

**MaKe Boot Manager PARTition
(MKBMPART)
Users Guide**

© Copyright David G. Miller
MKBMPART Program and Documentation
All rights reserved
(This is an unpublished work)

Licensing Notes

This documentation accompanies a beta release of the MaKe Boot Manager PARTition (MKBMPART) program. This program together with its accompanying files and documentation is copyrighted material by the author and is being released in a limited form by the author for additional testing. Please **do not** disseminate this program in its current form without the consent of the author. I do not intend to change the terms of the registration as described in Chapter 5 of this document which means unauthorized dissemination of the program does not do anyone a service and may simply spread around copies of the program that could contain some bugs that my testing has not uncovered. The program will be released as "honorware" after completion of beta testing (Shareware assumes that you will evaluate the program for some limited period of time and, after this period and sufficient reminders to register the program, you will either register the program or discontinue its use. This program will, in general, be used once by a user. If a user chooses to compensate me for my time and effort in developing this program after this use, I congratulate him or her on his or her honesty and integrity). Beta testers need only tell me the results of their testing to register.

License Agreement

This is a single copy software license granted by David G. Miller, the author of the work, with a mailing address of 2615 Plaza Del Amo #620; Torrance, CA 90503. The MKBMPART software and documentation is licensed to you. It is not sold.

The MKBMPART program and documentation are copyrighted material. You have the non-exclusive right to use this program. You may make additional copies of this program for backup purposes so long as these copies, together with your original, are kept in your possession or control.

You may not make changes or modifications to the MKBMPART software, and you may not decompile, disassemble, or otherwise reverse-engineer the software. You may not rent it or lease it to others.

This software is Commercial Computer Software under Federal Government Acquisition Regulations and agency supplements to them. The MKBMPART software and documentation is provided to the Federal Government and its agencies only under the Restricted Rights Provision of the Federal Acquisition Regulations applicable to commercial computer software developed at private expense and not in the public domain. The use, duplication, or disclosure by the government is subject to the restrictions set forth in subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at 252.227-7013.

Limited Warranty and Limits of Liability

The author, David G. Miller, makes no warranty or representation, either expressed or implied, with respect to this software, its quality, performance, merchantability, or fitness for a particular use. As a result, this software is licensed "as is", and you, the licensor, are assuming the entire risk as to its quality and performance. In no event shall the author be liable for direct or indirect, special, incidental, or consequential damages resulting from any defect in the software or its documentation. Such damages include, but are not limited to, loss of profit, loss of programs, and loss of data. In no case shall the author's liability exceed the license fee paid to use MKBMPART.

The warranty set forth above is exclusive of all others, oral or written, expressed or implied. This warranty provides you with specific legal rights. You may have other rights which vary from state to state.

Table of Contents

1.	Introduction	1
2.	Before Running the Program	2
2.1	Where should MKBMPART make the boot manager partition?	2
2.2	What to Name the Bootable Partitions	3
2.3	Where are you going to install OS/2 and Primary vs. Extended?	3
2.4	What Else You'll Need	4
3.	Running the Program	5
4.	After Running MKBMPART	5
5.	How MKBMPART Works	7
6.	If Things Go Wrong	7
	Appendix: Error Messages	9

1. Introduction

New users of OS/2 2.1* are faced with choosing between three possible installation options:

1. Installing OS/2 as the operating system on the primary partition of their hard disk.
2. Installing OS/2 with the "dual boot" option so both OS/2 and their previous operating system (usually some version of MS-DOS or PC-DOS) co-exist in the same partition.
3. Installing OS/2 into an extended partition using the "boot manager" option.

Each of these options have advantages and significant disadvantages for the new user. Installing OS/2 as *the* operating system may be what IBM would prefer that you do but it makes "going back" difficult and means that you can not use several OS/2 incompatible programs. On the surface, dual boot seems like the best answer since it means you can still boot to DOS whenever you need to. Unfortunately, dual boot also means that your C: partition and root directory get rather crowded with the files and directories for two operating systems plus a system failure during a boot change can leave you with a system that can only be booted from floppy and hopefully one of the file recovery programs can sort out what happened to your hard drive. Finally, there is boot manager that lets you install OS/2 into another primary partition or an extended partition (the partition can even be on a second hard disk). The "gotcha" with boot manager is you must create a partition on your primary hard disk for boot manager which means deleting an existing partition and then creating a new partition for boot manager and a truncated partition from the remaining free space. Any files or programs that were in the deleted partition have now been lost and must be restored from backup or re-installed. To add insult to injury, the boot manager program only requires 14 K bytes of disk space but the FDISK programs (both DOS and OS/2) insist on a minimum partition size of 1 MB or one complete cylinder.

When confronted with this quandary, I initially opted for dual boot. I liked what I saw in OS/2 and luckily had created several partitions on my hard disk way back when. When I wanted to switch to boot manager, I was able to do some logical drive to logical drive copies, delete my primary partition, create a boot manager partition, and restore my C: drive with more logical drive to logical drive copies. This was a hassle and I was lucky to have enough free disk space and to have created several logical drives. I still didn't like devoting a full 1 MB to a boot manager partition when only 14 KB is required.

As with most other people in this situation, I wondered how hard it would be simply clear out sufficient space on my hard disk to create the boot manager partition without going through the whole process of deleting a partition, creating the boot manager partition, etc., etc. The difference was that some previous work I had done with DOS partitioned hard disks meant that I either knew the answers to these questions or could easily find out. A little research and an initial prototype showed that the answer was it wasn't that hard at all! A little more research though (like trying to install OS/2 with boot manager using the newly created boot manager partition) showed that I wasn't there yet. OS/2's FDISK program didn't like my abbreviated boot manager partition. It wanted 1 MB of free disk space in which to create a partition for and install the boot manager program.

* OS/2 is a trademark of International Business Machines Corporation (IBM). All trademarks mentioned in this document are the property of their respective owners. These trademarks are mentioned in this documentation only for clarity; this use *does not* reflect any endorsement or fitness of use of MKBMPART by the trademark owner nor by me of the trademarked software.

I could have left "things" at this point but felt that solving the whole problem also meant

minimizing the boot manager footprint. A little more research was required to come up with how boot manager is installed and what is required for the OS/2 installation program to recognize the installation. The final result is the program MKBMPART (**MaKe Boot Manager PARTition**). MKBMPART adjusts the size of an existing partition to create a partition for boot manager from free space available and installs the boot manager program.

2. Before Running the Program

MKBMPART can create a partition for the boot manager program and install boot manager. In order to do this, you must make two decisions:

1. Where should MKBMPART make the boot manager partition?
2. What do you want to call your current primary partition in the boot manager menu?

Sections 2.1 and 2.2 provide some guidance in how to answer these questions. Section 2.3 provides some general information about OS/2 installation considerations that may affect how you answer the previous questions. Section 2.4 describes what you'll need in order to run MKBMPART.

2.1 Where should MKBMPART make the boot manager partition?

This question is actually fairly easy to answer since there are only two possible answers. Due to the nature of the File Allocation Table (FAT) file system and the way extended partitions "work", MKBMPART can only create the boot manager partition at 1) the end of the (DOS) primary partition or 2) at the end of an extended partition (i.e., at the end of the space allocated to the last logical drive in the extended partition). Furthermore, the end of the extended partition can only be used if the last logical drive in the extended partition uses the FAT file system. If you only have a single primary partition on your hard disk but have two hard disks, the question is moot since there is only one possibility. If you have an extended partition and the last logical drive uses the FAT file system then this is the recommended location. If you only have a single hard disk and you didn't divide it into logical drives, MKBMPART can't help you unless you either buy another hard disk or change the partitioning of your existing hard disk. Sorry.

Once you have made this decision, you must ensure that no files use the "end" of this partition. This can usually be accomplished by running a disk defragmentation program. MKBMPART will tell if there are any allocated clusters or clusters marked as unusable in the region required for the installation of the Boot Manager program. Simply run the program and see which partitions it identifies as suitable or unsuitable. Disk diagnostic programs such as the Norton Utilities or PC Tools can be invaluable in identifying which file or files are "in the way".

If MKBMPART reports that the partition you would like to use is unsuitable after defragmenting there are two possibilities. The first is that there is an unusable cluster in the area that MKBMPART requires to be free. If this is the case, you need to either choose a different location or you can not use MKBMPART to make your boot manager partition. The second is that an "unmovable" file resides in this area. Since this is the "end" of the disk, this file is probably the result of running a system area save program (e.g., the MS-DOS 5.0 "mirror" program, Norton Utilities "Image" program, etc.). Type the command `dir/a:h` in the root directory of the drive and consider the files identified. The files MSDOS.SYS and IO.SYS (or their PC-DOS equivalents IBMDOS.COM and IBMIO.COM) are generally not the problem since these files are

at the very beginning of the drive. Any other files reported *may* be the culprits. Move the offending file(s) to a different location or delete them. If you are using a deletion tracking program that allows you to undelete files, you will also need to "purge" the file(s) after you delete them since the deletion tracking program is managing your disk's free space and the location these files occupied will still be marked as allocated in the FAT.

The files reported may have other attributes set such as "read only" and "system" that makes deleting them difficult. Consult your DOS documentation for how to change file attributes. Some disk defragmentation program allow you to see which files occupy a particular portion of your disk. Consult the documentation for whatever disk tools you have available.

After the release of OS/2 2.1, much ado was made about installing the boot manager program at either the very beginning of the drive or at the very end. This was because installation of the boot manager partition "in the middle" of a hard disk would result in one megabyte of "dead space" in the middle of drive that was never accessed once the system had booted. This would degrade disk performance since the disk's head would have to move back and forth across this space during normal system operation increasing the seek time. By using MKBMPART in either standard or aggressive mode this becomes less of an issue since the boot manager partition will only consume at most one cylinder of disk space. The existence of a single cylinder of dead space should have a negligible effect on the performance of the disk. The bottom line here is that you can install boot manager using MKBMPART wherever it will fit as long as you are using standard or aggressive mode.

2.2 *What to Name the Bootable Partitions*

If you have three bootable partitions, "Larry", "Curly", and "Moe" might be fun... You can call them anything you like as long as the names are eight (8) or fewer characters long. Most people find meaningful (but boring) names like "OS/2 2.1" and "DOS 5.0" to be more helpful in deciding which partition to boot from. The choice is yours and you can change the names by running OS/2's FDISK once you have OS/2 installed.

2.3 *Where are you going to install OS/2 and Primary vs. Extended?*

The considerations that go into answering these two questions are beyond the scope of this document. There are two things to consider:

1. If you want to use the high performance file system (HPFS) for your OS/2 partition, it should (but isn't required to be) the last partition on the physical disk so that drive letter assignments don't change between DOS sessions and OS/2 sessions. This does NOT present a conflict with the MKBMPART requirement that the last logical drive in the extended partition use the FAT file system if you want the boot manager partition created at the end of the extended partition and actually makes things easier. MKBMPART runs before you install OS/2. Thus, as long as this partition uses the FAT file system when you run MKBMPART, there isn't a problem. This situation makes the preparation for MKBMPART easier since it allows you to simply format what will be the destination OS/2 drive using the DOS format command. Formatting the drive easily removes any possible conflicting files.
2. If you install OS/2 into a primary partition on the first hard disk, the DOS primary partition will NOT be visible from an OS/2 session. Likewise, the OS/2 primary partition

will not be visible from a DOS session even if you elect to use the FAT file system for this partition. Regardless of whether you put OS/2 in a primary partition or in a logical drive in an extended partition, the OS/2 partition will not be visible from a DOS session if you elect to use the HPFS for it. Finally, if you install OS/2 in a logical drive in an extended partition or a primary partition on the second physical hard disk, all partitions will be visible to the OS/2 session and you will still be able to access the OS/2 partition from DOS running under OS/2 even if the OS/2 partition is formatted HPFS (but only those files and directories that are DOS legal). An advantage of putting OS/2 in an extended partition on the second physical disk is that drive letter assignments for any logical drives on the first physical disk do not change regardless of whether you boot to DOS or OS/2 even if the OS/2 partition uses the HPFS.

Note: I have had mixed success with OS/2's installation program updating the boot manager menu correctly when OS/2 is installed on the second physical disk. On my test system (80386DX/25 with two ST-251-1 MFM hard disks), the menu is constructed correctly. On my development system (80486DX2/66 with a SCSI hard disk), I have had to start OS/2 from floppy and use the installed copy of FDISK to first remove and then to add the OS/2 partition name to the boot manager menu *after* the installation was complete.

2.4 *What Else You'll Need*

In order to run MKBMPART you will need two blank, formatted floppy disks and disk 1 from the OS/2 2.1 distribution set*. The first floppy is used to hold the MKBMPART program and its related files. Since MKBMPART modifies the information that allows the data on your hard disk to be addressed, it would not be wise to run the MKBMPART program from this same hard disk. The second floppy is used to save a back-up copy of the system area of your hard disk. In case there is a problem with the operation of MKBMPART on your system, this back-up copy allows the RMBMPART (**ReMove Boot Manager PARTition**) program to restore your system to its original configuration. Disk 1 from the OS/2 2.1 distribution set contains the OS/2 FDISK.COM program. Within FDISK.COM is the boot manager program. MKBMPART extracts boot manager from FDISK.COM in order to install it on your hard disk (you may wish to use a copy of the OS/2 2.1 distribution disk for this purpose).

- * Note to owners of the CD-ROM version of OS/2 2.1: The CD-ROM version of OS/2 comes with four floppies (two pair of 5.25 inch and 3.5 inch floppies that are used to start the installation process). The floppy disk labelled "Disk 1" is *not* the same as Disk 1 from the floppy installation set. You will need to create a disk image using the LOADDISKF program of DISK 1 from the floppy distribution set. The diskette images can be found in R:\DISKIMGS\OS2\35 or R:\DISKIMGS\OS2\525 where R: is the drive letter of your CD-ROM drive and the "...35" subdirectory holds disk images for 3.5 inch floppies. The "...525" subdirectory holds the 5.25 inch floppy images. The disk size used should match your A: drive.

3. Running the Program

Start MaKe Boot Manager PARTition by typing MKBMPART and hitting the enter key in the directory where you have placed the MKBMPART files. MKBMPART allows you to quit at virtually any time up to the point at which you confirm your selections. Help is available from some screens by striking the "?" key. After you start the program, MKBMPART will ask you to choose between three boot manager partition size options. These options are:

1. Full Compatibility - This option allocates a full 1 MB to the boot manager partition. More than 1 MB may be allocated since complete cylinders are used.
2. Standard - This option allocates space for the boot manager partition on a cylinder boundary. At a minimum, one complete cylinder is allocated for boot manager.
3. Aggressive - This option allocates the minimum space possible to the boot manager partition. Some disk tools (the Norton Disk Doctor in particular) complain if a partition consists of other than complete cylinders. Allocation is done in complete tracks. At least one complete track is allocated to boot manager.

MKBMPART will examine your system to determine which location(s) is(are) suitable for making the boot manager partition based on the boot manager partition size option chosen. MKBMPART will then ask you to identify where you want the boot manager partition created. Once you have successfully provided this information, MKBMPART will ask you to provide a name for your current primary partition to be used in the boot manager menu. Finally, MKBMPART will ask you to confirm your selections. If you wish to change a selection, exit the program and re-start MKBMPART.

After you confirm your selections, MKBMPART will then ask you to provide a floppy disk so that it can save a snap-shot of your current system definition. This can be the same floppy as you use to run MKBMPART as long as there is sufficient space. After saving this snap-shot, MKBMPART will ask you to provide disk 1 from the OS/2 2.1 distribution set. This is the distribution disk that contains the OS/2 version of FDISK.

Once you provide the requested distribution disk, MKBMPART will request that you confirm that you want the partitioning of you hard disk to be modified and the boot manager program installed. If you confirm that MKBMPART should proceed, the partition table for your primary hard disk is modified and the boot manager program is installed. In addition, the boot record for the affected logical drive is updated to reflect the change in the space available on that drive. After the necessary data has been modified, MKBMPART will ask if it can re-boot your system so that the changes can take affect.

4. After Running MKBMPART

You can now install OS/2 but with one critical difference: boot manager is already installed. And your system will now start-up using boot manager to determine which partition to boot from. When you install OS/2, you will be presented with several introductory screens explaining the installation process. You will then be presented with a screen titled "Installation Drive Selection" (see page 64 of the *OS/2 2.1 Installation Guide*). You can either accept the default drive selection presented (C:) or choose the alternative "Specify a different drive or partition". You wouldn't have gone through running MKBMPART if you wanted the default so select "Specify different drive or partition". This starts the OS/2 FDISK program and allows you to identify another

partition and/or drive as the destination for your OS/2 installation and to add this partition to the boot manager menu*. Refer to the *OS/2 2.1 Installation Guide* for how to proceed from this point.

Had you not used MKBMPART, at this point you would have been forced to delete an existing partition in order to create sufficient free space to make the boot manager partition. All programs and data in the deleted partition would be lost and you would have to re-install or restore them from back-up. In addition, a full 1 MB of disk space would be allocated to a 14 KB program. I created this innovative program because I saw a need that was not being met by the existing OS/2 installation capabilities. For individual users, if you found this program useful, please let me know about it. I can be reached electronically at any of the following addresses:

Delphi: MILLER2615
CompuServe: 72040,1017
Prodigy: CGVD94A

I can be reached from the internet as either:

miller2615@delphi.com or
<72040.1017@compuserve.com>.

If you wish, a \$5.00 (U.S) registration fee gets you good karma and supports me in creating additional innovative programs such as this (I have several ideas for hybrid DOS-OS/2 programs and pure OS/2 programs). The \$5.00 registration will be credited toward the registration of a future program I release that you choose to register. Corporate users can obtain a site registration for \$50.00 (U.S.). This site registration entitles you to the same good karma as individual users get (but for your complete site) and e-mail and phone support. I consider this pricing reasonable since corporate users are more likely to use this program on multiple machines and thus have a higher likelihood of needing support. The program is fully functional in its distributed form so registration is simply an act of honesty.

Registrations can be mailed to:

Mr. David G. Miller
2615 Plaza Del Amo #620
Torrance, CA 90503

- * During testing, I noticed that OS/2's FDISK program did not identify the Boot Manager partition created by MKBMPART as "STARTABLE". Not sure why this occurred since running FDISK under OS/2 (once it was installed) did correctly identify the Boot Manager partition as STARTABLE. All you need to do under FDISK is identify a partition as "INSTALLABLE" after setting up Boot Manager using MKBMPART. Do not be concerned that FDISK does not say that the Boot Manager partition is STARTABLE.

5. How MKBMPART Works

MKBMPART takes advantage of the simplicity of the FAT file system to chop off a small portion of one logical drive to make room for the boot manager partition. In particular, the FAT file system never "looks at" the partition table except when a logical drive is being formatted or when FDISK is used to modify a hard drive's partitioning. Instead, the size of a logical drive is determined by entries in a portion of the boot record known as the BIOS Parameter Block (BPB). This approach is probably a carry over from DOS's early years as a floppy disk based operating system. Floppy Disks do not have a partition table and all information about the size of the disk is in the BPB. By changing the number of sectors on the logical drive in the BPB, the size of the drive can easily be changed by a small amount.

In theory, any change to the number of sectors available should also be reflected in the amount of disk space allocated to the file allocation tables. A little arithmetic shows that even if the full 1 MB of disk space is allocated to the boot manager partition, very little space would be gained by adjusting the size of FAT to reflect the changed size of the disk. For the worst case, assume the logical drive being shortened has 2 KB allocation units (also known as the cluster size). 1 MB of disk space then requires 512 FAT entries. Each FAT entry is a 16 bit or two byte number so 512 FAT entries take a total of 1 KB of disk space. Since there are two copies of the FAT for a disk, we double this amount to get 2 KB of disk space taken up by the FATs should be freed for a 1 MB boot manager partition. Since this is all of one 2 KB allocation unit, I decided that it isn't worth the trouble or the risk to recover it. Testing of MKBMPART showed that DOS is insensitive to having more space available for the FATs than is required.

If you use either the standard or aggressive partition sizes for the boot manager partition, the amount of space "lost" in unneeded FAT sectors rapidly shrinks to insignificance. Since disk space for each copy of the FAT is allocated in whole sectors, it is very likely that the amount of space required for the FATs will not change as a result of creating the boot manager partition using the aggressive mode.

In addition to the changes MKBMPART makes to the boot record of the selected drive, boot manager makes changes to the primary partition table for the hard drive to add the boot manager partition. If you choose to make the boot manager partition at the end of an extended partition, boot manager must also update the chain of extended partition definitions. This update consists of simply shortening the length of the partition.

6. If Things Go Wrong

For any number of reasons, you may decide that what MKBMPART did to the logical organization of your disk isn't quite what you wanted. Included with the MKBMPART distribution file set is a program call RMBMPART (ReMove Boot Manager PARTition). This program can restore your system to the state it was in prior to when you ran MKBMPART. In order for RMBMPART to do this, it must be able to access the data saved by MKBMPART when you originally installed the boot manager program. To restore your previous configuration, simply run the RMBMPART program. This program will prompt you to provide the floppy disk that has the system image saved by MKBMPART. It will restore your partition table(s) and the boot record of the logical drive that was modified to make room for the boot manager program to the previous settings. When RMBMPART completes, your system ***must*** be rebooted for the changes made by RMBMPART to take affect.

Please note: MKBMPART and RMBMPART make no provision for renaming the file containing the system image saved as part of the installation process. If you use the same floppy to save the system image for several systems, the last system's image will be the only one that "survives". If there is sufficient interest and demand (read that as paying registrations plus people who need the capability), I can add a capability to append additional images to the file rather than overwriting the existing image and to provide identification information for each image.

Appendix: Error Messages

The following messages may be displayed by MKBMPART if it encounters certain conditions that should not occur. Please make a note of the message and contact the author for technical assistance. For each possible error, a brief explanation is provided and, if possible, appropriate remedial action. In general, on a properly operating system, these errors should never occur.

1000 Unable to read primary partition table.

The DOS service used to read the partition table was unable to read the sector at cylinder 0, head 0, sector 1. If your system otherwise operates correctly, this means things are pretty strange.

1001 Primary partition table has four entries. No room for Boot Manager entry.

A partition table can have at most four partitions defined. Your primary partition table already has four entries thus leaving no room for the Boot Manager partition entry.

1002 No bootable partition in primary partition table.

MKBMPART was unable to find a partition marked as bootable (or startable) in your primary partition table. Use FDISK to make a partition "active".

1003 Unable to read extended partition table.

The DOS service used to read the extended partition tables was unable to read one of them.

1004 Unable to read boot record for candidate partition.

The boot record is on the first sector in a partition. For FAT file system drives, it contains the number of sectors for the drive. This number must be changed in order to define the Boot Manager partition.

1005 Unrecognized disk logical structure (only one FAT).

One of the parameters in the Boot Parameter Block of the boot record that defines the drive's logical structure indicates that only a single copy of the FAT is defined for this disk. This is not standard and MKBMPART will not work on such a drive.

1006 Unable to retrieve FAT for candidate partition.

The DOS service used to read the File Allocation Table (FAT) indicated that an error occurred while attempting to read the FAT.

1007 Unable to allocate memory required to build boot manager program.

MKBMPART was unable to allocate temporary memory. Given the small size of this program, it is unlikely that a system will be encountered that has insufficient memory for the operation of MKBMPART.

1008 Unable to read file 'FDISK.COM'.

MKBMPART attempts to open (read only) the file FDISK.COM in order to retrieve the Boot Manager program. Either the open or the file read failed.

1009 Unable to modify boot record for changed partition size.

MKBMPART was unable to rewrite the boot record to reflect the change to the drive's size.

1010 Unable to write partition table.

A write error occurred when MKBMPART attempted to write the modified partition table with the changes required for the installation of the Boot Manager partition.

1011 Bad data value in partition table.

An invalid data value was encountered when MKBMPART was attempting to modify the partition table. Check the partition table using a disk integrity tool.

1012 Unable to write boot manager program to boot manager partition.

A write error occurred when boot manager attempted to write the Boot Manager program to the intended location for the chosen partition creation strategy. This could occur if a marginal sector exists in the region MKBMPART will use to install the boot manager program. Depending on the type of hard disk involved, you may be able to "revive" the sector using a tool such as "Spinright" or Norton's Calibrate. If all else fails, choose a different partition creation strategy since this will result in the Boot Manager program being written to a different location on your disk.

1013 Unable to restore previous boot record data (write error).

An error occurred when RMBMPART attempted to restore the previous boot record data.

1014 Invalid extended partition definition encountered.

An extended partition table should have exactly one or two entries. If it has more than two entries, MKBMPART does not know how to deal with this type of extended partition.

1015 More than the maximum number of logical drives have been found.

MKBMPART assumes that no more than 26 logical drives can be defined.

1016 Unable to determine logical drives defined on system.

MKBMPART queries DOS to determine the characteristics of the logical drives currently defined on the system. If the query fails, MKBMPART can not continue.