESS (cuu): 123
ENTER DEVICE TYPE:         3350
ENTER DEVICE LABEL:        VMSRES
ENTER ALLOCATION DATA:     TYPE CYL CYL
                           .... ... ..
```

To look at the existing allocations, specify

END

NOTE: This does not alter or harm the existing allocation information in any way.

The following information is then displayed:

```
PERM 000 100
DRCT 101 102
PERM 103 233
TEMP 234 276      (43 CYLIDERS)
PERM 277 277
TEMP 278 317      (40 CYLIDERS)
TDSK 318 417
PERM 418 554
```

ENTER FORMAT OR ALLOCATE:

To make any changes, specify ALLOCATE, and repeat the above responses. When you are done, key in END.

WARNING: NEVER KEY IN THE "DRCT" STATEMENT UNLESS YOU REALLY WANT TO CHANGE IT, SINCE KEYING "DRCT" INVALIDATES THE CURRENT VM DIRECTORY MAKING IT IMPOSSIBLE TO IPL VM!! If you have to change your VM Directory size while you are running under VM, you must invoke the DIRECT or DIRMAINT DIRECT command before you log off. If you are using the CP Format program standalone - not under VM - you must first dump your VM Directory onto cards or tape and then restore it with the standalone version of the Directory program - IPL DIR.

Reducing Paging, Spooling, or T-Disk Space

The size of the T-disk area distributed with the VM SIPOE is as large as it is to hold the 3705 Emulator Program (EP) when it is generated and maintained under CMS. If you don't maintain the EP program under CMS, you should be able to reduce the amount of T-disk by some percentage if you wish to. If you have added TEMP space to additional disks, you may also want to cut down the size of the original TEMP areas.

Even though CMS users may be using T-disk space and CP may be using the TEMP areas to be cut, it is possible to use the ALLOCATE function of CP Format/Allocate while VM is running to change the allocation bit map of the affected disk. As you will recall, the current allocation of TEMP space was 83 cylinders, 43 below the center of the pack and 40 cylinders above the center.

If we want to cut down the TEMP space by 20 cylinders, we should take 12 cylinders away from the bottom and 8 cylinders from the high end, that is, we need to change those cylinders from TEMP to PERM. If we also eliminate 50 cylinders from the high end of the T-disk area, we should key in:

```
PERM 234 245
PERM 310 317
PERM 378 417
END
```

I only have to key in the changes; the rest remain unchanged. The following results will be displayed:

```
PERM 000 100
DRCT 101 102
PERM 103 245      change
TEMP 246 276      change
PERM 277 277
TEMP 278 309      change
PERM 310 317      change
TDSK 318 377      change
PERM 378 554      change
```

The darker numbers indicate the changed values.

Warning No. 1 - Don't forget to update the NOLOG entries for \$TEMP\$ and \$T-DISK\$ to conform to these reduced sizes.

Warning No. 2 - Don't try to use those areas for anything new, such as CMS minidisks, since CP is still using them. Shut down and reIPL