VAXMATE MS-DOS V3.3 INFORMATION

FOR PCSA V.20

```
*******************************************
*                                         *
*  PCSA V2.0 PROBLEMS -- MSDOS 3.30 (VM)  *
*                                         *
*******************************************
```

SPR#	Component	Abstract
01719	Microsoft Code	Comma And Semicolon Legal Prefix To Commands
01159	Other	FC.EXE With /N And /LBN Switches
01161	Other	FC.EXE /B And /NNNN Switch Conflict
01629	Microsoft Code	Terminate And Stay Resident Doesn't Check Memory Allocation
00960	Format	FORMAT Marks Whole Tracks Bad, Not Just Clusters
01043	Format	FORMAT Does Not Automatically Ask For A Volume Label
01086	Other	FC.EXE: Does Not Take Wildcards
01351	Microsoft Code	CHKDSK Parsing Of Command Line
01518	Graphics	GRAPHICS Aborts During PRINT SCREEN on IBM

SPR #:	1719
Product:	MSDOS
Version:	3.30 (VM)
Class:	Feature
Component:	Microsoft Code

PROBLEM: Comma And Semicolon Legal Prefix To Commands

If you prefix any command with a ',' or a ';', it still works. This is true of PCDOS as well as MSDOS. Is this a feature?

ACTION/REMARKS:

This is a feature. The "parser" that command.com uses ASSuMEs that the ',' and ';' characters are white space. This can not be changed without a significant change to COMMAND.COM

SPR #:	1159
Product:	MSDOS
Version:	3.30 (VM)
Class:	Bug
Component:	Other

PROBLEM: FC.EXE With /N and /LBN Switches

I tried the following commands in a standalone environment with 584688 bytes of free memory (from the hard disk):

```
fc /n /lb115 autoexec.bat autoexec.bak  --- this was OK

fc /n /lb116 autoexec.bat autoexec.bak  --- received an "out of
                                            memory" error
```

ACTION/REMARKS:

This SPR was forwarded to Microsoft.

SPR #:	1161
Product:	MSDOS
Version:	3.30 (VM)
Class:	Bug
Component:	Other

PROBLEM: FC.EXE /B and /NNNN Switch Conflict

Environment: Standalone, hard disk <--> floppy

```
With the command:  fc /b /35 treed.com a:treed.com
         -or-      fc /35 /b treed.com a:treed.com
```

FC.EXE displays that the switches are incompatible. The Microsoft User's Guide for 3.3x shows this syntax as valid for binary comparisons.

ACTION/REMARKS:

This SPR will be sent to Microsoft.

SPR #L: 1629
Product: MSDOS
Version: 3.30 (VM)
Class: Vendor Related
Component: Microsoft Code

PROBLEM: Terminate And Stay Resident Doesn't Check Memory Allocation

When I issue the terminate and stay resident call and request more memory than the system has available, it doesn't return me an error to stop me from terminating. This has two bad side effects:

1) If on the VAXmate I am loading into network RAM and there is enough memory to load a module, but then it modifies it by specifying on the Keep Process call the amount of memory to keep, the program exists as through everything worked OK. The load module thinks it has the requested memory and the system hangs since the requested memory when the system tries to use it is the ROM data space.

2) If I am not loading into VAXmate extended memory, the effects are even more catastrophic. I get the following useless error message from DOS:

   Memory Allocation error
   Cannot load COMMAND, system halted.

   And my system is hung. Please notice it got the memory allocation error when it tried to load COMMAND and not the TSR.

   The obvious workaround is to modify the allocated memory block before you do the Keep Process call, but Keep Process should return an error if it can't make the memory allocation requested.

ACTION/REMARKS:

In the documentation of the terminate and stay resident call, it states that there is NO error returned from the call. What you should be be doing is verifying that you can get the amount of memory you desire, then do the call. Since this call provides a way of returning error codes to the parent process, you can return success when there is sufficient memory to allocate, and if not, then do à normal terminate, and return an error code.

Since this is a MS-DOS problem, and it will exist in PC-DOS as well as our version, I suggest that this SPR be sent to Microsoft.

SPR #:	960
Product:	MSDOS
Version:	3.30 (VM)
Class:	Issue
Component:	Format

PROBLEM: FORMAT Marks Whole Tracks Bad, Not Just Clusters

This isn't really a bug, since the FORMAT code was written to work this way, but it IS different from the way it worked in DOS 3.1. When FORMA encounters a bad track, it marks the ENTIRE track bad (9 sectors on low density, 15 sectors on high density), not just the single sector. A whole track is a significant amount of space on a floppy, and the chang would not be trivial to change the way it works.

On winchesters, format would have worked the same way if the DEC Media-scan routines were not there. The media scan will only mark sectors and their adjacent neighbors bad.

It is easiest to see this by making a radial scratch in a floppy, then watch half your space go away...

ACTION/REMARKS

SPR #:	1043
Product:	MSDOS
Version:	3.30 (VM)
Class:	Issue
Component:	Format

PROBLEM: FORMAT Does Not Automatically Ask For A Volume Label

Previous versions of FORMAT would prompt the user for a volume label, the latest version does not. The spec says that the user will be prompted for the volume label.

ACTION/REMARKS:

Easy work around. If you want a volume label format with the /V switch

This isn't a BUG, since it's not supposed to prompt unless you use the /V switch, but it IS different from the way it used to work under MS-DOS 3.10.

SPR #: 1086
Product: MSDOS
Version: 3.30 (VM)
Class: Issue
Component: Other

PROBLEM: FC.EXE: Does Not Take Wildcards

FC.EXE does not take wildcards like the IBM COMPs program.

ACTION/REMARKS:

FC has not and will most likely never take wildcards, so the actual suggestion would boil down to writing our COMP that would take wildcards.

The following short batch file will compare multiple files of the same name but different drive/directory. Suggest putting it in your search path:

Filename: WILDCMP.BAT

Contents:

```
ECHO OFF
for %%n in (%1) do fc %%n %2%%n
ECHO ON
```

Usage: 1) Go to the directory containing the source files to be compared.

2) Type: WILDCMP wildsource targetdir

for example, to compare all the "C" files in two directories, you might type:

```
WILDCMP *.C F:\TARGETS\
```

Where your sources are in the current directory and your comparison targets are in directory F:\TARGETS\*.C. Note the trailing \ is required.

We will be looking into areas like this for Version 2.1.

SPR #:	1351
Product:	MSDOS
Version:	3.30 (VM)
Class:	Won't Be Fixed
Component:	Microsoft Code

PROBLEM: CHKDSK Parsing Of Command Line

This SPR is mainly to bring up a problem with the command line parsing of CHKDSK when there is a switch between the command and the drive.

Following are some examples:

CHKDSK/F A:

This command will preform the CHKDSK on the current drive and not drive A.

CHKDSK/invalid_char A:

This command will give an invalid parameter error message and then continue on to perform the CHKDSK on the current drive rather than terminating.

ACTION/REMARKS:

I have been looking at CHKDSK and have determined what is expected in the command line. The drive to be checked must occur before the switch. If it occurs after the switch then it is the drive on which specific files are being checked for contiguity. If no drive is specified before the switch then the default drive is assumed. The following command will check drive C: and then check the file specifyed for drive A:

CHKDSK C:/F A:XXX.TXT

Attempting to modify the parsing of the command line may alter the expected operation of the command.

This SPR will be reopened for version 2.1 consideration.

SPR #:	1518
Product:	MSDOS
Version:	3.30 (VM)
Class:	Won't Be Fixed
Component:	Graphics

PROBLEM: GRAPHICS Aborts During PRINT SCREEN on IBM

This problem occurs while running on an IBM with the printer redirected to the serial port using the IBM supplied MODE. The printer is setup fo hardware handshaking using READY/BUSY. GRAPHICS aborts when the printer buffer fills and READY therefore goes low. The timeout for the comm por is in the area of 1-2 seconds.

This problem could be solved by reattempting to send the character when running on the IBM. Although doing this may extend the timeout period when running with a parallel printer since this timeout period for a parallel printer may be longer. On the IBM there is not method to determine if a parallel printer port is redirected.

This problem does not occur on the VAXmate since the VAXmate ROMBIOS handles the timeout when a parallel port is redirected to a serial port such that the timeout period is sufficient. Using XONXOFF on the IBM allows GRAPHICS to work properly.

ACTION/REMARKS:

Additional testing shows the timeout period on the XT to be 30 seconds if a parallel printer is being used. Therefore using the retry method would not work as this would increase the timeout period considerably for a parallel printer and there is not method that I know of to determine if the printer is redirected. The VAXmate firmware and XONXOFF handled this by doing the retries internal to INT 17, if the printer was redirected, to provide the appropriate delay. If this cannot be corrected then the problem could be corrected by running XONXOFF. The only supported printer that this problem will effect is the LA50 since it is the only one which will operate using hardware handshaking.