Scripting

Click a link below to view the following information:

- Using Script Processing
- Sample Scripts
- Script File Statements for PartitionMagic
- Script Suggestions and Notes
- Sample Scripts

Using Script Processing

You can use PartitionMagic scripts to make changes to the partitions on a machine. A script is an ASCII text file with text statements that define the operations you want to perform. You can create a script file with any text editor.

To play the script from ScriptBuilder, you click **Script ➤ Play** or **Save and Play**. If the program encounters an error, the script will end immediately and the error is displayed in the message pane.

When executing from a script, if the program encounters an error it will terminate immediately, without processing the rest of the script. The user can determine if an error has occurred and what error it was by looking at the log file or error file. These two files are only created if specified by the user on the command line.

Each operation in a script is performed on the partition that was last specified. You must specify the correct partition before running a script.

The script text file allows comments in the C++ form (//). These can be either a full line or after a valid statement.

To use scripting to configure remote systems, you must be able to do the following:

- Set up remote access to your machine.
- Load PartitionMagic on a remote system.
- Launch PartitionMagic on a remote system
- Get feedback as necessary from a log file on the remote system

Command Line Options

The following command line options are supported by the DOS version of PartitionMagic.

When you specify multiple options, the order is unimportant.

/CMD

This is the parameter that will be used to pass the name of the script file to the program. For example, if the script file were named SCRIPT.TXT, the syntax for running the program from the script would be:

```
PQMAGIC /CMD=SCRIPT.TXT
```

/LOG

Whenever the program is run from a script, it is suggested that a log of all that occurred should be kept. The /LOG parameter is not unique to the site license version, but most useful when used in conjunction with the /CMD parameter to review exactly what transpired during the script execution. The /LOG parameter will specify the name of a file where all output will be directed. The output will appear as if a user had been executing the program through keyboard input, displaying each script command and all that transpired because of that command.

To specify a log named RESULTS.FIL (with the above script file), the command line would be:

```
PQMAGIC /CMD=SCRIPT.TXT /LOG=RESULTS.FIL
```

IMPORTANT! Do not attempt to modify the partition on which the log file is created. The log file is created on the partition that PQMAGIC is run from. If you need to modify this partition don't specify the /LOG switch. If you do, you will damage your partition.

/ERR

The Error parameter is used to specify a file to be created if the program should terminate with an error. Because many operations will require the machine to be rebooted following script execution, it will not always be possible to return the error to the process that called the PQMAGIC program. For this reason, if the program should end in an error and the /ERR parameter is specified, the error number will be placed in the file designated by /ERR. If /ERR is specified and the program terminates WITHOUT an error, the file specified will be deleted if it exists. Using this parameter, the user can write a program to just check for the existence of the error file to determine if the script was run successfully. Even if this parameter is not used, the error number will appear in the log file (if specified by /LOG) along with a text description of the error that occurred. To use the error parameter with the above parameters, the syntax would be:

```
PQMAGIC /CMD=SCRIPT.TXT /LOG=RESULTS.FIL /ERR=ERROR.FIL
```

/NRF

The No Run File parameter is used to specify the name of a file, which if it exists, will prevent the script from executing. This parameter is used to keep the program from running a second time, if it were placed in a login script or autoexec.bat file. If the user had specified a /LOG file or a /ERR file, the /NRF parameter could check for the existence of these files and prevent the program from running if either existed. For example, if a script were run with the parameters specified above (in the /ERR option), using the syntax shown below would prevent the program from running if the RESULT.FIL existed because the program had been run once already.

You can use the /NRF parameter more than once on the command line if it makes sense to check for more than one file.

```
PQMAGIC /CMD=SCRIPT.TXT /LOG=RESULTS.FIL /ERR=ERROR.FIL
/NRF=RESULTS.FIL
```

/SCO

The Syntax Check Only parameter is used to check the syntax of a script. It will make sure that a partition is always selected before an operation is executed and check the syntax of all the script commands. It will also check to ensure that any labels specified in a select partition statement are unique. It will not actually run the script. The syntax check will not detect logical errors such as trying to move the partition when there is not space to move.

This parameter can be used with the /LOG file if desired. A successful syntax check will show a statement saying that it was successful. Usage for the /SCO parameter would be as follows:

```
PQMAGIC /CMD=SCRIPT.TXT /SCO
```

Script File Statements for PartitionMagic

Several special characters are used when describing the syntax of script file statements. These are described below. Script file statements are not case-sensitive. Before performing an operation, you must first select the drive and partition that you wish to act upon.

{}- Denotes a required parameter

[] - Denotes an optional parameter

| - Denotes a choice among two or more options

Allow Manual Reboot

Allow the script to run even if it is determined that the program cannot reboot the machine remotely, after changes are made. This should be the first statement if used.

No parameters.

Bad Sector Retest

Retest the current partition for bad sectors and unmark any bad sectors that have been set incorrectly.

Check

Use to check a selected partition for errors.

No parameters.

Cluster Analyzer

```
[/ClusterSize={ 512 | 1 | 2 | 4 | 8 | 16 | 32 | 64 }]
[/ShowClusterWaste]
[/SetToRecommended]
```

Get Cluster Analysis information about a particular partition.

If this operation is used with out any parameters it will output a Cluster Analysis screen.

Parameter	Description
/ClusterSize	Changes the cluster size to the specified size.
/ShowClusterWaste	Shows the Cluster Analysis Screen.
/SetToRecommended	Sets the ClusterSize to the recommended size.

Convert To FAT

Convert a FAT32 or NTFS partition to FAT.

Convert To FAT32

Convert a FAT or NTFS partition to FAT32.

Convert To HPFS

Convert a FAT partition to HPFS.

Convert to NTFS

Convert a FAT partition to NTFS. Under Windows 2000, you can convert a FAT32 partition to a NTFS partition.

Convert To Primary

Convert a logical partition to a primary partition.

Convert To Logical

Convert a primary partition to a logical partition.

Copy

The Copy command should be preceded by the following commands:

- Select Drive {Num}
- Select Partition {PartitionLetter | "Volume Label" | Extended | Next | Previous | Num }
- Select Copy Drive {Num}
- Select Copy Partition {PartitionLetter | "Volume Label" | Next | Previous | Num }

For the Copy command to work correctly, a drive and partition need to be selected and a copy drive and copy partition need to be selected. If the selected partition is free space and the partition specified by the copy partition is a partition smaller than the free space, this operation will copy the specified partition to the free space.

If the selected partition is a valid partition and the partition specified by the copy partition is a block of free space large enough to hold the partition, the copy operation will copy the selected partition to the specified free space.

Create /FS={FAT | FAT32 | HPFS | LINUXEXT2 | LINUXSWAP | NTFS | EXTENDED | UNFORMATTED}

```
[/Label="NEW LABEL"]
[/Size=Value]
[/Position={BEGINNING | END}]
```

Create a new partition, and, optionally, format it.

Parameter	Description
/FS	*(Required)* It can be any of the above specified strings. There may be cases where creating with a certain /FS would fail. For example, trying to create an Extended partition when one already existed.

Parameter	Description
/Label	*(Optional)* Replace "NEW LABEL" with the desired volume label. It must be 11 characters or less for FAT partitions. Labels must be16 characters or less for Linux Ext2 partitions and 32 characters or less for NTFS partitions. The label must be in double quotes. The script may fail if invalid characters are entered.
/Size	*(Optional)* Specified in megabytes and will default to the size of the free space if not specified.
/Position	*(Optional)* Must be followed by either END or BEGINNING to specify where the partition will be created in the free space.

Delete {Volume Name | "NO NAME" | "LINUXSWAP" | "UNKNOWN" }

Use to delete a selected partition.

Parameter	Description
Volume Name	Required to delete a partition UNLESS the partition is either an extended partition, Linux swap partition, or an unformatted partition. This is a check to ensure that you are destroying data in the proper partition only. The value entered must always be preceded and followed by double quotes. If the partition label is blank and the partition is FAT or HPFS, the label entered should be "NO NAME". If the partition is not FAT or HPFS then "UNKNOWN" should be entered.

Format {Volume Name | "NO NAME" | "UNKNOWN" }

```
/FS={ FAT | FAT32 | HPFS | LINUXEXT2 | LINUXSWAP | NTFS }
[/Label="NEW LABEL"]
```

Use to format a selected partition.

Parameter	Description
Volume Name	Required to format a partition UNLESS the partition is either an extended partition or an unformatted partition. This is a check to ensure that you are destroying data in the proper partition only. The value entered must always be preceded and followed by double quotes. If the partition label is blank and the partition is FAT or HPFS, enter “NO NAME” as the label. If the partition is not FAT or HPFS, enter "UNKNOWN" as the label.
/FS	*(Required)* Can be any of the above specified strings. There may be cases where formatting with a certain /FS would fail, for example trying to format a FAT partition past 1024 cylinders.
/Label	*(Optional)* Replace "NEW LABEL" with the desired volume label. It must be 11 characters or less for FAT partitions. Labels must be16 characters or less for Linux Ext2 partitions and 32 characters or less for NTFS partitions. The label must be in double quotes. The script may fail if invalid characters are entered.

Hide

Hide the currently selected partition.

Info

```
[/Usage ]
[/Waste ]
[/Partition ]
[/FS] -------> {FAT | FAT32 | HPFS | NTFS}
```

Use to get information about a selected partition.

Parameter	Description
/Usage	The Disk Usage screen is available for the FAT, FAT32, NTFS, and HPFS file systems. This screen shows you the following information in bytes, megabytes, and as a percentage: • Used space on the partition, including space wasted by clusters • Free space on the partition • Bad space on the partition • Total space on the partition (found by adding the three previous lines)
/Waste	The Cluster Waste screen applies only to partitions that use either the FAT or FAT32 file system. This screen shows the following: • Current cluster size in bytes or kilobytes • Data stored on the partition in bytes and megabytes • Wasted space on the partition in bytes and megabytes Total Used space in bytes and megabytes (found by adding the numbers on the two previous lines)
/Partition	This screen is available for all types of partitions, including free space and extended partitions. Information on this tabbed page includes the following: • Partition type is shown in hexadecimal followed by a text description of the partition or file system type (such as FAT, FAT32, NTFS, HPFS, and so on). The hexadecimal designation is a conventional way to display partition types. • Serial number is shown here if the partition's file system uses serial numbers. Not all file systems use serial numbers.

Parameter	Description
/Partition (continued)	The next section of the screen shows physical information about the partition, including the following: • First Physical Sector shows the logical number and the location (cylinder, head, and sector) where the partition begins. • Last Physical Sector shows the logical number and the location (cylinder, head, and sector) where the partition ends. • Total Physical Sectors gives the number of sectors in the partition. • Physical Geometry shows the number of cylinders, heads, and sectors of the physical disk drive on which the partition resides.
/FS FAT	This screen applies only to partitions that use the FAT or FAT32 file system. The first section on this page provides the following information about the file system: • Sectors per FAT • Root directory capacity • First FAT sector • First data sector The next section of this page gives the following information: • Number of byes in files on the partition, the number of riles, and the number of those files that are hidden • Number of bytes in directories on the partition, the number of directories, and the number of those directories that are hidden Several extensions to the FAT file system exist. The final section of this page gives the following information about FAT extensions: • Number of byes used for OS?2 Extended Attributes and how many files and directories the Extended Attributes are associated with Number of byes used for long file names and the number of files and directories the long file names are associated with

Parameter	Description
/FS NTFS	This screen contains information that pertains only to NTFS partitions. This screen shows the following file system information for the selected partition: • NTFS Version shows the version number. The most recent version is 1.2. • Bytes per NTFS sector displays the number of bytes in each logical sector on the selected partition. (There are always 512 bytes in each physical sector.) • Cluster size • First MFT Cluster • File Record Size The next section shows information similar to that shown by NTFS CHKDSK, including the following: • Number of files and the byes and clusters allocated to them • Of the clusters used in files, the number of wasted bytes resulting from the cluster size • Number in indexes (directories) and the space allocated to them, shown in bytes and clusters • Space reserved for other system structures, shown in both bytes and clusters
/FS HPFS	This screen applies only to partitions that use the HPFS file system. The first section on this page provides the following information about the file system: • Partition status (that is, is partition active?) • DirBlock sectors • Free DirBlocks • Hot Fixes Used

Parameter	Description
/FS HPFS (continued)	The next section of this page gives the following information: • Number of bytes in files on the partition, the number of files, and the equivalent amount of sectors • Number of bytes unused in file sectors • Number of bytes in directories on the partition, the number of directories, and the equivalent number of sectors • Number of bytes in file/dir Fnodes and equivalent sectors • Number of bytes reserved by system and equivalent sectors • Number of bytes in extended attributes

Label

```
[/GetLabel ]
[/SetLabel="" ]
```

This option lets you change the name of a selected partition. Labels can be up to 11 characters long for FAT, FAT32, and HPFS partitions, 32 characters long for NTFS partitions, and 16 characters long for LinusExt2 partitions. Labels for FAT, FAT32, or HPFS partition types follow the same rules as DOS names, with two exceptions: spaces are allowed, and no period is required between the first eight characters and the last three.

Merge {Target First | Target Second}

Use Merge to join two FAT or FAT32 partitions that are adjacent (unallocated space can exist between them, however) to each other on a hard disk. This is useful if you have reached the maximum number of partitions on your disk, but you do not want to delete a partition.

The Merge command must be preceded by the following syntax in the script edit pane:

```
Select Partition {{Number} | {PartitionLetter} | {"Volume
Label"} | First | Last | Next | Previous | Extended}

Select Merge Partition {Next | Previous}
```

For the Merge command to work correctly, two adjacent FAT or FAT32 partitions must be selected.

Parameter	Description
FAT | FAT32	(Optional) You can specify the resulting file system, or you can let Merge choose the best file system automatically.

Move Left {Max | Min | Value }

Move a partition to the left. If the partition is the extended partition, only the right boundary is changed.

Parameter	Description
Max	Move the partition as far to the left as possible. Flush with the previous partition or beginning of the drive.
Min	Move the partition to the left, the minimum amount possible (1 cylinder).
Value	Move the partition left by the amount of the value specified (in megabytes).

Move Right {Max | Min | Value }

Move a partition to the right. If the partition is the extended partition, only the right boundary is changed.

Parameter	Description
Max	Move the partition as far to the right as possible. Flush with the previous partition or beginning of the drive.
Min	Move the partition to the right the minimum amount possible (1 cylinder).
Value	Move the partition right by the amount of the value specified (in megabytes).

Move Space Before {Max | Value }

Same as Move Right.

Parameter	Description
Max	Same as Move Right Max. Makes as much space before the partition as possible by moving the partition right.
Value	Moves the partition right such that the space before is equal to the value specified if possible (specified in megabytes).

Move Space After {Max | Value }

Same as Move Left.

Parameter	Description
Max	Same as Move Left Max. Makes as much space before the partition as possible by moving the partition right.
Value	Moves the partition left such that the space before is equal to the value specified if possible (specified in megabytes).

Resize {Max | Min | Value } [/Clustersize={512 | 1 | 2 | 4 | 8 | 16 | 32 | 64}]

Resize a partition.

Parameter	Description
Max	Resizes to the maximum size possible. The right edge of the partition will be flush with end of drive or next partition, if possible.
Min	Resizes to the minimum possible (determined by the size of the data).
Value	Resize to value specified (in megabytes).
/Clustersize	Will set the cluster size to the size specified during the resize operation. The cluster size must be valid for the partition size specified.

Resize Larger {Max | Min | Value} [/Clustersize={512 | 1 | 2 | 4 | 8 | 16 | 32 | 64 }]

Resize a partition larger by specifying the incremental change in size.

Parameter	Description
Max	Same as Resize Max. Make the partition as big as possible.
Min	Grows the size of the partition by the smallest amount possible (1 cylinder).
Value	Grows the size of the partition by the size specified (in megabytes).
/Clustersize	Sets the cluster size to the size specified during the resize operation. (The cluster size must be valid for the partition size specified.)

Resize Left Boundary {Max | Min | Value }

Resize the extended partition by moving the left boundary. This operation is for extended partitions only. .

Parameter	Description
Max	Resizes to the maximum size possible. The right edge of the partition will be flush with end of drive or next partition, if possible.
Min	Resizes to the minimum possible (determined by the size of the data).
Value	Resize to value specified (in megabytes).

Resize Left Boundary Larger {Max | Min | Value }

Resize an extended partition larger by specifying the change in position of the left boundary. This operation is for extended partitions only.

Parameter	Description
Max	Same as Resize Left Boundary Max. Make the partition as large as possible.
Min	Grow the size of the partition by the smallest amount possible (1 cylinder).

Parameter	Description
Value	Grow the size of the partition by the size specified (in megabytes).

Resize Left Boundary Smaller {Max | Min | Value }

Resize an extended partition larger by specifying the change in position of the left boundary. This operation is for extended partitions only. See also, "Resize Left Boundary."

Parameter	Description
Max	Same as Resize Left Boundary Min. Make the partition as small as possible.
Min	Partition's size will be decreased by the minimum amount possible (1 cylinder).
Value	Partition's size will be decreased by the amount specified (in megabytes).

Resize Smaller {Max | Min | Value} [/Clustersize={512 | 1 | 2 | 4 | 8 | 16 | 32 | 64 }]

Resize a partition smaller by specifying the incremental change in size. See also, "Resize."

Parameter	Description
Max	Same as Resize Min. The partition will be as small as possible.
Min	Partition's size will be decreased by the minimum amount possible (1 cylinder).
Value	Partition's size will be decrease by the amount specified (in megabytes).
/Clustersize	Sets the cluster size to the size specified during the resize operation. The cluster size must be valid for the partition size specified.

Resize Space After {Max | Min | Value} [/Clustersize={512 | 1 | 2 | 4 | 8 | 16 | 32 | 64 }]

Resize a partition by specifying the free space desired after the partition after the resize is completed. See also, "Resize."

Parameter	Description
Max	Resizes so that the space after the partition is as large as possible. The partition is as small as possible.
Min	Resizes so that the space after the partition is as small as possible. The partition is as large as possible.
Value	Sizes the partition such that the space after is the size of value (in megabytes).
/Clustersize	Sets the cluster size to the size specified during the resize operation. Cluster size must be valid for the partition size specified.

Resize Space Before {Max | Min | Value }

Resize an extended partition by specifying the free space desired before the partition after the resize is completed. This operation is for extended partitions only.

Parameter	Description
Max	Resizes so that the space before the partition is as large as possible. The partition is as small as possible.
Min	Resizes so that the space before the partition is as small as possible. The partition is as large as possible.
Value	Sizes the partition such that the space before is the size of value (in megabytes).

Resize Root {Value | Min | Max }

Change the number of entries in the root directory of a FAT partition.

Parameter	Description
Value	Must be a value between 512 - 1024. This will change the maximum number of root entries possible for this partition. The number actually set will be the closest number possible to the number specified.
Min	Sets the partition to having the smallest possible maximum number of root entries.
Max	Sets the maximum number of root entries for the partition to the largest number possible.

Select

You do not need to select a disk when selecting a partition by Letter or Volume. All the disks are searched in order until the specified Letter or Volume Name is encountered. This means that if two partitions have the same Volume Name, the first partition found is the one that is selected.

The Next and Previous commands are relative to a previous selection of the same type. You must already have a partition selected before using Select Partition Next or Select Partition Previous. Likewise, you must have an unallocated space selected before using Select Unallocated Next or Select Unallocated Previous.

Select Disk {Number}

Use to select a hard disk.

There are no parameters.

Select Partition [{Number} | {Letter} | {"Volume"} | First | Last | Next | Previous | Extended}]

Use to select the partition.

There are no parameters.

Select Unallocated | [{Number} | First | Last | Largest | Next | Previous | After Selected Partition | Before Selected Partition}]

Use to select unallocated space on the partition.

There are no parameters.

Select Merge Partition | {Next | Previous}

Select a partition to merge with an adjacent partition.

There are no parameters.

Select Destination | Disk {Number}

Use to select the destination hard disk.

There are no parameters.

Select Destination | Unallocated | {{Number} | First | Last | Largest | After Partition {Number} | Before Partition {Number} | Next | Previous}

Use to select the destination for unallocated space on the partition.

There are no parameters.

Set Active

Mark the selected partition as the active, or bootable, partition.

Set Default Bad Sector Test State { ON | OFF}

Set the bad sector testing ON or OFF for all partitions on the currently selected drive. The "/BadSectorTest" option overrides this setting.

Set Drive Read Only Mode { ON | OFF}

Set the read-only flag ON or OFF for all partitions on the currently selected drive. When set on for a drive, modifications to the partitions on that drive will not be allowed. Some changes to boot.ini files may be allowed if they exist on the read-only drive when add, delete or copy partition operations are done.

Set Ignore OS/2 EA Errors { ON | OFF }

Set NT 64K FAT Clusters { ON | OFF }

Set Force User Logoff { ON | OFF }

Set user logoff to on or off. When set to on, it will force all users to logoff the system just before the remainder of the script is played.

Show {Partitions | Preference | Destination}

Display a summary of information for the selected command.

There are no parameters.

Unhide

Unhide the currently selected partition. See also, “Unhide.”

There are no parameters.

Script Suggestions and Notes

Although it is not necessary, PowerQuest recommends that you check each of the partitions that will be modified at the beginning of the script. Because a script file will terminate as soon as an error occurs, checking each of the partitions first will keep the script from making any changes before it finds errors.

Partitions must start on cylinder boundaries. For example, if you specify 10 MB, the real value could be10.2 MB. The difference between the specified and actual values varies depending on the geometry of the drive.

When specifying an amount for one of the script options, the program will allow a margin of error of 1 cylinder above or below that amount (or a range of 2 cylinders centered on the amount specified). For example, if 10 MB were specified for a resize and a cylinder was .5 MB, that the operation would complete successfully if it could resize the partition to at least 9.5 MB. The actual range would be 9.5 to 10.5 MB. If it could not resize the partition within this range, it will return an error.

Under normal operation, if the script determines that it will not be able to reboot the machine after making the changes specified in the script, the script will terminate with an error. This condition will occur under OS/2 if the DOS.SYS file is not in the CONFIG.SYS (such as when you boot from the utility disks). You should include ALLOW MANUAL REBOOT as the first script statement in the script if performing a manual reboot from the keyboard is not a problem.

All commands must be contained within one line. They cannot start on one line and finish on the next. The maximum length of a script line is 180 characters, which should be sufficient for any command.

Use extreme caution when selecting a partition by its number. The select by number feature must be available to select free space or partitions that have no drive letter or label. The problem with selecting a partition by number is that the numbers can change throughout a script. If you select partition 2 and move it to the right, any free space that has been moved from the right to the left side of the partition will now become partition 2. (The partition moved will still be selected regardless of the number). Using the Select Partition Next and Previous commands are usually preferable to selecting free space by partition number. With most operations, the partition selected after an operation will be the partition operated on. For example, Resize and Move will always leave the partition operated on as the selected partition after the operation. On a Create command, the partition created will be selected after the command whether it is at the beginning or end of the free space it was created in. If there is any doubt as to which partition will be selected after an operation, you can use the TTY version of the program (without scripting) and perform the same operation on a test machine and observe which partition is selected after the operation. You can also use the Show Partitions command to show the current status of partitions.

Sample Scripts

Scenario 1: General Example

You have primary C:, D:, and E: drives. There is no free space on the disk. You want to take 10 MB from D: and add it to E:.

```
SCRIPT1.FIL
//Check the partitions to be operated on first
Select Partition D
Check
Select Partition E
Check
//Select the first partition I want to change
Select Partition D
//Shrink the partition by 10 megabytes
Resize Smaller 10
//Select the partition to add the 10 meg to
```

```
Select Partition E
//Move the partition as far as possible to the left so that
//the free space just created will be on the left edge (the
//end)
Move Left Max
//Take up all of the available space
Resize Larger Max
```

Scenario 2: General Example

You have one large C: partition on the drive. The drive is 1.2 GB in size. You have only 300 MB of data on the partition and would like to create logical drives D: and E:. The E: drive needs to be 300 MB, and the rest of the disk space is to be split between the C: and D: drives.

```
SCRIPT2.FIL
//Check the partition first
Select Partition C
Check
//Partition C is already selected so shrink it to 450 MB
Resize 450
//Since the C partition is still selected after the resize,
//we need to select the free space created
//after C.
Select Partition Next
//Create the extended partition to the default size, which
//will be all of the free space currently selected
Create /FS=EXTENDED
//The Extended partition is now selected, and we want to
//select the next free space in the extended partition.
Select Partition Next
//Create the partition that we need to be 300 MB first at the
//end of the free space that is currently selected. (What
//will be the E partition)
Create /FS=FAT /Label="DBFILES" /Size=300 /Position=END
//Select the rest of the free space within the extended
//partition. Since the last partition was
```

```
//created at the end of the free space, we need to move to
//the free space previous to the selected partition
Select Partition Previous
//Create the partition in the rest of the free space
Create /FS=FAT /Label="APPS"
```

Scenario 3: General Example

You have C:, D: and E: partitions on one physical drive. There is no free space on the disk.

The C: partition is a primary partition, and the D: and E: partitions are logical drives in an extended partition. You want to create an F: partition with 40 MB of free space that is in the C: partition. The F: partition will be a FAT partition with a volume label of “DATA,” Since the drive is fairly new, you would like to skip bad sector testing for all operations.

```
SCRIPT3.FIL
//Check all of the partitions first
Select Partition C
Check
Select Partition D
Check
Select Partition E
Check
//Since a partition on this drive had already been selected,
//we can set the default bad sector testing to off for this
//drive
Set Default Bad Sector Test State Off
//Select the C partition and shrink it by 40 MB
Select Partition C
Resize Smaller 40
//Select the extended partition and resize the left
//boundary to the right edge of the C partition (max),
//putting the free space within the extended partition. To
//Select an extended partition, the drive must first be
//selected, and then the partition.
Select Drive 1
Select Partition Extended
Resize Left Boundary Max
```

```
//Select the D partition and move it to the left, essentially
// flush against the Extended and C partitions, leaving the
// free space between the D and E partitions
Select Partition D
Move Left Max
//Select the E partition and move it as far as possible to
//the left, so that the free space will be at the end of E,
//within the extended partition
Select Partition E
Move Left Max
//The free space is now after E and the user
// can create an F partition (logical drive)
//Move to the free space after E
Select Partition Next
//Create the FAT partition called DATA with all defaults.
//This will use all of the size available in the free space.
Create /FS=FAT /Label="DATA"
```

Scenario 4: General Example

The user has a C partition which is a primary having 100 MB. The next partition is a hidden partition called WARP_OS which is FAT and a primary partition at 100 MB. There are also 2 logical drives, D and E in an extended partition which are each 70 MB.

The user wants to reduce both the C and Hidden partitions to 60 MB, add 40 MB to the D partition, and create an F partition (HPFS) with the remaining free space. The user also wants to convert the hidden partition from FAT to HPFS. The user also does not care whether the machine can reboot under program control or not.

```
SCRIPT4.FIL
//Inform that a manual boot is acceptable
Allow Manual Reboot
//Check all of the partitions first
Select Partition C
Check
//The hidden partition is selected by using the volume label
//in quotes
```

```
Select Partition "WARP_OS"
Check
Select Partition D
Check
Select Partition E
Check
//Select the C Partition and resize it to 60 MB
Select Partition C
Resize 60
//Select the hidden partition
Select Partition "WARP_OS"
//Move the partition flush against the C partition (since it
//was just resized) putting the newly created free space
//after the hidden partition
Move Left Max
//Resize the hidden partition to 60 MB
Resize 60
//Convert the partition from FAT to HPFS
Convert To HPFS
//Expand the extended partition so that the free space is now
//inside the expanded partition
Select Drive 1
Select Partition Extended
Resize Left Boundary Max
//Move the D partition flush against the hidden and extended
//partitions
Select Partition D
Move Left Max
//Add 40 MB to the D partition
Resize Larger 40
//Move the E partition next to the D partition
Select Partition E
Move Left Max
//The Free space is now available at the end of the extended
```

```
// partition so that the user can create an F logical drive.
// Select the free space.
Select Partition Next
//Create the HPFS partition.
Create /FS=HPFS
```

Scenario 6: Cluster Analyzer

Imagine you have a 4 GB drive. You have a 2 MB primary partition and a 1 GB primary FAT C: partition. You also have a hidden primary FAT partition that is 1 GB. You boot multiple operating systems, and the third partition holds another operating system. You also have an extended partition with logical drives that use up the rest of the drive space.

You would like to analyze the two FAT partitions to see if you can reduce the cluster waste.

```
// Show Cluster Waste for Partition 2
Select Drive 1
Select Partition 2
Cluster Analyzer /ShowClusterWaste
// Show Cluster Waste for Partition 3
Select Drive 1
Select Partition 3
Cluster Analyzer /ShowClusterWaste
```

You can now know you can reduce cluster waste, so you use the Cluster Analyzer to reduce waste again. You will set the third partition to the recommended size and the second partition to a cluster size of 8K.

```
// Set Partition 3 to Recommended Cluster Size
Select Drive 1
Select Partition 3
Cluster Analyzer /SetToRecommended
// Set Partition 2 to 8K Clusters
Select Drive 1
Select Partition 2
Cluster Analyzer /ClusterSize=8
```

Scenario 7: Copy

You just installed a new drive. You would like to copy the first three partitions on drive 1 to drive 2. Drive 2 is formatted and is unused space.

```
// Select Drive 1, Partition 1
Select Drive 1
Select Partition 1
// Select Copy Drive 2, Copy Partition 1
Select Copy Drive 2
Select Copy Partition 1
// Copy First Partition
Copy
// Select Drive 1, Partition 2
Select Partition 2
// Select Copy Drive 2, Copy Partition 2
Select Copy Partition 2
// Copy Second Partition
Copy
// Select Drive 1, Partition 3
Select Partition 3
// Select Copy Drive 2, Copy Partition 3
Select Copy Partition 3
// Copy Third Partition
Copy
```

Scenario 8: Info

You have a FAT C: partition that you would like to get some information about.

You do not know how big the partition is or much about it. You would like to know about the disk usage, the cluster waste, partition information, and file system information.

```
Select Drive 1
Select Partition 1
Info /Usage /Waste /Partition /FS
```

This could also be accomplished as follows.

```
Select Drive 1
Select Partition 1
Info /Usage
Info /Waste
Info /Partition
Info /FS
```