Paragon Hard Disk Manager 6.0
Getting Started Guide

Contents

1 Introduction

Hard Disk Manager is a fast, convenient and reliable solution of disk copying, upgrading and configuring needs. It provides a wide-range functionality in the field of backup on-disk information and managing disk layout structures.

Presently hard disks are the primary part of modern informational systems based on IBM PC family computers. Hard disks hold not only user's data but also applications and system files required for running operating system. Parameters and configuration of disk-based storage systems noticeably affect on overall system performance and reliability.

1.1 How to Use this Guide

Use this Getting Started Guide to obtain the information you need to get Hard Disk Manager installed, configured and working correctly.

1.2 Contacting Paragon Technology GMBH

If you have questions about Hard Disk Manager please contact Paragon Technology GMBH.

To do this	Contact
Visit Paragon GMBH web site	www.paragon-gmbh.com
E-Service System registration & updates	www.penreader.com/paragon/site
Knowledge Base & Technical Support	kb.paragon.ag
Get pre-sale information	sales@paragon.ag
Pre-sale technical information	info@paragon.ag
Contact Technical Support	support@paragon.ag

1.3 Hard Disk Manager 6.0 key features

(New)
Friendly user interface for Windows, DOS and Linux platforms

Now Hard Disk Manager works in DOS, Windows and Linux environments. All versions have almost identical functionality and similar interface layout.

(New)
Virtual pre-execution of operations: what you see is what you *WILL* get

Hard Disk Manager allows previewing the resulting layout of hard disks before actually executing operations(so-called *virtual operations*). The special predicting module forecasts the post-operation state of the hard disks. You are able to execute multiple virtual operations and then evaluate the future state of hard disks. If needed, you can undo one or more virtual operations.

Making backup images of integral filesystem state

Hard Disk Manager makes the "momentary photography" of the partition. The backup image includes contents of all user-made files, the exact structure of directories, information about location of files on the disk, all file attributes and related information (e.g. security information).

(New)
Selectable restoration of partitions from backup archives

Another new feature included in the Hard Disk Manager is the ability to restore only selected partitions from multi-partition archives (such as images of entire hard disks or images of the Extended Partition).

(New)
Built-in tool for exploring contents of backup archives (Image Explorer)

Hard Disk Manager includes the Image Explorer tool that allows exploring contents of backup images and extracting selected files or directories without performing the restoration of a partition.

(New)
Built-in CD burning tool

The Windows and Linux based versions of the Hard Disk Manager provide the ability to burn backup images on recordable CD and DVD media directly from the program's interface, without using additional CD burning software.

(New)
Built-in filesystem drivers for NTFS, Ext2 and Ext3 file systems

All versions of Hard Disk Manager can read and write backup images on NTFS, Ext2 and Ext3 file systems without using the file service of the operating system. This unique feature allows placing backup images or separate volumes of multivolumic archives on any partition.

Fast processing algorithms for popular filesystems

Hard Disk Manager uses fast algorithms for copying, moving and backup partitions for FAT16, FAT32, NTFS, Ext2, Ext3, ReiserFS and HPFS filesystems (named *known filesystems*). Hard Disk Manager uses the knowledge of internal structure of these filesystems and manages only usable sectors of the partitions.

Supporting batch operations in unattended mode: macro language

With Hard Disk Manager, the macro language provides the ability to construct very flexible and smart scripts for automated working. The embedded macro language includes all the functionality available from the program interface.

Basic functions for initializing, partitioning and formatting hard disks

Traditionally, Hard Disk Manager provides basic disk partitioning functions such as creating, deleting and formatting partitions. Instead of the standard disk tools in Windows and Linux, Hard Disk Manager supports all filesystems and allows automating the process.

Non-destructive changing parameters of partitions

Hard Disk Manager includes functions of changing the size of the partition or changing the cluster size without re-formatting the partition. These features are very comfortable in case of migrating the system on more capacious hard disks.

Non-destructive conversion of FAT and NTFS filesystems

Hard Disk Manager provides the unique ability to convert FAT16, FAT32 and NTFS filesystems to each other without re-formatting the partition. This feature can be useful for adjusting the system for better performance and/or more abilities.

(New)
Built-in tool for checking filesystem integrity

With this feature, you are able to check the filesystem integrity on FAT16, FAT32, NTFS, Ext2, Ext3, ReiserFS and HPFS partitions from the program interface to validate if the partition is capable for modifications: the thing is that almost all advanced operations such as resizing, conversion or changing cluster size, can be executed only on valid partitions.

(New)
Undelete function for FAT16, FAT32, NTFS, Ext2 and Ext3 partitions

Hard Disk Manager allows to restore back "recently deleted" partitions that are not overlapped by other existing partitions. This feature may be very useful in case of occasional deletion of a partition.

(New)
Incremental Backup function

Hard Disk Manager allows adding only changes in partition's contents to a backup image. Alternatively, the program provides the ability of selecting the backup session, which should be actually restored.

2 Installing Hard Disk Manager

The setup package can be either the CD installation package or the downloadable one. The installation CD contains setup files and the Paragon Recovery CD. Paragon Recovery CD is the Linux-based bootable CD with Hard Disk Manager for Linux platform installed and configured. It can be used:

- as the fully functional disk management tool based on the bootable CD.
- as the autonomous recovery tool in case of severe malfunction of on-disk software.

2.1 Minimum System Requirements

To install and use the Hard Disk Manager on your computer, make sure your system meets the following minimum system requirements:

- IBM AT compatible computer with i486 or higher CPU
- A 32-bit version of Microsoft Windows: Windows 95, 98, ME, NT, 2000 or XP.
  Windows is required to install Hard Disk Manager and Diskette Build Wizard.
- 16 Mb of RAM (see Comments)
- 12 Mb of free disk space
- VGA-compatible monitor
- Mouse (recommended)
- CD-ROM drive.
  CD-ROM drive is required to install Hard Disk Manager from CD.

Additional requirements to the system for using the Paragon Recovery bootable CD:

- ATAPI compatible CD-ROM drive.
- On-board BIOS should support the ability "Boot from CD".

2.1.1 Comments

2.1.1.1 Memory requirements

Most advanced functions (changing partition size, converting filesystem type etc) require large amounts of memory. The actual numbers fundamentally depend on data being modified and cannot be declared a priori. This guide includes estimation formulas for most memory spending operations.

2.1.1.2 CD/DVD recordable drives

Hard Disk Manager includes the built-in CD/DVD burning module that is available in the Windows-based and Linux-based versions. To use this module, your system should meet additional requirements:

- CD-R(W) or DVD-R, DVD+R(W) recordable drive that is available through WinASPI interface. In particular, Windows Device Manager must display this CD/DVD drive in the list of available hardware.
- 70Mb of RAM is required; the additional memory is used for the burning cache.

2.2 Package Contents

Hard Disk Manager installation package contains following contents:

- Hard Disk Manager 6.0 for Windows 95, 98 and ME
- Hard Disk Manager 6.0 for Windows NT, 2000 and XP
- Hard Disk Manager 6.0 for DOS platform
- Hard Disk Manager 6.0 for Linux platform
- Diskette Build Wizard utility
- Paragon Partition Explorer

- Paragon Image Explorer

2.3 Installing Paragon Hard Disk Manager

2.3.1 Downloadable installation package

The downloadable installation package contains the single executable file, which is the self-extracting Windows-based application. You can get this file in following cases:

- You have purchased Hard Disk Manager over the Internet and have downloaded the setup file.
- You have downloaded an update/upgrade package of Hard Disk Manager from the Paragon E-Service System.

The following steps walk you through the installation process:

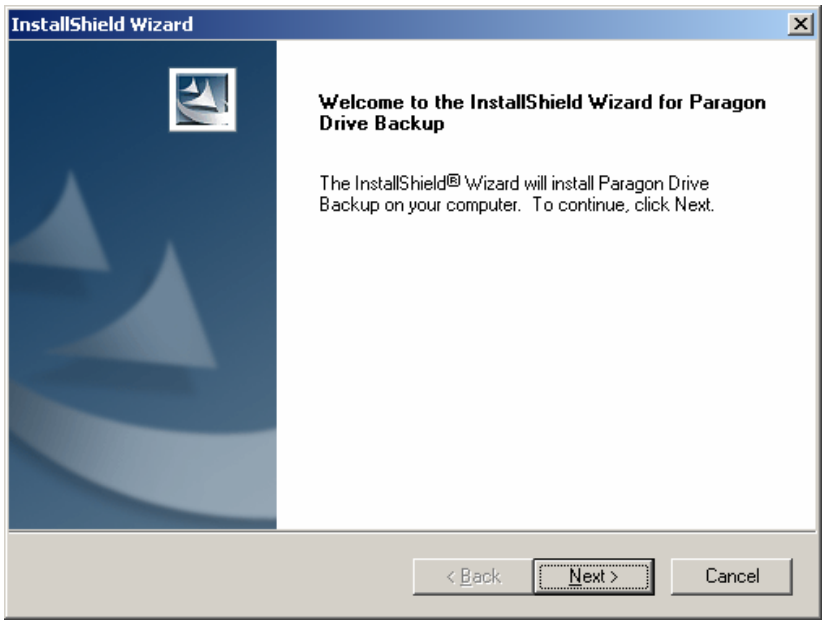
Step 1. Unzip setup files

Run the downloaded executable file and unzip the contents of the self-extracting archive to some folder. By default the utility places setup files to the folder "**\Paragon Hard Disk Manager 6.0**" on the current logical drive.

Step 2. Run Setup application

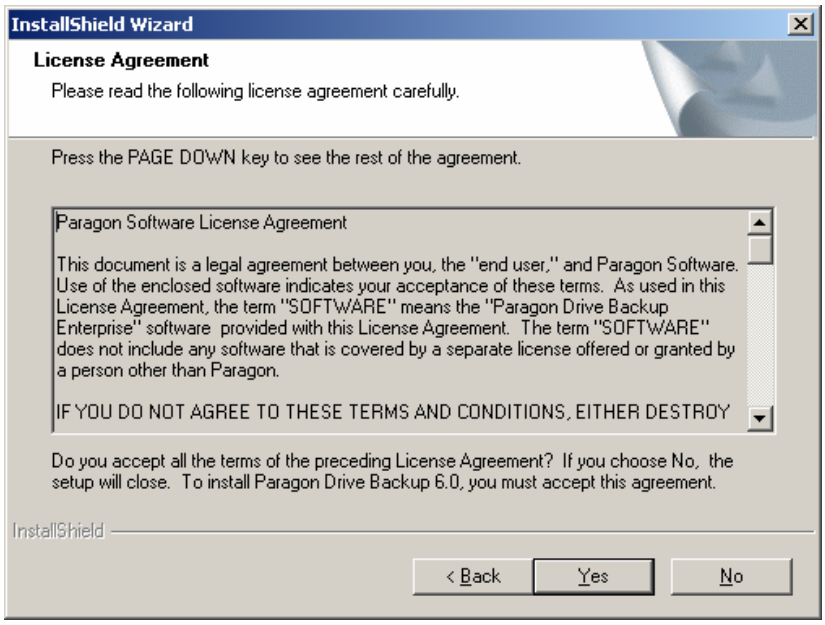
Go to the directory where setup files were placed and run the **SETUP.EXE** file. This application will walk you through the procedure of the complete program installation. The setup utility is made with using InstallShield SDK. It contains the standard user interface and the standard set of installation screens.

Step 3. Starting Setup



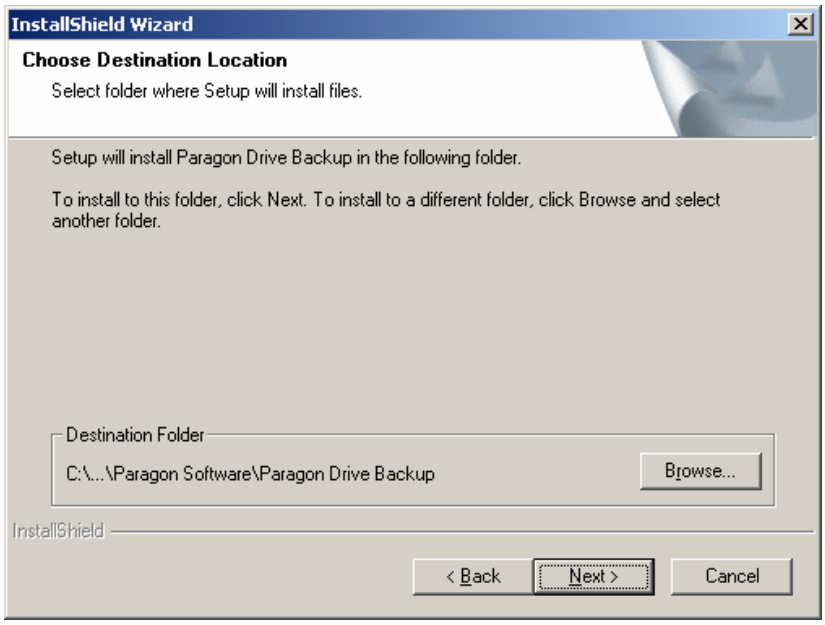
The **Welcome Screen** informs you which application is being installed. Just press **Next** button to move further.

Step 4. Confirm Licence Agreement



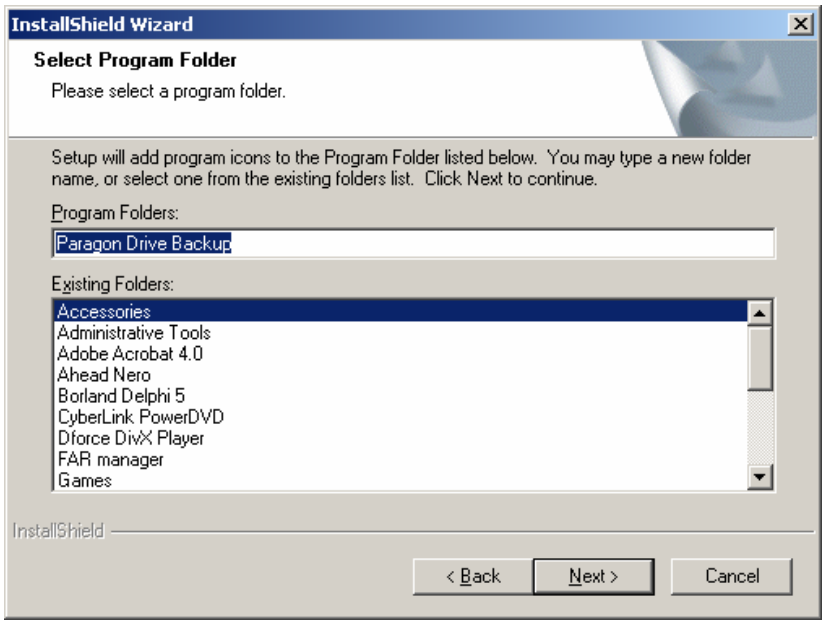
The **License Agreement Screen** displays the Paragon License Agreement. Read the Agreement and then press **Yes** button to accept the Agreement and continue the installation process.

Step 5. Choose an Installation Folder



The **Destination Location Screen** allows choosing the folder where Hard Disk Manager will be installed. Press **Browse** button to customize the name of the installation folder. Press **Next** button to apply selected name. The default value for the installation folder is:
C:\Program Files\Paragon Software\Paragon Hard Disk Manager 6.0

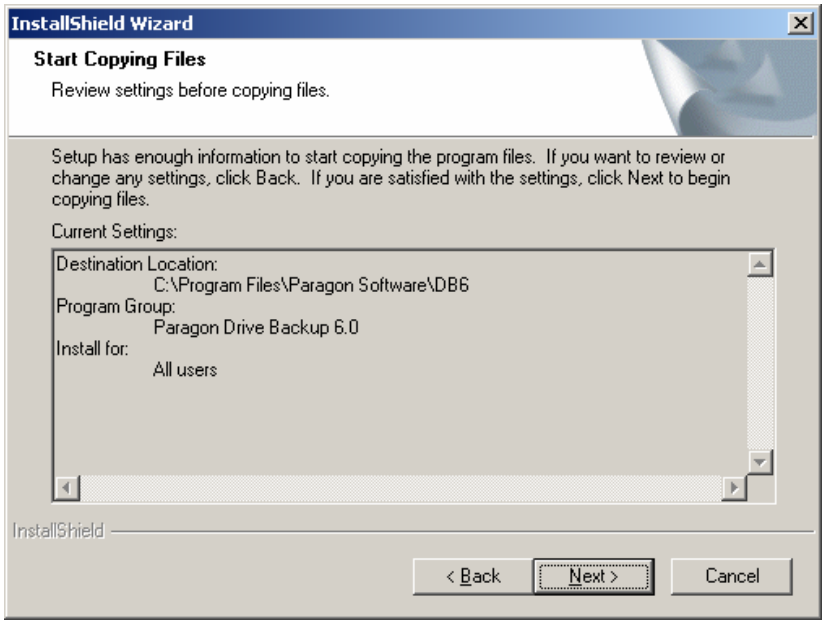
Step 6. Choose a Program Group



The **Program Folder Screen** allows selecting the application's program group in the **Start** Menu. By default, it will be the program group:

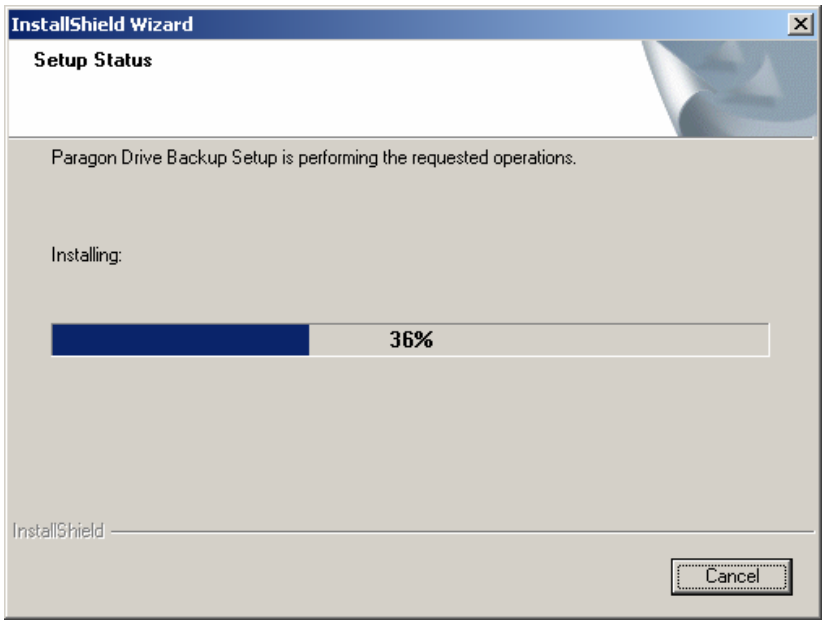
Start ➔**Programs** ➔**Paragon** Hard Disk Manager **6.0**

Step 7. Verify Setup Settings



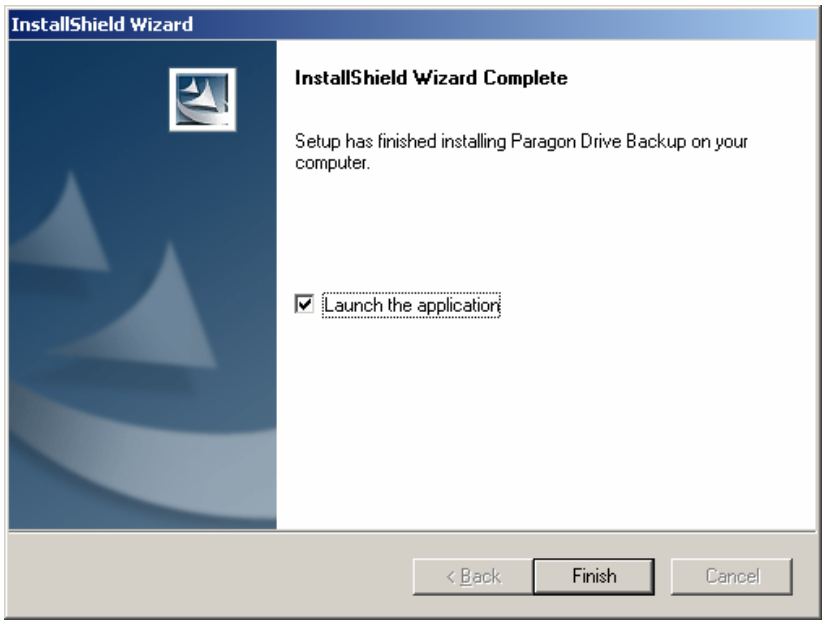
The **Start Copying Screen** allows verifying settings you've made before and probably make corrections. Press **Back** button to walk back and modify the installation settings. Press **Next** button to complete the installation process.

Step 8. Copying Files



The **Setup Status Screen** shows the overall progress of the installation. You are allowed to abort the process by pressing the **Cancel** button.

Step 9. Finishing the Installation



The **Final Screen** reports the setup process end. From this moment, Hard Disk Manager is ready to use.

2.3.2 Comments

2.3.2.1 Acceptable Installation locations

There are some fine points concerning *installation folder*:

1. Do not install Hard Disk Manager on network drives.
2. Do not use Terminal Server sessions to install and use Hard Disk Manager.

In both cases, there will be the limitation of the program functionality. In addition, it may lead to the incorrect working in particular situations.

2.3.2.2 Installing the program on the NTFS partition

Attention to owners of Windows NT/2000 Server and Advanced Server versions: if you are installing Hard Disk Manager on NTFS formatted partition, make sure that the generating of *short filenames* is enabled. Otherwise, the program may fail to run the *BlueScreen Component* that is responsible for processing *system* and *locked partitions* in Windows NT, 2000 and XP.

In such a situation it is recommended to reinstall Hard Disk Manager in the directory having the name that meets the *"8.3" filename format*.

By default, the *short filenames* generation is enabled, yet sometimes administrators disable it for improving NTFS performance.

The feature of generating short filenames for newly created files is controlled by the *Windows Registry key*:

```
NTFSDisable8dot3NameCreation
```

This key is duplicated within multiple registry keys, which describe hardware and software system configuration:

```
HKEY_LOCAL_MASHINE\SYSTEM\ControlSet001\Control\FileSystem
```

To enable *short filenames* generation:

1. run REGEDIT or REGEDT32 utility
2. find the `NTFSDisable8dot3NameCreation` key
3. set it to zero (0)
4. and then reboot Windows.

2.3.2.3 User privileges

There are some fine points concerning *user account* and *privileges*:

- The user should have enough privileges to modify the Windows system partition and modify the Windows Registry, exactly, registry branches:

  ```
  HKEY_LOCAL_MACHINE\SYSTEM
  HKEY_LOCAL_MACHINE\SOFTWARE
  ```

- Roaming user accounts are not allowed. The user must have the local account.

2.3.2.4 No concluding system reboot is required

Components included in the Hard Disk Manager installation package are ready for use immediately after completing the installation process.

2.4 Registering and Updating Hard Disk Manager

Paragon Software GmbH provides a wide array of online service through the Paragon Electronic Service System.

2.4.1 Paragon E-Service System

The Paragon Electronic Service System (hereafter ESS) provides following services:

- Registering new users
- Registering purchased products for registered users
- Providing registered users with the around-the-clock available downloading center. Registered users are able to download free updates and upgrades of purchased products, any language versions available, and the documentation.
- Providing all users with free demo versions and open documentation.
- Providing the online Knowledge Base of the Technical Support Team.

To enter the ESS, visit the web site mentioned in the chapter Contacting Paragon Technology GMBH.

It is recommended to use Internet Explorer 5+ or any compatible browser.

Generally, it is supposed the following scheme:

1. The user should register in the ESS. At this moment, the user need not be an owner of any Paragon product.

2. After purchasing some Paragon product (say, Paragon Hard Disk Manager 6.0), the user should login the ESS and register the product. From this moment, he is able to download commercial updates and upgrades of the product.

2.4.2 Registering in the E-Service System

1. Run the Internet browser and visit the E-Service System page.
2. Click on the "Registration" menu item. Then follow on-screen instructions:
   - ⇨ On the 1st screen, choose the country.
   - ⇨ On the 2nd screen, fill in the form.

The most important field is the *registered e-mail address*. First, ESS will send you the password to the registered e-mail address. Second, the registered e-mail address is used as the login to the ESS.

2.4.3 Registering new products in the E-Service System

1. Run the Internet browser and visit the E-Service System page
2. Click on the "Login" menu item.
3. On the Login screen, enter the registered e-mail to the **User Name** field. Also, enter the received password to the **Password** field.

After submitting the form, you will enter the E-Service System.

4. Click on the **Product Reg.** menu item. The list of the registered products should appear followed by the new product registration form.
5. Select the general name of the product in the list **Base Product**.
   For example, to register Paragon Hard Disk Manager 6.0, select the item "Paragon Hard Disk Manager".
6. The screen should update.
7. Select the particular version of the product in the list **Product**.
   For example, for Paragon Hard Disk Manager 6.0, select the item "Paragon Hard Disk Manager 6.0".
8. Enter the serial number of your product to the **Serial Number** field.
9. Press the button **Submit**.
10. ESS will verify your registration. On success, it will save your information in the database and enable using the downloading center for you.

2.4.4 Downloading updates

1. Enter E-Service System page.
2. Click on the item **Download Update** to get access to commercial downloads of your products.
3. Select the desired update and click the button **Download**

On the **Download page**, you can see the list of registered products on the top of the page. Below the list of products, you can see the list of available commercial updates (only for products you have registered). These updates are free of charge for the registered users. Each item is provided with brief information about new features, the size of the downloadable file and the date.

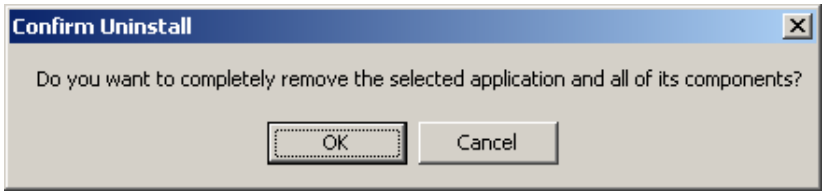
2.4.5 Updating and Upgrading registered products

Usually, *updates* and *upgrades* are the fully functional installation packages of proper products. To apply an update, you should uninstall the previously installed version of the program and then install the update/upgrade.

2.5 Uninstalling Hard Disk Manager

To uninstall Hard Disk Manager, select the shortcut in the Start menu:

Start ➔ **Programs** ➔ **Paragon** Hard Disk Manager **6.0** ➔ **Uninstall** Hard Disk Manager



Confirm the deletion of the Hard Disk Manager and all of its components. No reboot is required to complete the uninstallation process.

3 Hard Disk Manager Functionality

3.1 Functionality overview

This chapter introduces the overview of the Hard Disk Manager functionality and interface, basic concepts, key features and terms.

3.1.1 Known filesystems

Hard Disk Manager uses the knowledge of the internal structure of FAT-12/16/32, NTFS, Ext2, Ext3, ReiserFS, HPFS, L-Swap 1&2 filesystems.

The program provides the advanced functionality only for partitions that are formatted to the *known filesystems*: fast copying, resizing, changing the cluster size, filesystem conversion and so on. The basic functionality (copy, move, backup and restore) is ***available for partitions of all types***, even "unknown".

Operation	FAT, NTFS	Ext2, Ext3	ReiserFS	HPFS	L-Swap 1&2	Other filesystems
Backup & Restore	YES	YES	YES	YES	*re-create	** in 1:1 mode
Copy	YES	YES	YES	YES	*re-create	** in 1:1 mode
Format	YES	YES	YES	YES	YES	NO
Resize & Move	YES	YES	YES	NO	*re-create	NO
Change Cluster Size	YES	YES	NO	NO	NO	NO
Convert	YES	NO	NO	NO	NO	NO

* Hard Disk Manager does not save data located on the Linux Swap partitions. The program just deletes existing L-Swap partitions and makes the new ones.

** The partitions of unknown types are always processed by using the sector-to-sector algorithm (1:1 mode)

3.1.2 Fast copying algorithm

The basic functions of the Hard Disk Manager are:

- Backup partition & Backup hard disk
- Restore partition & Restore hard disk
- Copy partition & Copy hard disk
- Move partition

These functions can be executed in the two different modes:

1. The *fast copying mode*
2. The *sector-to-sector mode* (also named *1:1 mode*)

3.1.2.1 Fast copying mode

In the *fast copying mode,* the program uses the knowledge of the filesystem structure to detect, which sectors of a partition are not used by files or metadata. The program optimizes read-write operations to skip unused sectors. This technique allows accelerating significantly the performance of basic operations.

There are restrictions of the fast copying mode:

1. It is applicable only for Known filesystems.
2. It is inapplicable for corrupted filesystems.

Every operation begins with the filesystem integrity check. In case of filesystem corruption, the program shows the error message "**Incorrect filesystem**" and cancels the operation. In this case, you should fix the filesystem integrity with using the system tools (e.g. run **SCANDISK** in Windows 98, **CHKDSK /F** in Windows 2000 or **e2fsck** in Linux).

3.1.2.2 Sector-to-sector copying mode

In the *sector-to-sector mode*, the program simply operates all sectors of a partition. The program does not use the map of usable sectors so that it accepts partitions of any type.

Fast copying Mode	Sector-to-sector Mode
Advantages	
• The program copies only usable sectors • Require less time to complete the operation (Copy, Move, Backup) • Backup images are shorter	• Applicable for all filesystems, even unknown • Applicable for all partitions including corrupted ones. The only way to backup a corrupted partition is to switch the program in the sector-to-sector mode
Disadvantages	
• Inapplicable for corrupted filesystems. In particular, you're unable to backup corrupted partitions. • Inapplicable for unknown filesystems	• The program copies all sectors, even unused • Require more time to complete the operation • Large Backup images

By default, Hard Disk Manager works in the "smart mode": the program automatically switches to the fast copying mode to operate partitions of *known types*. When processing unknown filesystems, the program automatically switches to the sector-to-sector mode.

Hard Disk Manager can be forced to work in the sector-to-sector mode for all partitions (see the section Settings overview➔Copy all sectors 1:1):

(menu) General ➔ Settings… ➔ (tab) General ➔ Copy all sectors 1:1

3.1.3 Virtual operations

Hard Disk Manager supports two different operational modes: the *Immediate Execution mode* and the *Virtual Execution mode*.

3.1.3.1 Immediate Execution mode

In the *Immediate Execution mode*, Hard Disk Manager performs each operation immediately after a user enters operation parameters (in the same fashion as it was made in old versions of Hard Disk Manager).

3.1.3.2 Virtual Execution mode

In the *Virtual Execution mode*, Hard Disk Manager emulates the post-operation configuration of hard disks; the program allows to *preview* the expected layout (i.e. the *virtual state*) of hard disks before actually executing operations.

The program does not execute operations immediately, but places them in the *List of Pending Operations*. The special predicting module forecasts the post-operation state of hard disks. A user can execute multiple virtual operations and then evaluate the future state of hard disks. If needed, a user can *undo* one or more virtual operations. To really execute pending operations, a user should *apply pending operations* by pressing the **Apply** button.

The advantage of the *Virtual Execution mode* is that a user is able to execute quickly a set of virtual operations, and then Hard Disk Manager will really complete them in unattended mode.

3.1.3.3 Smart mode for virtual operations

Additionally, Hard Disk Manager supports the mixed execution mode named "smart mode". In this mode, Hard Disk Manager virtually executes all lengthy operations. Quick operations are executed in the following manner:

- If some virtual operations are already accumulated, a quick operation will be executed virtually, too: it will be placed in the *List of Pending Operations* for future execution.
- If no virtual operations are scheduled, a quick operation is executed immediately.

Quick operations are:

- ⇨ Hide/Unhide partition
- ⇨ Set partition active/inactive
- ⇨ Mount partition
- ⇨ Set partition Label

3.1.4 Incremental Backup functionality extension

Common backup tools can only backup all contents of a partition. In case of making multiple backup archives of a partition, unchanged data are duplicated in all archives and take redundant space on backup media.

Hard Disk Manager provides the ability of archiving only changes in the partition's contents. This functionality extension is named *Incremental Backup*.

The Incremental Backup functionality extension can be very useful for PC users who need periodically to backup the same working partition having most of its contents unchangeable, in a compact form. For example, backup operators can make a "history" of important data, beta-testers and QA engineers can save multiple evolutionary states of a system in order to examine a behavior of tested software.

To perform the Incremental Backup of a partition, Hard Disk Manager requires using a previously made archive of this partition in order to distinguish changes in the partition's contents. The backup image (i.e. archive) that is used as a "previous partition state" is named the *parental image* or the *base image* of a partition. The program produces the exact bitwise comparison of the previous partition's data (that are saved in the parental image) with the current ones (that is actually the partition itself). The difference in contents is saved in the newly created incremental backup archive.

Hard Disk Manager allows using a previously made incremental image as a parental image for a new incremental archive. So that one can build a set of incremental images that save the evolution of on-partition data. The most first image in the chain of incremental images should be an ordinary one.

An incremental backup archive cannot be used detached from its parental image. When restoring a partition from an incremental archive, the program at first finds and restores the parental archive, and then implements changes saved in the incremental archive.

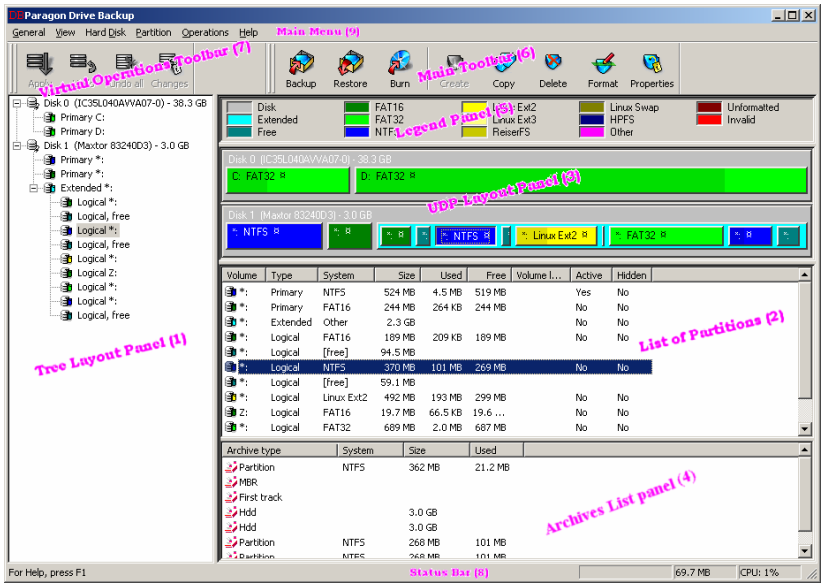
The creation of an incremental archive usually takes more time than creation of an ordinary one. However, an incremental archive usually takes *much* less space.

The Incremental Backup functionality extension is currently enabled only for single Primary and Logical partitions.

3.2 Interface overview

3.2.1 Interface Layout

The main window of Hard Disk Manager is conditionally subdivided in several parts that differ in their purpose and functionality:



1. Tree Layout panel
2. List of Partitions
3. UDP layout panel
4. Archives List panel
5. Legend panel
6. Main Toolbar
7. Virtual Operations Toolbar
8. Status Bar
9. Main Menu

Some of the panels nearly duplicate the functionality, and they have the synchronized layout. The program allows hiding some panels to simplify the interface management.

All panels are separated from each other with vertical and horizontal expandable sliders, so that a user can customize the screen layout.

3.2.1.1 Tree Layout panel

The panel represents the tree-like hierarchic list of hard disks and partitions.

Hard disks are represented with the top-level expandable nodes that contain the *disk number*, the OEM model name and the overall disk capacity. Disks are sorted by their logical numbers in the system.

Extended Partitions are also represented with expandable nodes. The sub-tree of each Extended Partition contains the list of logical partitions.

Nodes that represent *primary partitions* and *logical partitions* contain some information about partitions: the *type* of a partition (Primary, Extended or Logical), the *drive letter* assigned (if exists) and the colored representation of the *filesystem type*.

In addition, *blocks of free space* are represented with simple nodes, which include information about the type and size of a free block.

You can call the context-sensitive popup menu for every type of nodes:
- the popup menu for *Disk* nodes is equal to the **Hard Disk** menu
- the popup menu for *Partition* nodes is equal to the **Partition** menu
- the popup menu for *Free* blocks is equal to the **Partition** menu

The *Tree Layout panel* is synchronized with the List of Partitions and the UDP layout panel.

3.2.1.2 List of Partitions

The *List of Partitions* represents only partitions within the selected hard disk. The panel displays the extended information about the partition: *drive letter* (if exists), partition *type* (Primary/Extended /Logical), *filesystem* type, *size*, amount of *used* and unused (*free*) space, *volume label* and flags "Active" and "Hidden".

The list of partitions is sorted by the starting position of partitions in ascending order.

You can call the context-sensitive popup menu for every item in the list.

The *List of Partitions* is synchronized with the Tree Layout panel and the UDP layout panel.

3.2.1.3 UDP Layout panel

The *UDP Layout panel* displays the layout of hard disks with using the UDP controls. UDP controls represent in convenient fashion the relative size of on-disk partitions.

Also, UDP layout panel can be used for visual managing partitions with using *drag-&-drop* technique, in case of Hard Disk Manager works in the Virtual execution mode.

The activity of UDP controls is explained in the section UDP control activity. Briefly, you can virtually execute *Copying*, *Moving* and *Resizing* of partitions in the UDP Layout panel.

The *UDP Layout panel* is synchronized with the Tree Layout panel and the List of Partitions.

3.2.1.4 Archives List panel

The *Archives List panel* is designed for the purpose of simplification the navigation between backup images. It holds brief information about recently used and created images: a type of saved objects (disk/partition/MBR/1st track), location, filesystem type, partition size and amount of used space.

One can activate the context-sensitive popup menu for the list contents. The popup menu provides the functionality of browsing and validating image contents, modifying the list of images and restoration of image contents (see Restore Partition and Restore Hard Disk chapters).

The database of recently used images is kept in the file ARCHIVES.INI, which is located in the same directory with the Hard Disk Manager's executable file; by default, it is located in the folder:

```
C:\Program Files\Paragon Software\Hard Disk Manager 6.0\WinDB
```

3.2.1.5 Legend panel

The *Legend panel* describes the color indication of the filesystem types being used in Hard Disk Manager interface.

Hard Disk Manager distinguishes following types of partitions:

- Known filesystems (FAT12/FAT16, FAT32, NTFS, Ext2, Ext3, ReiserFS, HPFS, Linux Swap-1&2).
- Extended Partition (exactly, the space that is reserved in the Extended Partition for allocating logical partitions).
- Free space (the *unpartitioned* disk space that does not belong to any partitions). Within the Extended Partition, the free space is one that does not belong to any logical partition.
- *Invalid* partitions. This category includes only partitions with invalid parameters and ones having *corrupted filesystems* of *known filesystem types*.
- *Other* partitions (i.e. partitions that have *unknown* filesystem types). In addition, Hard Disk Manager marks as *other*, partitions that are incompletely modified by the program (e.g. some operation was abnormally interrupted).
- *Unformatted* partitions. The category includes just created partitions. The program can confuse *wiped* and severe damaged partitions with *unformatted* ones.

3.2.1.6 Main Toolbar

The *Main Toolbar* provides the fast access to most frequent partitioning operations: backup/burn image/restore for partitions & disks and create/delete/format/copy for partitions.

3.2.1.7 Virtual Operations Toolbar

The *Virtual Operations Toolbar* provides the fast access to the functions of manipulating with the *List of Pending Operations* (see Virtual operations).

Available operations:

Apply	Execute all accumulated pending operations (the List of Pending Operations will be emptied).
Undo	Undo the last pending operation in the list.
Undo All	Cancel the entire list of pending operations.
Changes	Display the advanced dialog of managing virtual operations.

3.2.1.8 Status Bar

The *Status Bar* displays additional information:

- In the left corner, the menu hints (brief descriptions) are displayed
- In the right corner, the current *CPU usage* and the *memory used* values are displayed.

3.2.1.9 Main Menu

The *Main Menu* provides the access to the entire functionality of the program. Available functions are the following:

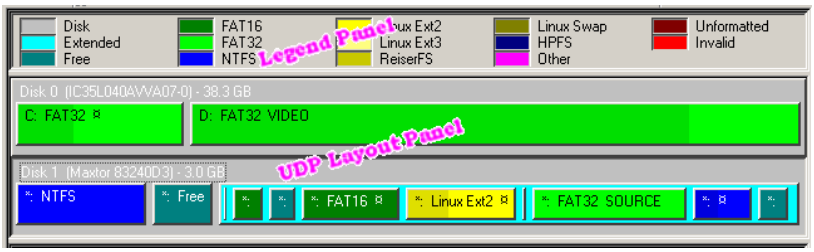
Menu Item	Functionality
General	**(settings and getting information)**
Show archive info	Explore the structure of partition(s) stored in the backup image with the ability of browsing their contents
Show CD/DVD Burners	Display the list of CD/DVD burning drives
Settings	Change program settings
Exit	Exit the program
View	**(controlling interface layout)**
Toolbar	**(controlling the view of toolbars)**
Main toolbar	Show/hide Main toolbar
Virtual operations toolbar	Show/hide Virtual operations toolbar
Large buttons	Switch between large and small buttons in all toolbars
Text labels	Show/hide text labels under buttons in all toolbars
Status Bar	Show/hide Status Bar
Disk Map	**(UDP Layout settings)**
Size	Select the width of UDP controls (three alternatives)
Proportional view	Follow proportions of hard disks capacities in the UDP Layout panel. By default, the program ignores the difference in size of hard disks.
Legend	Show/hide the Legend panel that displays the color indication of filesystem types
Configuration tree	Show/hide the Tree Layout panel
Recent archives list	Show/hide the Archives List panel
Hard Disk	**(hard disk based operations)**
Copy hard disk	Copy all contents of entire disk (Track#0+all partitions) with the ability of proportional resizing of all partitions.
Create an image of hard disk	Make an image of all contents of entire disk
Burn an image of hard disk to CD	Place the disk image to recordable CD/DVD discs

Restore hard disk from image	Restore disk from image with keeping the disk layout and the ability of proportional resizing of all partitions at once.
Selective partition restore on hdd	Restore chosen partitions from the hard disk image with the ability of independent resizing.
Wipe hard disk	(Multi-pass) wiping entire hard disk.
Change primary slot	Re-order MBR records that refer to primary partitions (controlling of DOS and Windows boot-up behavior)
View sectors	Explore & edit sectors of the disk
Browse disk	Browse disk contents with the Paragon Partition Explorer (OS-independent exploring of FATxx, NTFS, Ext2/3)
SID Changer	Run the built-in SID Changer on all primary partitions of the hard disk chosen.
Properties	Show properties of hard disk
Partition	**(partitions based operations)**
Copy partition	Copy a single partition with the ability of resizing.
Create an image of partition	Backup a single partition
Burn an image of partition to CD	Place the partition image to recordable CD/DVD discs
Restore partition from image	Restore the partition from the image with the ability of partition resizing.
Selective partition restore	Restore separate partition(s) from a multi-partition image with the ability of independent resizing.
Create	Create a new partition (primary, Extended, logical)
Format	Format an existing partition to FAT/FAT32, NTFS, Ext2, Ext3, ReiserFS or Linux Swap with using built-in tools.
Delete	Delete an existing partition
Wipe partition	(Multi-pass) wipe contents of the single partition
Clear free space	Wipe unused sectors of the partition
Resize/Move	Reallocate the partition within the enveloping disk space
Mount	Assign/remove the drive letter assigned to the partition (available only in Windows NT, 2000, XP)
Hide/Unhide	Hide/unhide the partition. The property will affect the system working in next reboot.
Set active/Set inactive	Set/reset primary partition active (=bootable). The property will affect the system working in next reboot.
Modify	**(changing parameters of filesystem)**
Convert	Change the filesystem type without reformatting (available for FAT16, FAT32, NTFS)
Set Label	Change the volume label placed in the boot sector
Change Cluster size	Change the cluster size filesystem parameter without re-formatting the partition
Change Root size	Change the Root Directory capacity on a FAT16 partition
Change Boot size	Change the amount of sectors reserved for keeping the bootable code on FAT16 and FAT32 partitions
Change Serial Number	Change the Serial Number of the partition that is placed in the boot sector on FAT16, FAT32 and NTFS partitions
Change partition ID	Change the code of the filesystem type in the MBR/EPT
Make primary/Make logical	Exclude/include the partition to the Extended Partition. Available only for primary and logical partitions that are close to the left or the right edge of the Extended Partition.
SID Changer	Run the built-in SID Changer on the partition chosen.
View sectors	Explore & edit sectors of the partition
Retest surface	Perform the surface test on the existing partition or the block of free space.
Check filesystem integrity	Check filesystem integrity by built-in tools (available for FAT16, FAT32 and NTFS filesystems)
Undelete	Find and revive occasionally deleted partitions
Browse partition	Browse partition contents by using the built-in filesystem drivers (available even for unmounted partitions)
Properties	Display the detailed properties of the partition
Operations	**(managing virtual operations)**
View pending changes	Display the advanced manager of pending operations
Apply changes	Immediately execute all accumulated pending operations
Undo last operation	Undo the last operation in the List of Pending Operations
Undo all operations	Cancel all the List of Pending Operations
Reload	Re-read the current state of hard disks (available only in case of the List of Pending Operations is empty)
Help	**(help and troubleshooting)**
Contents and Index	Run program's Help
Troubleshooting	**(automated troubleshooting assistance)**
Send log files	Compress and send the log to the Paragon Support Team
About Hard Disk Manager	Display the *About* window

3.2.2 UDP controls

In terms of Hard Disk Manager project, *UDP* is the servicing abbreviation of the phrase *Unified Drive Presentation*. *UDP windowed controls* are designed for the purposes of better presenting and advanced controlling of disk layout.

UDP controls are widely used in the program's interface in various places, where it's required to display the layout of partitions within some range of disk space (or within entire hard disk). UDP controls are sensitive to user's actions and can be effectively used for the fast visual management of disk layout.



The main screen of Hard Disk Manager includes the special UDP Layout Panel that represents the layout of all available hard disks with the aim of UDP controls.

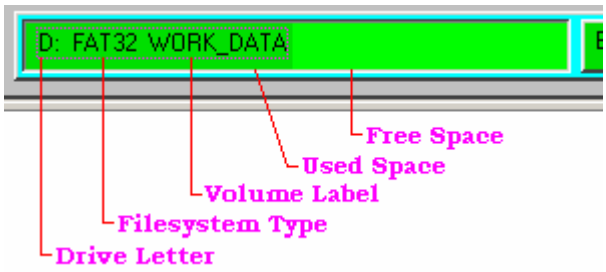
The Legend Panel that is located just above the UDP Layout Panel is aimed to explain the colored indication that is used in UDP controls.

3.2.2.1 Indication of partition parameters

The partition type is indicated by the color

The UDP control represents the layout of some range of the disk space. Partitions and blocks of free space are displayed by colored bars; the color indicates the partition type. Blocks of free space are marked with the *teal color* ( ), and the *Extended Partition* is marked with the *aqua color* ( ). The Extended Partition is only the container of so-called *logical partitions*, so that the bars that represent logical partitions are enclosed in the aqua-colored bar. The *silver color* (, light-gray) is reserved for the hard disk space indication. The Legend Panel explains the colored indication of partition types that is used in UDP controls.

The partition type, drive letter and the volume label is are displayed on the UDP control:



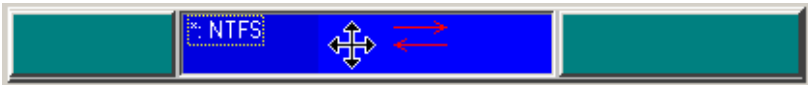
Each colored bar that corresponds to some partition, contains the indication of the following properties:

- The *drive letter* that is assigned to the partition.
  If the partition is not mounted, the **"*:"** text is displayed.
- The *filesystem type* (in textual form): FAT32, NTFS, Linux Ext2, etc.
- The *volume label* (the value taken from the boot sector).
- The percentage of *used* and *free space*: the part of partition space that corresponds to "used space" is marked slightly gray.

3.2.2.2 UDP control activity

The UDP controls are active on-screen elements. They react on user input and support the following functionality:

1. The UDP control responds to the clicking on the right mouse button. In this case, the context-sensitive partition menu will be displayed.
2. The UDP control allows to move the selected partition using *drag-&-drop* within a range of disk space that is combined from:
   - ⇨ the partition itself
   - ⇨ the right-hand adjacent block of free space
   - ⇨ the left-hand adjacent block of free space.



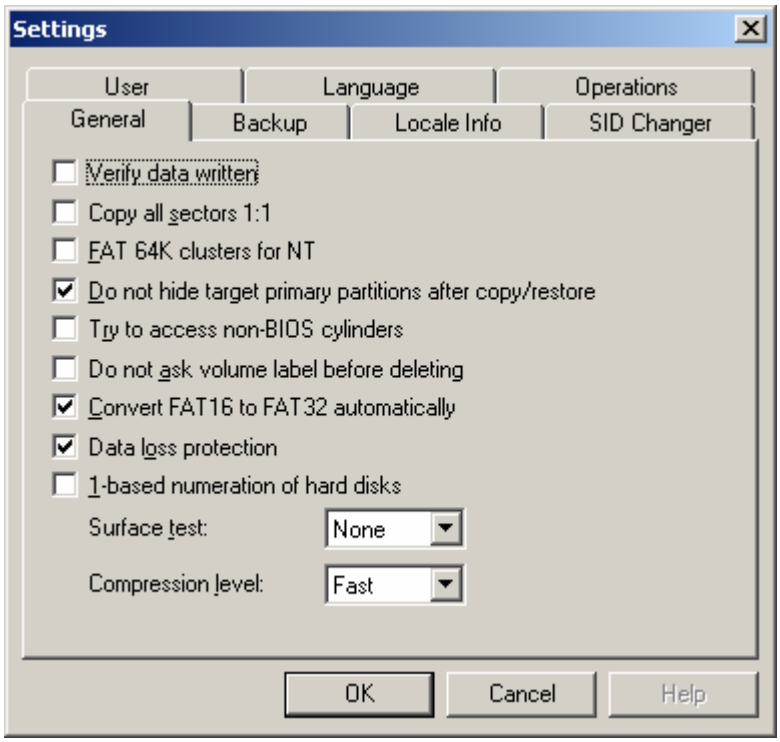
3. The UDP control allows to resize the selected partition using *drag-&-drop* within a range of disk space that is combined from:
   - ⇨ the partition itself
   - ⇨ the right-hand adjacent block of free space
   - ⇨ the left-hand adjacent block of free space.

4. The UDP control allows to copy the selected partition using *drag-&-drop*:
   - ⇨ Select the source partition, press and hold the left mouse button.
   - ⇨ Press *Ctrl* key (the small plus sign should appear in the right bottom corner of the mouse cursor).
   - ⇨ Drag the partition and drop it to the targeted location.

The UDP controls are used in the program interface for the purposes of displaying the arrangement of partitions and managing the location of partitions and their size.

3.3 Settings overview

3.3.1 General Page



3.3.1.1 Verify data written

The option forces the program to verify data being written: every *write-to-disk* action is followed by the *read-and-compare* action. The feature may be usable in case of unstable hard disk working. It may significantly slow the overall performance.

3.3.1.2 Copy all sectors 1:1

This option forces the program to work in the *sector-to-sector mode* (see Fast copying algorithm). The option affects on the following functions:

- Backup partition & Backup Hard Disk
- Copy partition & Copy hard disk
- Move partition

3.3.1.3 FAT 64K clusters for NT

The option allows to make FAT16 partitions having 64K cluster size. Hence, the program allows making FAT16 partitions up to 4Gb in size.

In fact, only Windows NT 4.0 really supports 64K clusters. Other operating systems do not support 64K clusters on filesystems on any type!

3.3.1.4 Do not hide target primary partitions after copy/restore

The option controls whether the program automatically sets the *Hidden* attribute for just restored or copied partitions. There is no the obviously preferred value for that option. See the explanation below to choose the most appropriate value for this option.

The fundamental feature of the *Restore partition* and *Copy partition* functions is that *they change the amount and probably the relative order of partitions*. The primary after-effect is the probable changing of drive letters assigned to partitions. From this point, the difference in Windows versions plays the main role:

- In Windows NT, 2000 and XP, one can fully control the drive letters assigned to any partition. These operating systems do not automatically change drive letters for already mounted partitions in case of adding new ones – the user should make changes manually.
- In Windows 95, 98 and ME, the operating system automatically assigns drive letters to detected partitions, according to some predefined rules. An unreasoned adding a new partition may lead to the mixing of drive letters after the next system startup, so that this may also lead to the non-operability of some software or even the operating system.

The way to avoid mixing of drive letters is to automatically *hide* new partitions.

The other side of the problem is that Windows (excepting Windows 2000 and XP) are unable working with hidden partitions, so that one must manually unhide partitions to work with them.

In case of the restoration of the system partition, the user must take care of un-hiding the just restored system partition, otherwise Windows will unable to start from the hidden system partition.

3.3.1.5 Try to access non-BIOS cylinders

The option forces Hard Disk Manager do not trust the hard disk capacity value that is returned by BIOS but detect the disk capacity by the non-standard procedure.

In fact, the option is required only for the compatibility with some old hardware. This feature is effective only in DOS and Windows 95, 98, ME. In Windows NT, 2000 and XP and in Linux the option takes no effect.

3.3.1.6 Do not ask volume label before deleting

Activate the option to suppress the acquiring the volume label of the partition being deleted (see the chapter Delete Partition).

By default, Hard Disk Manager asks the volume label before deleting the partition (just to eliminate the chance of occasional deletion).

3.3.1.7 Convert FAT16 to FAT32 automatically

Activate the option to suppress the warning about the conversion FAT16 filesystem to the FAT32 during the resizing partition. The option affects on the following operations:

- Resize Partition
- Copy Partition (with Autoresize)
- Copy Disk (with Autoresize)
- Restore Partition (with Autoexpand)

- Restore Disk (with Autoexpand)

The thing is that FAT16 maximum capacity is limited to approximately 2Gb (in case of 64K cluster the limit is 4Gb, see the option FAT 64K clusters for NT). For this reason, partitions that are greater than 2Gb cannot be correctly formatted to the FAT16 filesystem.

Hard Disk Manager suggests to convert FAT16 filesystem to the FAT32 in case of the resulting size of a partition exceeds the maximum size for the FAT16 filesystem. By default, the program warns a user about the filesystem conversion because some old operating systems do not support FAT32:

- Windows NT 4.0 and lower versions
- Windows 95 OSR1
- All DOS versions that precede the MS-DOS 7.1 (from Windows 95 OSR2).
- MS Windows 3.11 and lower versions.

3.3.1.8 Data loss protection

Activate the option to force Hard Disk Manager to work in the *fail-safe mode*. In this mode, the program keeps the special journal of an operation's progress.

In case of hardware malfunction, power break or operating system failure, the modified partition may become corrupted and non-operable. Still, Hard Disk Manager is able to accomplish the interrupted operation and thus "revive" the partition.

The journaling of the operation progress significantly slows the performance.

To use this feature, build the bootable diskette with the DOS-based version of Hard Disk Manager prior to running operations in the fail-safe mode. The alternative is using the Hard Disk Manager Bootable CD.

In the fail-safe mode, if the system crashes during the operation, insert the bootable diskette (or the Bootable CD) and start the computer from that media. Hard Disk Manager will automatically detect the journal of the interrupted operation and accomplish the job.

The current version of Hard Disk Manager supports the *fail-safe mode* for the following operations:

- Resize&Move Partition
- Convert Filesystem
- Change Cluster Size on FAT16, FAT32, Ext2 and Ext3 filesystems
  (not available in the *Change Cluster Size on NTFS*).
- Change Root Size and Change Boot Size (FAT16, FAT32 only)

3.3.1.9 1-based numeration of hard disks

Activate the option in order to use the 1-based enumeration of hard disks. By default, Hard Disk Manager uses the zero-based enumeration of hard disks and partitions.

3.3.1.10 Surface test

The option defines the default value for the *media surface test* suboperation. The available values are:

None	No surface test
Normal	Single-pass read test
Extreme	Three-pass read-&-write test

The option affects following operations:

- Format partition
- Copy partition
- Resize&Move partition
- Restore partition from an image
- Retest surface (this operation ignores the setting *Surface Test = NONE*)

When performing the surface test, Hard Disk Manager is able to detect bad sectors and mark them unusable.

3.3.1.11 Compression level

The option defines the default compression level. The option affects following operations:

- Backup partition & Backup hard disk

- Burn an image of a partition & Burn an image of a hard disk

Using the image compression significantly slows the overall operation performance.

Approximate values for the compression ratio and speed deceleration:

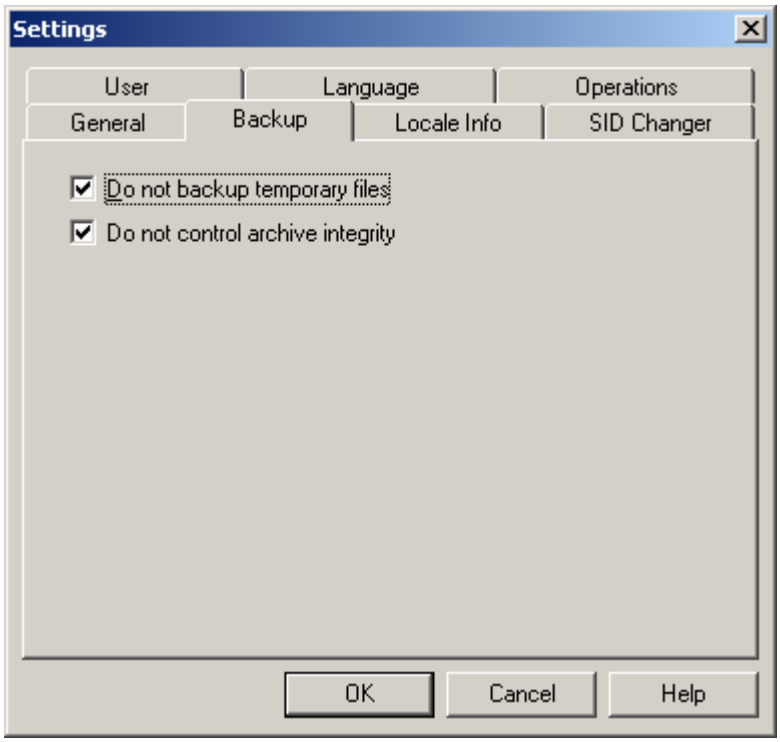
Level	Algorithm	Compression	Performance
None	(none)	no compression	no deceleration
Fast	RLE	95-80 % (1.05 – 1.25 times squeezing)	~90 % (~1.1 times slower)
Normal	LZW	65-70 % (1.4 – 1.5 times squeezing)	60-70% (1.4 – 1.7 times slower)
Best		40-50 % (2.0 – 2.5 times squeezing)	10-12% (8 – 10 times slower)

In fact, the real value of compression ratio fundamentally depends on the statistical properties of data being compressed. The performance depends on the CPU and hard disk performance.

The *RLE algorithm* (Run Length Encoding) is based on compressing of multiple repetitions of a single character. Concerning to the Hard Disk Manager functionality, RLE provides good results in case of backup partitions that have multiple small files and/or the partition have been performed the *Clear Free Space* operation, or in case of the partition contains sparse data (like .TXT/.DOC/.XLS/.BMP-files).

The *LZW algorithm* (Lempel-Ziv-Welch) is the very popular and effective compression algorithm that is based on building the dictionary of repetitive patterns. Modern archivers (ZIP, RAR etc) use modifications of the LZW algorithm. It provides good compression ratio on almost all kinds of data.

3.3.2 Backup Page



3.3.2.1 Do not backup temporary files

This option provides the ability of ignoring unimportant files when creating backup archives. This feature allows substantially reduce the creation time and size of the backup archive. The current version of the program allows ignoring contents of the PAGEFILE.SYS and HIBERFIL.SYS system files on NTFS partitions.

If this option is activated, Hard Disk Manager backups information about size and allocation of these files, but the program skips saving their contents. At the restoration of a partition, Hard Disk Manager will retrieve PAGEFILE.SYS and HIBERFIL.SYS files with the original size and allocation. Contents of restored PAGEFILE.SYS and HIBERFIL.SYS files may differ from the original ones.

If this option is disabled, Hard Disk Manager places contents of these files in the backup archive.

PAGEFILE.SYS

In Windows XP, the following definition is provided: "A paging file is an area on the hard disk that Windows uses as if it were RAM". Windows NT, 2000 and XP place the swappable part of the virtual memory in the set of files that are generally named "paging file". These files can be located in the root folders of multiple logical drives. Each file is named PAGEFILE.SYS.

Contents of the paging file are re-initialized at the beginning of every Windows session; they are unimportant between Windows sessions.

The paging file usually takes large amount of disk space (by default, the paging file is equal to 1.25 of physical RAM size). Hard Disk Manager saves much time and disk space when excluding contents of this file from a backup image.

HIBERFIL.SYS

The system file HIBERFIL.SYS is used in Windows XP for saving contents of physical RAM when the computer is *hibernated*. The HIBERFIL.SYS is equal to physical RAM in size. Contents of this file are valuable only in case of Windows XP is hibernated and is not re-activated back yet, in other cases they are unimportant. Hard Disk Manager saves much time and disk space when excluding contents of this file from a backup image.

3.3.2.2 Do not control archieve integrity

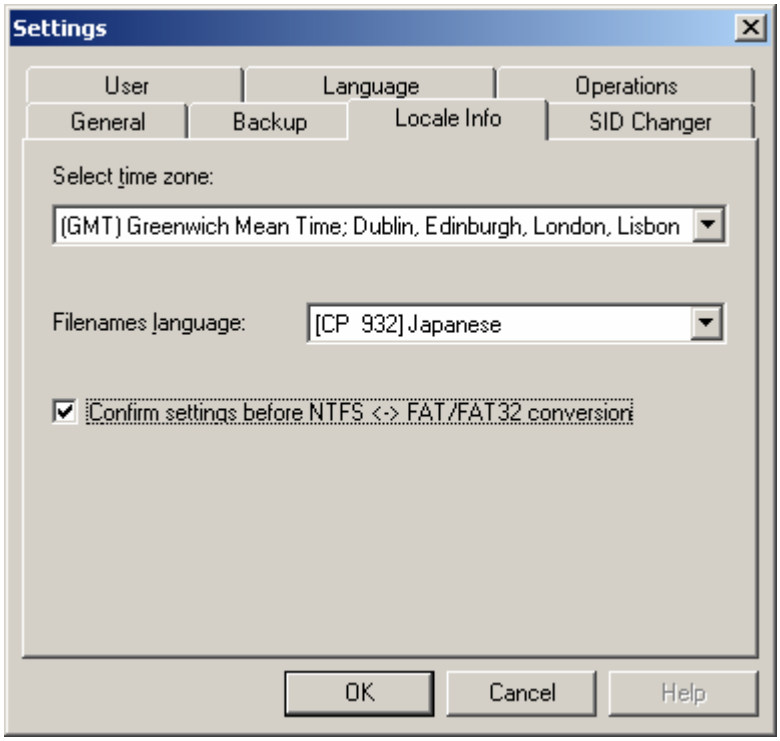
This option allows controlling the generation of the integrity validation code in a newly created backup archives.

By setting this checkmark, one can switch off the function of integrity code generation. It can provide 2-7% performance gain. On the other hand, it reduces the fault-diagnosis abilities of the program.

During creating a new backup image, Hard Disk Manager calculates additionally the special data-dependable code and includes it to the archive. This code allows performing the fast and effective verification of the archive integrity.

The current version of the program uses the so-called CRC32 (32-bit Cyclic Redundancy Code) integrity validation code.

3.3.3 Locale Info Page



The settings presented on this page affect on the conversion of filesystems "FATxx➔NTFS" and "NTFS➔FATxx".

The thing is that NTFS and FAT16/FAT32 filesystems use different standards for *filenames* and *file timestamps* (*Created*, *Modified* and *Last access* times). The program uses Locale (=Regional) Settings in order to correctly convert mentioned values. Incorrect settings may lead to corruption of non-English filenames.

By default, Hard Disk Manager takes locale settings from the system. Still, one can customize default locale settings that should be used in filesystem conversion. Non-Windows versions of Hard Disk Manager take into account these settings, too.

3.3.3.1 Select time zone

Set the *time zone* that should be used in filesystem conversion.

The thing is that NTFS keeps *file timestamps* (*Created*, *Modified* and *Last access* time) in GMT (*Greenwich Mean Time*) while FAT uses the unadjusted local date and time for these values. The program takes into account the difference between internal formats of file timestamps and uses time zone information for adjusting timestamp values.

In case of time zone is incorrectly assigned, all files and directories on a converted partition become "older" or "younger" for a fixed time value (which lies within the range 0-24 hours).

Usually, such a "time jump" does not lead to problems. Still, there is a probability that some software stops working correctly in this situation.

3.3.3.2 Filenames language

Set the *code page* value that should be used in filesystem conversion.

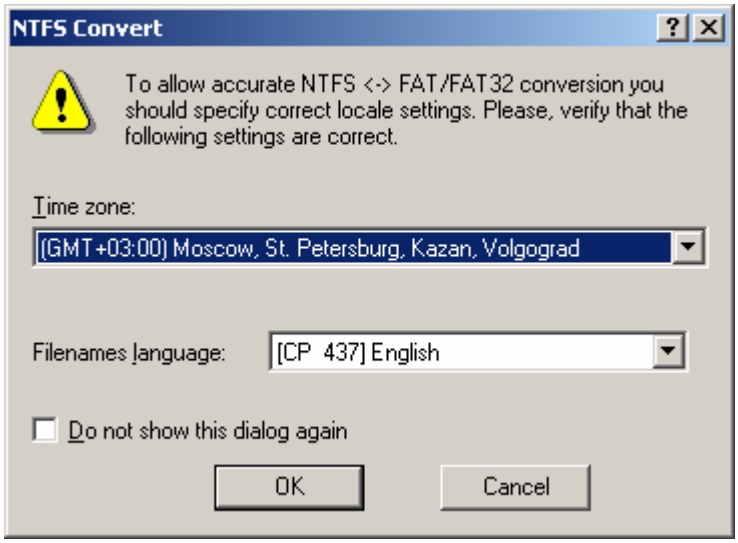
The thing is that NTFS stores filenames in *Unicode* while FAT16 and FAT32 filesystems use *ANSI encoding* for saving *short filenames* (which are also named *DOS aliases*). Code page information is required for correct conversion of non-English filenames from Unicode to ANSI and vice versa.

In case of code page is incorrectly assigned, non-English filenames and filenames with mixed languages may become unreadable.

Note, that if three or more languages are used in filenames on NTFS partition, the conversion to FATxx almost certainly leads to a corruption of non-English short filenames.

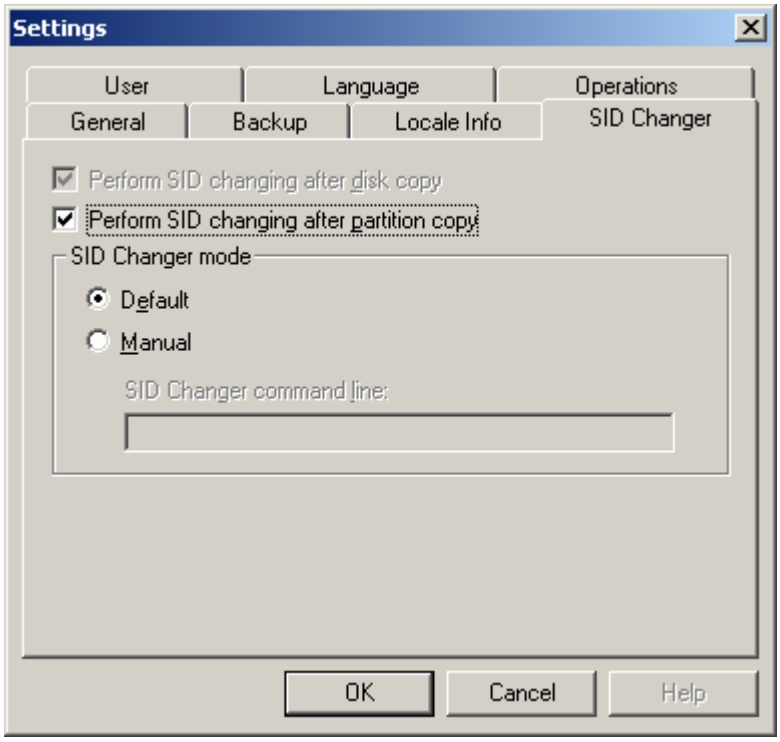
3.3.3.3 Confirm settings before NTFS <-> FAT/FAT32 conversion

If this option is enabled, the program will display the dialog of locale settings confirmation every time the Convert Filesystem operation is performed:



Initially, the program displays default locale settings, but a user can change for parameters to be used, for this time.

3.3.4 SID Changer Page



The settings presented on this page control the use of the built-in *SID Changer* module. Hard Disk Manager provides the ability of automatic activation of this module after the following operations:

- Copy a partition to another one
- Copy a hard disk to another one
- Restore a partition from image
- Restore a hard disk from image

See the Glossary and SID Changer to learn more about the situations when the SID Changer utility may be useful.

3.3.4.1 Perform SID changing after disk copy

Set this checkmark in order to the program automatically runs the built-in SID Changer module after every copying or restoration of an entire hard disk (i.e. after completing Restore Hard Disk or Copy Hard Disk operations).

3.3.4.2 Perform SID changing after partition copy

Set this checkmark in order to the program automatically runs the built-in SID Changer module after every copying or restoration of a partition (as the part of Restore Partition, Selective partition restore or Copy Partition operations).

3.3.4.3 SID Changer Mode

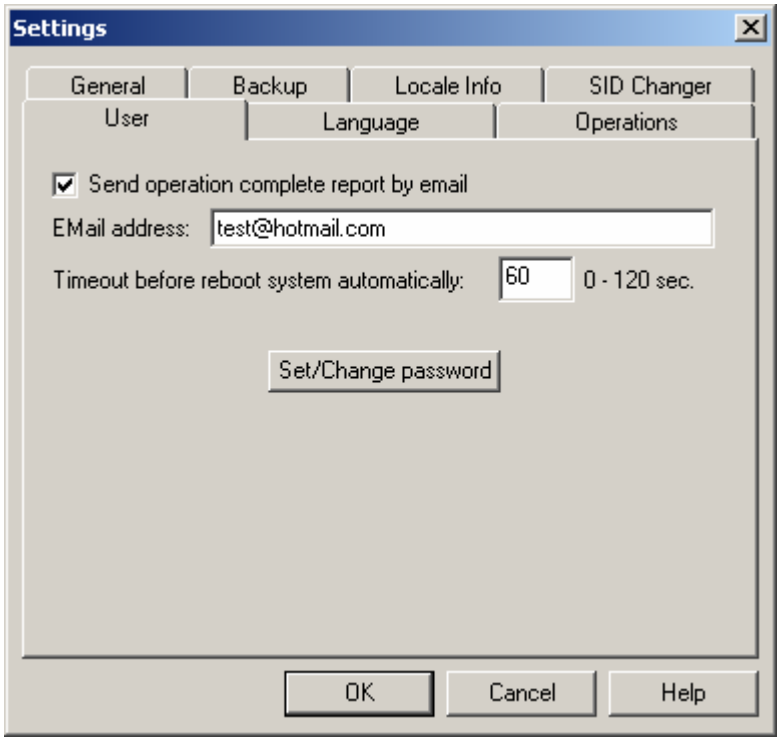
Select the functioning mode of the SID Changer.

- In the **Default** mode, the program:
  - ⇨ automatically generates the new computer SID value
  - ⇨ scans BOOT.INI files for searching Windows installations
- In the **Manual** mode, the program allows to define:
  - ⇨ new computer SID value.
  - ⇨ directories list for searching Windows installations

3.3.4.4 SID Changer command line

Enter the parameters that must be passed to the SID Changer utility. See the detailed description in the chapter SID Changer.

3.3.5 User Page



These settings are available in Windows-based version of the Hard Disk Manager only.

3.3.5.1 Send operation complete report by email

Set this checkmark so that the program sends the notification about completion of scheduled virtual operations via email.

Some operations may take noticeable time, and the completing of the large list of pending operations may take very large time, too. The program supports the feature of remote notification about finishing the execution of pending operations.

3.3.5.2 Email address

Enter the email address to be used for the email notification about completing operations.

3.3.5.3 Timeout before reboot system automatically

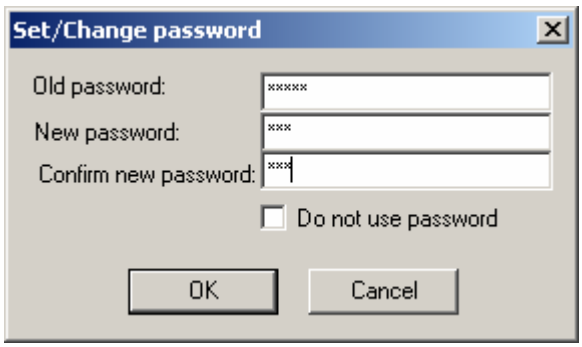
Set the **positive non-zero value** to in the textual field in order to define the timeout for auto-reboot. Clear the field or enter the **zero value** to disable the automatic rebooting feature.

Some operations require the system to be rebooted. This setting controls the behavior of the program in such situations. By default, the program stops the execution and waits until the user confirms the reboot. Set the timeout to eliminate the stopping, so that the program becomes able to complete any operations without the user's intervention.

3.3.5.4 Set/Change password

It is possible to protect the program from incompetent or unauthorized modifying of disk contents by setting the password on the physical execution of the operations.

To set the password, press the **Set/Change password** button.



To set the new password, enter the old one in the **Old password** field. Then set the new value in the **New Password** field and duplicate it in the **Confirm new password** field. The checkmark **Do not use password** allows to disable the password protection.

Note: the old password is required to change or disable the password.

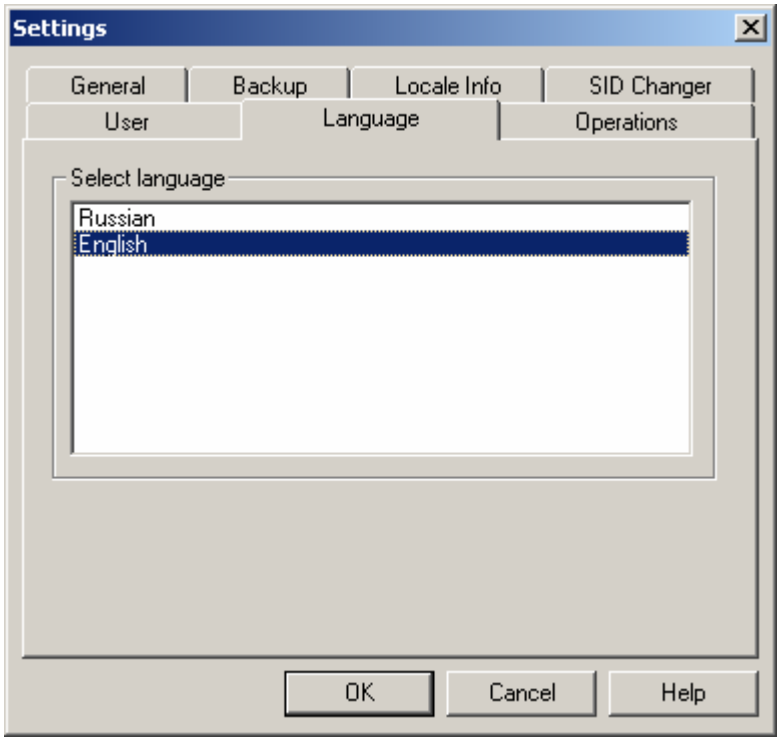
In case of the password is defined, the program will behave in the following manner:

- Anyone can call any operation dialogs;
- Anyone can virtually execute operations (if the program works in the *Virtual Execution mode*);
- Anyone can press **Undo** and **Undo all** buttons that only lead to the modification of the List of Pending Operations.
- Anyone can activate operations that do not modify disk contents, such as:
  - ⇨ Check Image Integrity
  - ⇨ Browse Image
  - ⇨ View Sectors
  - ⇨ Partition/Disk Properties
- All operations that produce the real modification of disk contents require password to be entered for the physical execution:

In case of working in the *Immediate Execution mode*, the program asks the password immediately after submitting parameters of the operation.

In case of working in the *Virtual Execution mode*, the program asks the password at initiating the applying of pending operations.

3.3.6 Language Page

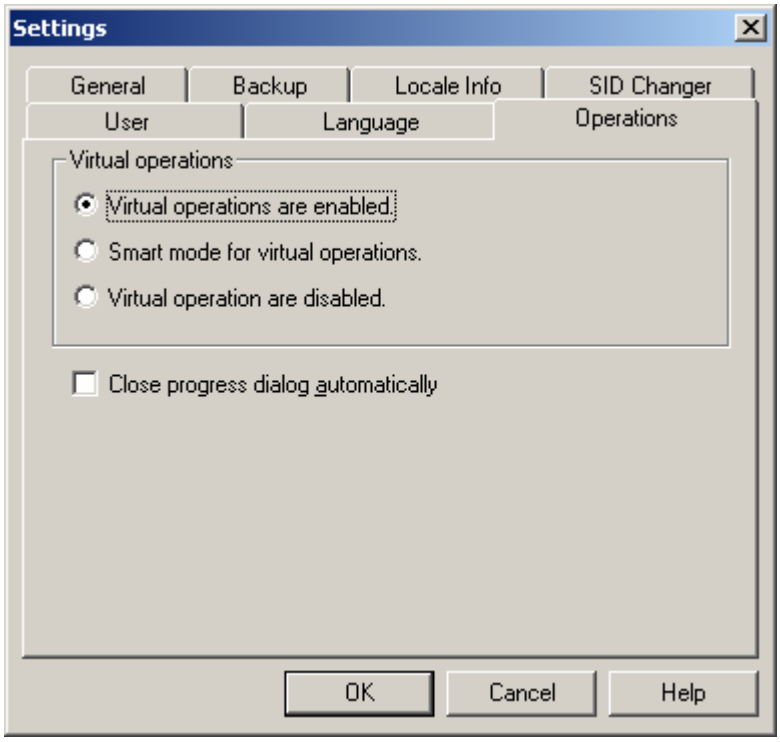


The Windows-based version of Hard Disk Manager supports the on-the-fly changing of the interface language.

3.3.6.1 Select language

This is contains the list of language resources available. Just select the desired interface language and press **OK** button.

3.3.7 Operations Page



3.3.7.1 Virtual operations are enabled

Select this choice to switch the program to the *Virtual Execution mode*.

In this mode, all operations, which support virtual execution, will be scheduled in the *List of Pending Operations* for future completing (see the chapter Virtual Execution mode for more details).

The program displays the *virtual state* of hard disks layout – this state will be reached after completing all of pending operations. To really execute pending operations, one should press the **Apply** button.

3.3.7.2 Smart mode for virtual operations

Select this choice to switch the program to the "smart execution" mode (see the chapter Smart mode for virtual operations for more details).

In the smart execution mode, Hard Disk Manager executes all lengthy operations in the virtual mode (i.e. these operations are scheduled in the *List of Pending Operations* for deferred execution).

Instead, Hard Disk Manager executes quick operations immediately, in case of there are no pending operations in the List.

Quick operations:

- ⇨ Hide/Unhide partition
- ⇨ Set partition active/inactive
- ⇨ Mount partition
- ⇨ Set partition Label

3.3.7.3 Virtual operations are disabled

Select this choice to switch the program to the *Immediate Execution mode*. All operations will be executed immediately after submitting operation's parameters (see the chapter Immediate Execution mode).

3.3.7.4 Close progress dialog automatically

When executing operations, Hard Disk Manager displays the *Progress Information* window that contains the performance statistics and the brief execution log (see the section Progress Information). By default, Hard Disk Manager holds the *Progress Information* window until the user presses the **Close** button, so that the user is able to explore the brief execution log and statistics.

Set this checkmark to force the program automatically closing the *Progress Information* window after completing execution of accumulated pending operations.

4 Getting Started with Hard Disk Manager

This chapter describes how to proceed with most frequent backup and partitioning operations by using Hard Disk Manager. It is recommended to familiarize with the Settings overview.

4.1 About executing the operations

4.1.1 Backup Wizard vs. manual management of operations

You can use either *Backup Wizard* or *Operation Dialogs* to execute operation from the program's interface. Hereafter you will find the detailed description of all Operation Dialogs, with mention how each setting affects the operation performance.

Backup Wizard

Backup Wizard guides you through the set of Wizard's screens. Each screen contains one simple question and prompts the limited list of possible answers. Passing all these screens the Wizard collects enough information to initiate the required operation. Finally, Hard Disk Manager will execute the operation.

Backup Wizard is easy-to-use, but it does not allow configuring the performance and implementing some advanced features. The way to fully control the behavior and performance of the program is to use *Operation Dialogs*.

Manual management with Operation Dialogs

In the manual management mode, the program allows customizing all controllable parameters of operations in appropriate dialogs. With using this ability, one can effectively control performance and use all abilities of the program.

Usually, all parameters required for an operation are collected in one dialog. The program suggests some consistent initial values for all operation's parameters, and in most cases, one can permit default settings.

4.1.2 How Hard Disk Manager executes operations

4.1.2.1 Initiation of an operation

First, one should initiate an operation. A common procedure is the following:

1. A user should select an object to operate (a *partition*, a *free block* or a *disk*).
2. A user should choose an operation to be performed.
3. The program displays the appropriate dialog with the parameters required for the operation execution.
4. A user should assign the parameters and then click the **OK** button.

Next actions depend on the selected operation and program's settings. In case of virtual operations are disabled, the program immediately executes the operation. Otherwise, the program includes the operation in the *List of Pending Operations*.

4.1.2.2 Inspecting pending operations

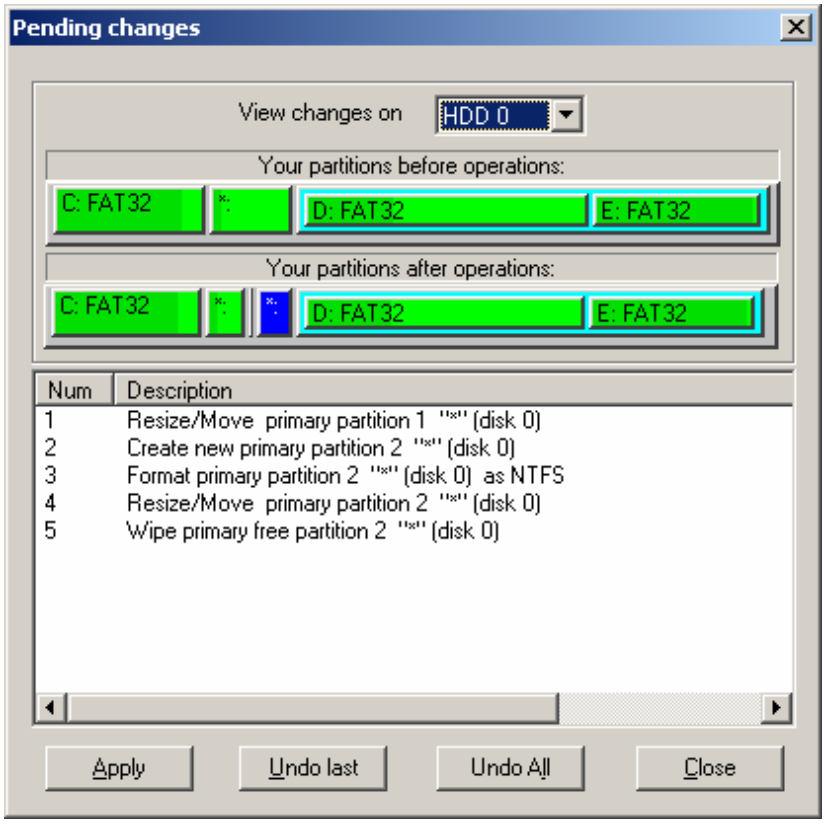
In case of virtual operations are enabled, the program includes operations in the *List of Pending Operations*. One can examine the list prior the program really completes operations. To inspect the List of Pending Operations, make one of following:

- Select in the main menu:

  (menu) Operations ➔ View pending changes…

- Press the **Changes** button in the Virtual Operations Toolbar

After this action, the *pending changes* dialog appears, in case of some virtual operations are accumulated in the List:



This dialog has the following functionality:

View changes on [HDD##]

This pull-down list allows selecting a hard disk for previewing changes in its layout.

Two *UDP controls*, which are placed below and labeled "**Your partitions before operations:**" and "**Your partitions after operations:**", display expected layout changes for the selected hard disk.

The textual box in the middle of the window contains the *List of Pending Operations*.

Buttons on the bottom of the window provide the following functionality:

Apply	Immediately execute all accumulated pending operations. The *List of Pending Operations* will be cleared.
Undo last	Undo the last pending operation in the list.
Undo All	Cancel the entire list of pending operations.
Close	Just closes the dialog. No changes in the *List of Pending Operations* will be performed.

The functionality of inspecting pending operations is available only in case of virtual operations are enabled (see the section Virtual operations).

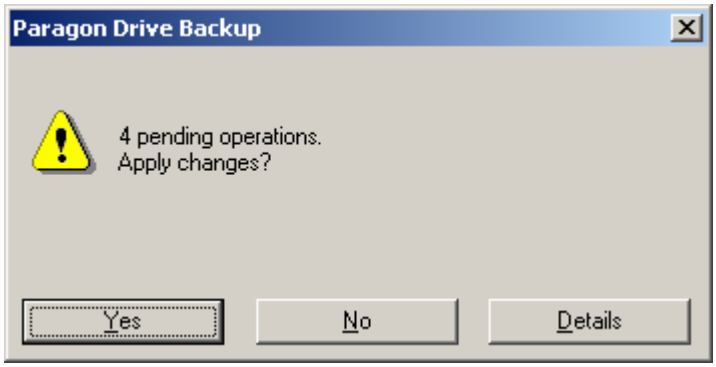
4.1.2.3 Applying operations

For immediate execution of accumulated operations, a user should make one of following:

- Press the **Apply** button in the *Virtual Operations Toolbar*.
- Select in the main menu:

  Operations ➔ Apply changes

In case of there are some operations are in the *List of Pending Operations*, the program displays the following warning message:



Buttons on the bottom of the window provide following functions:

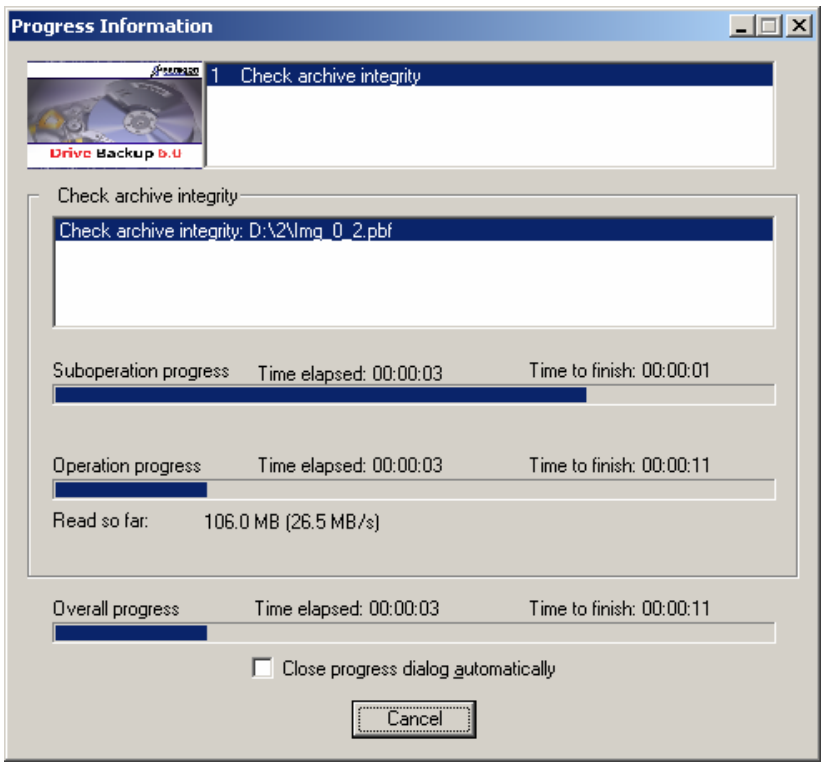
Yes	Immediately execute pending operations
No	Cancel all pending operations
Details	Inspect the List of Pending Operations

In DOS and Linux environments, the program executes the entire list of operations within the single session.

In Windows, the execution may vary: Hard Disk Manager requires the system reboot to process locked and system partitions or hard disks.

4.1.2.4 Progress Information

The real execution of operations is accompanied by displaying the *Progress Information* dialog:



The dialog displays:

- The complete *list of operations* to be executed (on the top of window); the currently executed operation is highlighted
- *Brief information* about actions being performed (a textual box in the middle of the window).
- Three progress bars:
  - ⇨ *Overall progress* (bottom) displays the overall progress for entire list of operations
  - ⇨ *Operation progress* (middle) displays the progress for currently performed operation
  - ⇨ *Suboperation progress* (top) displays the progress for currently performed suboperation, for compound operations only.

The program displays **Time elapsed** and **Time to finish** for each progress bar. The Operation progress includes extended information for some operations: amount of processed and remaining data, average transfer rate.

In addition, the *Progress Information* dialog contains the following control elements:

Cancel (Close)

This button provides the ability of aborting operations. The program will break the currently performed and all following operations in the list of pending operations.

After all operations in the list are completed or aborted, the **Cancel** button changes to its caption to "**Close**".

Close progress dialog automatically

This checkbox duplicates the functionality of the appropriate program's option (see the section Settings overview➔Close progress dialog automatically). In case of the checkbox is set, the program automatically hides the **Progress Information** dialog after completion all operations, without waiting user's reaction.

4.1.2.5 Working with locked/system partitions and hard disks

If some of pending operations is aimed to a locked/system partition, the program will save the rest of the List of Pending Operations and reboot the computer. All operations from the list will be executed in some single-tasking mode, until completion.

In Windows NT, 2000 and XP

Hard Disk Manager processes locked partitions in the so-called "Startup Bluescreen" by using the special utility named *Bluescreen Component*. After completion the operation, the program will reboot the system to the Windows session.

In Windows 95 and 98

Hard Disk Manager reboots to the "true" DOS session and starts the DOS-based version of the Hard Disk Manager in the unattended mode. After completion the operation, the program will reboot the system to the Windows session.

In Windows ME

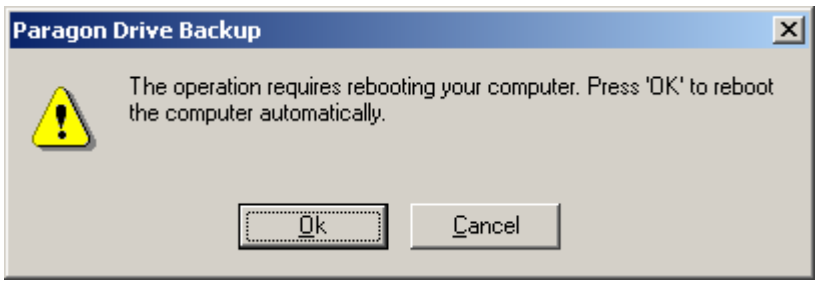
Hard Disk Manager requires using a preliminary made bootable diskette with the DOS-based version of Hard Disk Manager. The next chapter contains the detailed explanation of this functionality.

4.1.2.6 Working with locked partitions/disks in Windows ME

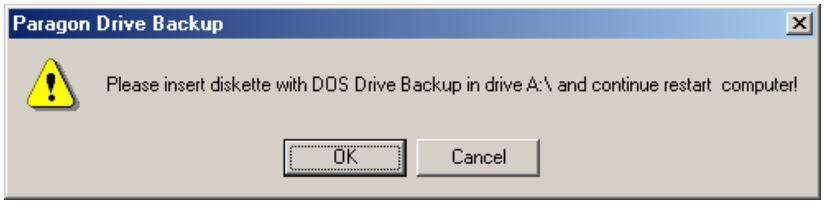
Unfortunately, Windows ME has no handy single-tasking environment. For this reason, Hard Disk Manager requires to use either a DOS bootable diskette with the diskette-based version of Hard Disk Manager, or the Paragon Recovery CD with Linux-based version of Hard Disk Manager, in order to process locked partitions and hard disks in Windows ME.

The special utility named Diskette Build Wizard is developed to simplify the process of making bootable diskettes with the diskette-based version of the Hard Disk Manager:

1. Before working with Hard Disk Manager, run the Diskette Build Wizard utility and make the DOS bootable diskette with the diskette-based version of the Hard Disk Manager.
2. Run the Windows-based version of Hard Disk Manager.
3. Initiate a required operation and define all parameters.
4. Before starting an operation, the program checks whether partition is locked or not.
5. If a partition/disk is locked, the program asks for reboot the system:



   Press **OK** button to reboot the system and complete an operation, or press **Cancel** button to abort it.
6. The program asks inserting a diskette with Hard Disk Manager in the floppy drive [A:]:



7. Insert the required diskette; the program will pass the task to the diskette-based program.
8. Reboot the computer, keep the diskette in the floppy drive. Ensure that the computer is configured to boot from floppy first.
9. The diskette-based program will automatically start the operation in unattended mode.
10. When the diskette-based program completes the operation, remove the diskette from the drive and reboot the computer.

4.1.2.7 Executing operations in the fail-safe mode

Hard Disk Manager provides the ability of performing some operations in the *fail-safe mode*. In this mode, the program can resume operations interrupted due to hardware/software malfunctions or power breaks.

When working in the fail-safe mode, the program permanently keeps the log of the operation progress. If the operation was interrupted abnormally, Hard Disk Manager will detect the log information about an incomplete operation and will suggest either resuming or finally canceling it.

The log supporting noticeably slows down the performance. For this reason, Hard Disk Manager supports the fail-safe mode only for operations that modify a "source" partition, such as Resize & Move Partition, Convert Filesystem and Change Cluster Size. Other operations, such as Backup Partition, Copy Partition and Restore Partition keep the source object intact, so that these operations can be simply restarted in case of an interruption.

Working in the fail-safe mode

A common procedure is the following:

1. A user should make the DOS-based Hard Disk Manager's bootable diskette by using the utility Diskette Build Wizard, or make the Linux-based Hard Disk Manager Bootable CD.
2. A user should run Hard Disk Manager and activate the *fail-safe mode*. in the program Settings (see the section Settings Overview ➔ General Page ➔ Data loss protection).
3. A user should initiate and execute a desired operation.
   The program will start the operation in a seemingly usual manner, but really, it will keep the log of the operation progress.
4. In case of unexpected interruption (failure), the program replaces the MBR bootstrap code with the special bootable code that asks a user continuing the interrupted operation from the bootable diskette or the bootable CD.
5. A user should boot the computer from the Hard Disk Manager bootable CD or from the Hard Disk Manager bootable diskette.
6. The program will detect the interrupted log and will ask a user either continuing the operation or canceling it.
   - ⇨ In case of continuing the operation, the program resumes the operation and completes the modification of a partition.
   - ⇨ Otherwise, the program only restores the old bootstrap, but leaves a processed partition in unusable state.

Manual interruption of an operation

In case of an operation was interrupted by a user, the program will not provide an ability of resuming the operation, even if it works in the fail-safe mode.

Remember, an interruption of some operations leads to the irrevocable corruption of a processed partition.

4.2 Backup Partition

This chapter explains how to make backup images of separate partitions on local and network drives, unmounted partitions and removable media under various conditions.

4.2.1 Overview

Backup is the primary function of Hard Disk Manager. One can backup partitions having important data and system partitions to protect oneself from downtime in case of system malfunction. The partition can be completely restored within few minutes from the backup image.

The backup image includes not only contents of all files but also *filesystem metadata*. In that way the program keeps all information associated with files: the exact structure of directories, location of files on the disk, security information, access quotas and so on. After the restoration, the partition becomes in the pre-backup state.

Take into account that new data added to the partition after making the backup image, will be lost after the partition restoration.

Hard Disk Manager allows to backup partitions of any type. The program backup partitions of known filesystem types by using the fast copying algorithm. All other partitions are saved in the *sector-to-sector* copying mode. See the section Processing partitions of unknown types for more details.

The implementation of the backup function varies in some cases:

1. In Windows, the operation differs for *unlocked* and *locked* (system) partitions. To process locked partitions, the program need to reboot the computer.
2. In addition, the processing of *locked partitions* differs for Windows 95/98, Windows NT/2000/XP and Windows ME.

4.2.1.1 Where you can save backup images

Hard Disk Manager allows saving backup images on the following storage media:

- *Mounted* local drives (i.e. partitions having the drive letter assigned).
- Mapped network drives (i.e. network resources with the drive letter assigned).
- Mounted removable media (ZIP, LS-120, MODD, ZIV etc).
- Mounted USB drives and Compact Flash cards (in Windows only).
- Any read-write media mapped in the system.

- Unmounted partitions formatted to NTFS, Ext2 and Ext3.
  Hard Disk Manager uses the built-in *universal filesystem drivers* (UFSD) to access unmounted partitions on local hard disks.
- CD-R(W) and DVD-R(W) drives (in Windows and Linux). The equipment should be detectable and available in Windows Device Manager. This feature is explained in the chapter Burning images of partitions on CD/DVD-R(W) under Windows.

4.2.1.2 Incremental Backup functionality

Hard Disk Manager provides the Incremental Backup functionality extension for single Primary and Logical partitions.

Briefly, the program produces the exact comparison of current contents of the selected partition with ones saved in the previously made archive of this partition (this archive is named the *parental image*). The difference in contents is saved in the newly created archive.

Hard Disk Manager requires the *parental image* to be corresponding to the partition being archived to a new incremental backup image. Before executing the operation, the program verifies this condition. To distinguish between partitions, the program uses the *location*, the *capacity*, the *filesystem type* and the *serial number* partition's properties.

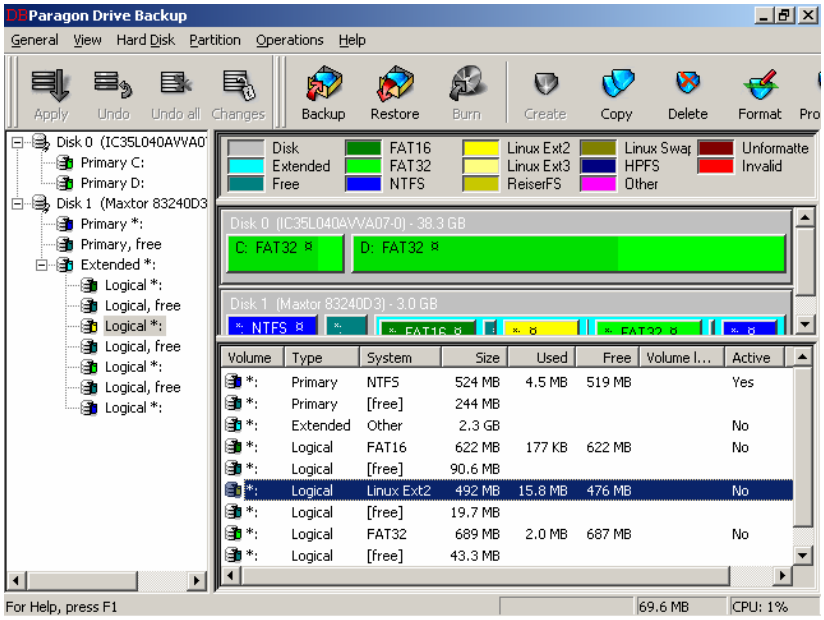
The incremental backup archive cannot be used detached from the parental image. When restoring a partition from an incremental archive, the program at first finds and restores the parental archive, and then implements changes that are saved in the incremental archive.

4.2.1.3 Restrictions

Hard Disk Manager is unable placing the backup image (or some of its volumes) on the partition that is being imaged. One should place the image on another partition, another hard disk, removable media or a mapped network drive.

4.2.2 Initiating the operation

Step 1. Select the partition you want to backup.

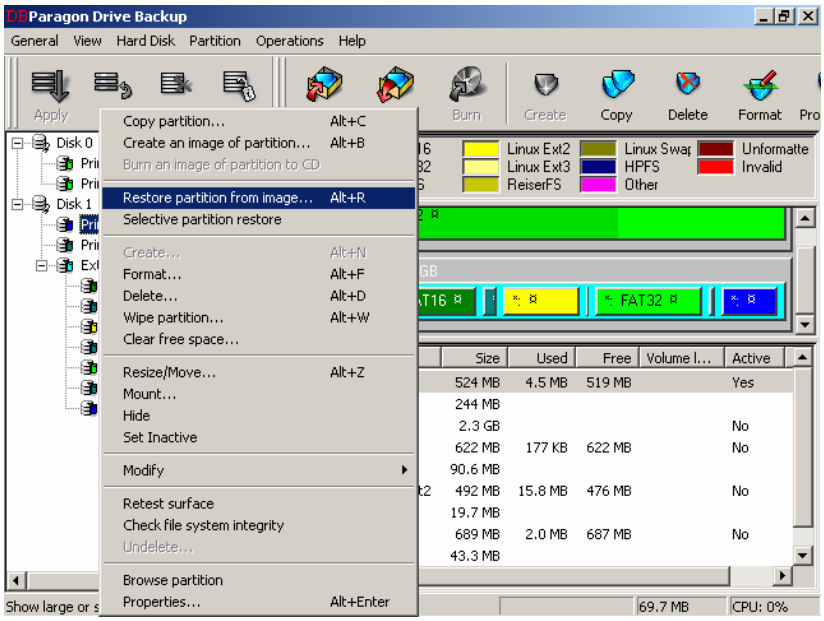


Select some existing partition you want to backup, in the Tree Layout panel in the UDP Layout panel or in the List of Partitions. In any case, the partition will be highlighted in all three panels.

The function is not available for blocks of free space.

After selecting a partition, most buttons on the Main Toolbar will become enabled.

Step 2. Select the operation to perform



Variants:

- Select in the main menu:

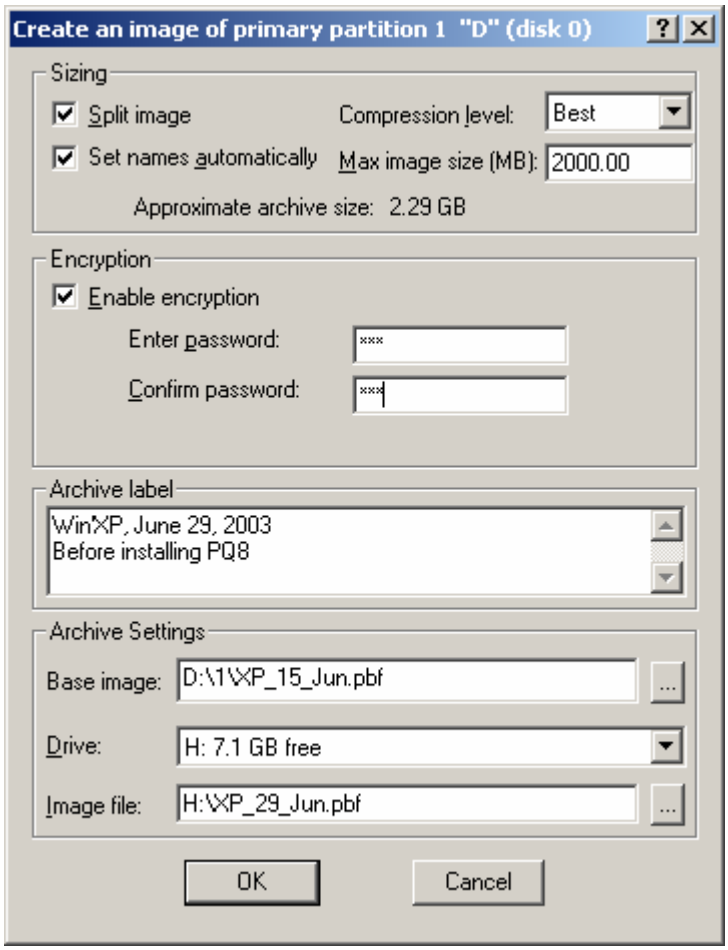
  Partition ➔ Create an image of partition…

- Call the *popup menu* for the selected partition in any of layout panels (click right mouse button) then select the menu item:

  Create an image of partition…

- Press **Alt+B** keyboard combination
- Press **Backup** button on the Main Toolbar.

Step 3. Assign properties of the backup image



Assign properties of the backup image. You can control:

- splitting properties (see Split settings)
- compression level (see Compression settings)
- encryption (see Encryption settings)
- archive description (see Archive label)
- image location (see Image Location)

Initially the program suggests some consistent values for all parameters. In most cases, you can just press **OK** button to confirm the operation. All parameters are explained in the chapter Description of the Backup parameters.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Backup Partition* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the Backup operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.2.3 Description of the Backup parameters

4.2.3.1 Common parameters of the Backup Hard Disk operation

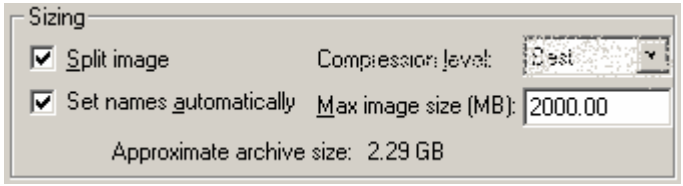
Parameters of the backup operation are subdivided in five groups:

- *Split* settings manage the ability to make multivolumic images and assign default volume size and the automatic generation of volume filenames.
- *Encryption* settings manage the ability to cipher image contents and set the access password

- *Compression* settings manage an ability to squeeze image contents and the compression level
- *Archive label* allows to place custom notes to the image file for easy navigation through images
- *Location* settings allow selecting the most convenient storage to place the image.

As the matter of fact, all *Backup* operations contain these parameters, with probably few minor differences. The only specific feature of the *Backup Partition* operation is the ability of using the *Incremental Backup* function.

4.2.3.2 Split settings



Split image

Set this checkmark ON to enable automatic breaking of large images in multiple volumes. Otherwise, the program will try to place entire image in a single file. See Comments for more details.

Max image size (MB)

This value defines the maximum size of each volume of the image being created. The default value is 600 Mb.

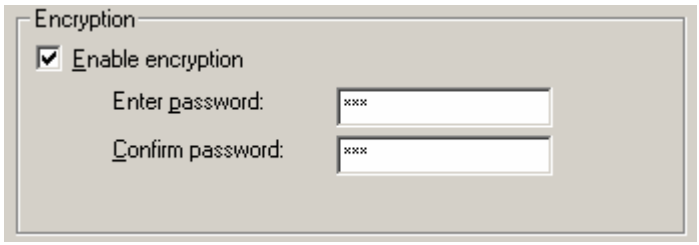
Set splitted file names automatically

When activated, this option forces the program to automatically generate filenames for next volumes of the image. Otherwise, the program will pause after filling each volume to ask the user about the new volume filename until completing the operation. See Comments, section Using multivolumic images for more details.

Approximate archive size

The program estimates roughly the resulting size of a new archive by compressing a small portion of data. This filed displays the estimation results.

4.2.3.3 Encryption settings



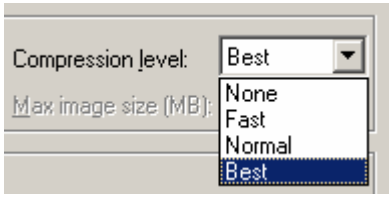
Enable encryption

Set this checkmark ON to enable ciphering image contents and protect accessing to the image by the password. Encryption reduces the overall operation performance. The deceleration percentage fundamentally depends on the hardware parameters.

Enter password & Confirm password

With enabling the encryption, provide the password to be used to access the image. **Note:** the program does not allow "empty" passwords.

4.2.3.4 Compression settings

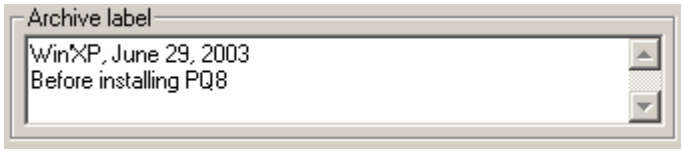


Compression level

Here you can change the compression level, in case of you wish to use the value other than the default one. The default compression level is defined in the program settings. Compression may significantly squeeze the backup image size, but it reduces the overall operation performance.

The greater level, the higher compression ratio and the slower performance. The deceleration percentage fundamentally depends on the hardware parameters. See General settings for more details.

4.2.3.5 Archive label



You can associate a short descriptive text with the image. When getting Archive Info or Restoring Partitions, you can inspect the archive label to ensure that you have selected the desired image.

4.2.3.6 Image Location

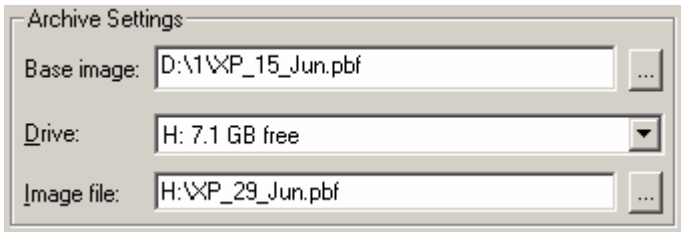


Image file

Assign the desired filename of the backup image in this text box.

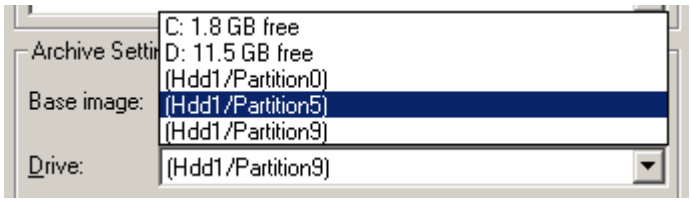
The program generates the default image filename by using the *disk number* and the *drive letter* (or the *partition number*) and suggests placing the image on the local drive having the maximum value of free space.

A user can assign another filename manually, or select an existing file by pressing the button … that is located on the right side of the text box.

Drive

The pull-down list of available drives is very useful when selecting the non-default image location. The list contains:

- *Mounted* local volumes, with mention the drive letter and the amount of available space.
- *Mapped network drives*, with mention the drive letter and the reported amount of available space.
- *Unmounted* NTFS and Ext2 partitions, with mention their location (**disk_number / partition_number**). The program accesses unmounted partitions by using the built-in filesystem drivers.



Base Image

Use this parameter in order to produce an incremental backup archive. In this text box, the *parental archive* filename should be entered (see the section Incremental Backup functionality for details).

A user can assign a filename manually, or select an existing file by pressing the button … that is located on the right side of the text box.

In case of this text box remains empty, Hard Disk Manager creates an ordinary backup archive.

Before executing the *Backup* operation, the program verifies that the selected parental archive corresponds to the chosen partition. Both the selected partition and the one saved in the parental archive should have:

- ⇨ the same Location
- ⇨ the same Capacity
- ⇨ the same Filesystem Type
- ⇨ the same Serial Number (see the Glossary)

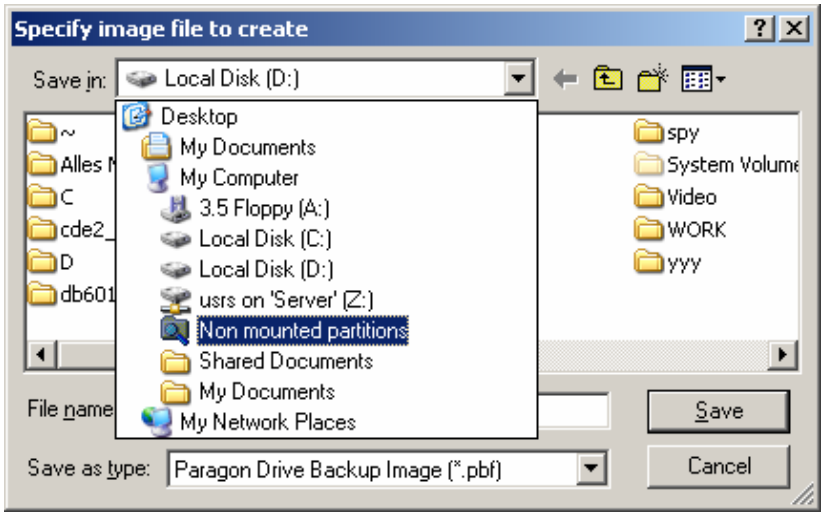
Browse buttons [...]

There are two buttons marked by three close dots in this section of the dialog. Each of them activates the advanced "Open/Save file" dialog. With using this dialog, you are able selecting files on local and network drives, including unmounted local partitions. See the next section for more details.

4.2.3.7 Selecting unmounted partitions as the target location for saving backup images

When creating the backup image of a partition or a hard disk, you are able to select unmounted local partitions of NTFS and Ext2/Ext3 type as the target location for saving the backup image.

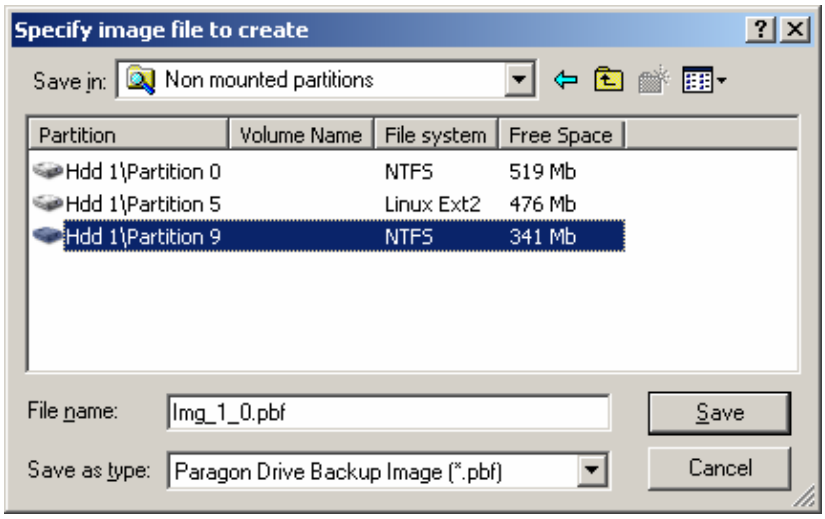
1. Open the "Create an image of partition" dialog.



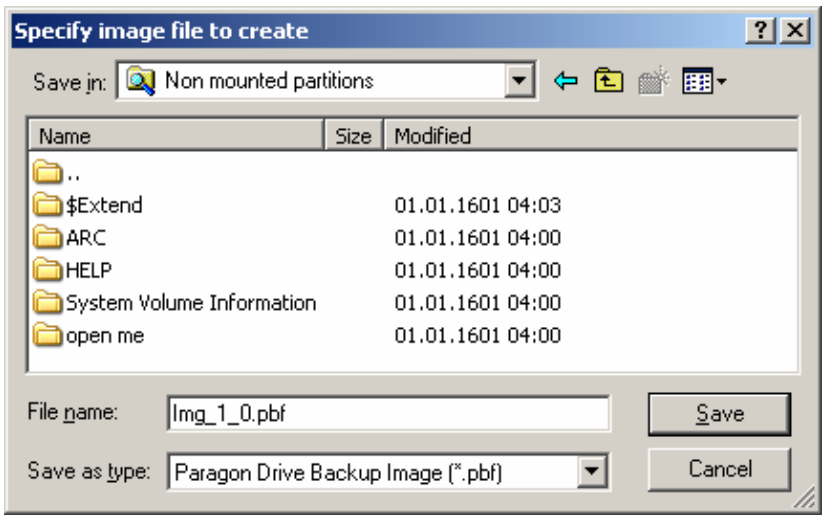
2. Press the **Browse** button (in the bottom of the window) to open the advanced "Save in file" dialog.

3. On the dialog, select the "Save in" pull-down list of volumes
(it is located on the top of the window).

4. In the pull-down list, click on the item "Non-mounted partitions".
(the item is located bottom the list of mapped volumes)



5. The summary list of unmounted local partitions will be displayed.
The list includes NTFS and Ext2/Ext3 partitions on all local hard disks. Every item contains the *drive number / partition number*, the filesystem type and the amount of available space.
6. Select some unmounted partition to proceed.



7. Next, the program will browse the contents of the selected partition. You can open any directory on the partition being explored.
8. Finally, click the button "Save" or press ENTER to select the target location of the backup image .

4.2.4 Running the Backup operation

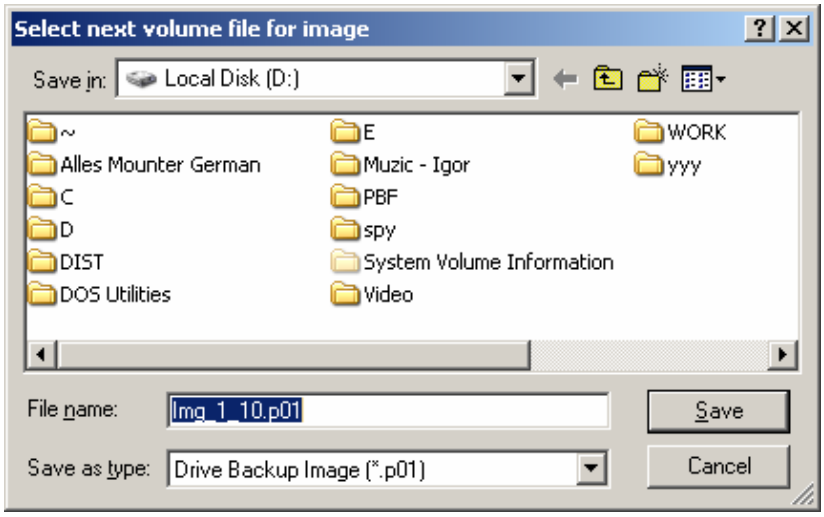
During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Estimated archive size
- Averaged read & write speed and summarized data transfer rate

In case of a multivolumic image is created, the writing of each volume is treated as suboperation, and the program displays information about suboperation progress.

In case of a user disables automatic filename generation, the program pauses after filling each volume and asks for the next volume filename:



By default, Hard Disk Manager suggests an automatically generated filename, but a user can change it. See the chapter How Hard Disk Manager generates filenames for multiple volumes for more details.

4.2.5 Comments

4.2.5.1 Processing partitions of unknown types

The program allows to backup partitions of any type. By default, the program uses the fast copying algorithm for copying partitions of known filesystem types; other partitions are saved in the *sector-to-sector* copying mode. In the fast copying mode, the program requires the on-partition filesystem must be in the consistent state.

In case of turning ON the *sector-to-sector* copying mode, the program copies all sectors of the selected partition regardless of the filesystem type and consistency (see the chapter Settings overview, section General Page, to learn how to switch between copying modes).

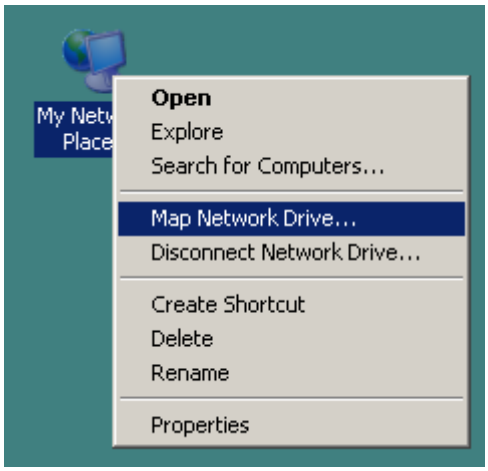
4.2.5.2 How to map a network drive in Windows

Hard Disk Manager uses the file service of the operating system to access network resources. For this reason:

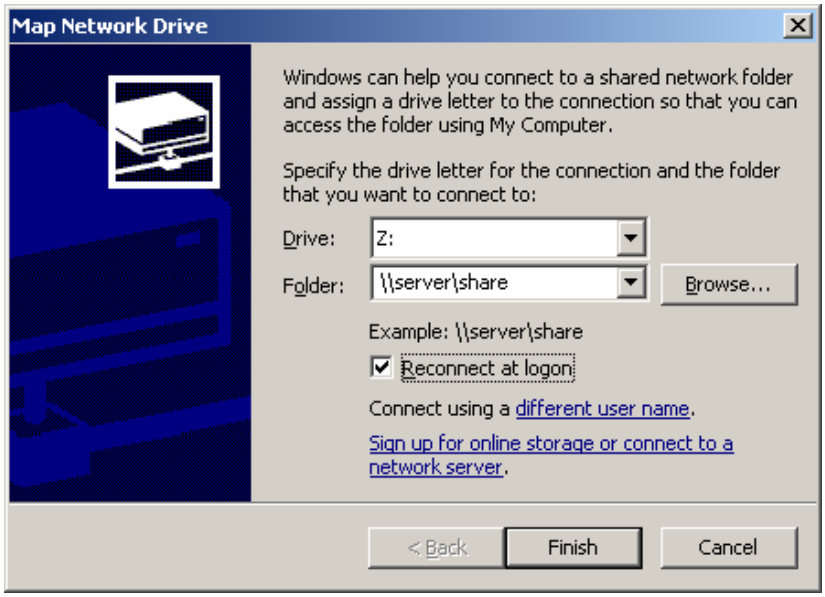
- In Windows NT, 2000 and XP the program can save backup images on any network resource that is available for the full access, regardless of is it mapped in the system or not.
- In Windows 95, 98 and ME the program can use mapped network drives only.

To map a network resource (i.e. assign the drive letter to the resource), make the following actions:

1. Select the icon "**Network Neighboring**" (or "**My Network places**") located on the Windows Desktop and activate the popup menu by clicking right mouse button:



2. Select the item "**Map network drive**". The system dialog "**Map network drive**" should appear:



3. Select a network resource to be mapped
4. Select a drive letter to be assigned to that resource
5. Click **Finish** button to complete the action

4.2.5.3 Most frequent situations in which you may need to use multivolumic images

By default, Hard Disk Manager tries placing the image of a partition in a single file. However, there are situations when a single file is inconvenient or impossible. In this case, one should use the multivolumic image:

You intend to place a large backup image on a FAT32 partition.

FAT32 filesystem limits file size to 4Gb. Therefore, you need to split an image in volumes of 4Gb or less. Generally, Hard Disk Manager detects such situations and automatically split large images in multiple volumes.

You intend to place a large backup image on a mapped network drive.

The network redirector supports only files less than 2Gb in size. Therefore, you need to split an image in volumes of 2Gb or less in order to avoid access problems. Generally, Hard Disk Manager detects such situations and automatically split large images in multiple volumes.

You wish to place a large backup image on removable media

In case of the program does not support the used media type, one can temporarily save a backup image on a hard disk and later move it on removable media by self.
In this case, one can force the program to split the image in volumes of required size.

4.2.5.4 How Hard Disk Manager generates filenames for multiple volumes

In fact, a *backup image* is a special database that can keep various types of information. Hard Disk Manager can combine backup images in hierarchic structures. A compound image consists of multiple files including subordinate images. The primary volume of a backup image keeps information about the hierarchic structure and subordinate volumes.

The program supports both automatically generated and manually defined volume filenames. In case of automatic generating volume filenames, the program uses the following rules:

- The filename of the *primary volume* (i.e. the first file of the multivolumic image) is actually one defined by a user in the *Backup Partition* dialog (or *Backup Disk* dialog). The default file extension of the primary volume is **.PBF**, but it can be changed by the user.
- By default, Hard Disk Manager generates the filename of the *primary volume*:

  `Img_DDpp.PBF`

  where:
  - `DD` - the two-figure hexadecimal number of the hard disk that contains the partition being archived
  - `pp` - the two-figure hexadecimal number of the partition being archived

- Filenames of all other volumes of the multivolumic backup image meet the requirements of the so-called *8.3 filename format* and have the following format:

  `xxxxDDpp.Tnn`

  where:
  - `xxxx` - the first four characters of the image name (4 characters from the filename of the primary volume)
  - `DD` - two-figure hexadecimal number of the hard disk that contains the partition being archived
  - `pp` - two-figure hexadecimal number of the partition being archived
  - `T` - the key of the partition type:
    - ⇨ **P** – primary partition
    - ⇨ **E** – Extended Partition
    - ⇨ **L** – logical Partition
  - `nn` - two-figure hexadecimal number of the volume (within the subordinate image)
- By default, the program places all the volumes in the same directory.

For example, the image named "MyLargeImage" will consist of the following files:

Single-volumic archive	Multivolumic archive	
	Primary partition	Logical partition

MyLargeImage.PBF	**MyLargeImage.PBF** **MyLa0002.P00** **MyLa0002.P01** **MyLa0002.P02** **…**	**MyLargeImage.PBF** **MyLa0007.L00** **MyLa0007.L01** **MyLa0007.L02** **…**

Switch off the automatic filename generation to manually set filenames of volumes. In this case, you are able to place volumes in different directories, on different drives and even on different media. After filling the current volume, the program will pause and ask you for the next volume name.

4.2.5.5 How to bypass the checking of the free space

When the **Split image** feature is switched off, Hard Disk Manager checks the amount of available space on the media targeted for saving the image. In this case, the program will not start the operation in case of the target media have insufficient memory to store the uncompressed image.

To bypass the check of amount of free space:

1. Switch ON the **Split image** feature.
2. Set the volume size to some (very) large value.
3. Disable the automatic filename generation (it is highly recommended).
4. Set the compression level to some high value (the highest level preferred).

When using the feature of the *automatic volume name generation*, avoid placing backup images on media that have no enough free space to fit entire image.

4.2.5.6 Using Compression

Compression allows reducing the image size at the expense of time required for the operation. The implementing of data compression slows the operation's performance. Deceleration and compression ratio fundamentally depend on hardware performance and statistical characteristics of data being compressed. The approximate parameters of the compression received in our testing lab are represented in the section Compression level.

4.2.5.7 How to backup corrupted partitions

To backup a partition that has a corrupted filesystem, switch the program to the *sector-to-sector* mode (see the section Fast copying algorithm):

(menu) General ➔ Settings… ➔ (page) General ➔ Copy all sectors 1:1

By default, Hard Disk Manager works in the *fast copying* mode. Before making the operation, the program performs the filesystem check and accepts only valid filesystems. In the *sector-to-sector* mode, the program skips the filesystem check and just copies all sectors "as is". The image obtained is much larger, but the program enables corrupted and unknown partitions.

4.2.5.8 How to automate the system reboot for processing the locked partitions

When processing locked partitions, Hard Disk Manager asks a user for reboot. By default, the program pauses the execution until a user makes a choice and presses **OK** or **Cancel**. Still, there is the ability to continue the program's working without the manual intervention: in the program settings, on the page User Page, set the non-zero value to the filed Timeout before reboot system automatically.

This option controls the period (in seconds) while the program waits for a user reaction. In case of no intervention during timeout, the program automatically reboots the computer.

4.2.6 Backup system and locked partitions

The basic difference between *locked* and *unlocked* partitions is that contents of locked partitions are "volatile" and can unpredictably change during the lengthy backup operation (see Glossary). In this case, the backup image may become inconsistent.

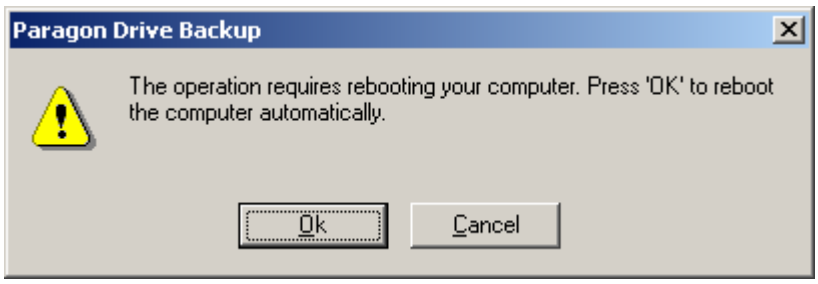
Hard Disk Manager cannot block write operations of other applications, so it tries to reboot the system to a single-tasking environment to eliminate the probability of mutual interference of programs.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

Hard Disk Manager uses different components in different versions of Windows. In addition, the processing of locked partitions differs in various Windows versions.

4.2.6.1 Backup locked partitions in Windows NT, 2000 and XP

In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase to operate the locked partitions:



1. Before starting the operation, the program checks whether partition is locked or not.
2. If the partition appears is the locked one, the program asks for reboot the system. Press **OK** button to reboot the system and complete the operation, press **Cancel** button to abort the operation.

By default, the program pauses the execution until the user makes the choice. For configuring the unattended working, see the section How to automate the system reboot for processing the locked partitions.

3. The program silently schedules the *BlueScreen Component* to run at next Windows startup and to execute the required operation.
4. Then the program reboots the computer.
5. At the next system boot-up, the Bluescreen Component executes the operation in place of the Windows-based version. The BlueScreen Component will display the operation progress in the console-like style.
6. In case of creating the multivolumic backup image with the option **Set splitted file names automatically** switched off, the program asks the user for the filename of the next volume. The most bottom lines of the console output will be the following:

```
Get new filename of the file for subsequent writing.
(Type exit to cancel operation.)
D:\Img_1_Z.p01
```

7. You should choose one of the following alternatives:
   - ⇨ press ENTER to proceed with the suggested filename.
   - ⇨ change the filename to proceed with another one.
   - ⇨ clear the filename and type the word "*exit*" to abort the operation.
8. By default, Hard Disk Manager suggests a new filename, but the user is able to change this value.
9. After the BlueScreen Component completes the operation, the Windows session begins and the Windows-based version of Hard Disk Manager comes up.

4.2.6.2 The text editing abilities in the BlueScreen Component

- Only printable characters and the *BackSpace* key are acceptable.
- Other keys (including *CAPSLOCK*, *Ctrl*, *Alt*, *Shift*) are unacceptable.
- Only English letters are available
- All alpha letters are entered in the *lowercase*.

4.2.6.3 Limitations of the BlueScreen Component

There are some minor functionality limitations of the BlueScreen Component that come from the inaccessibility of some Windows services during the Startup Bluescreen phase:

- The version of the BlueScreen Component provided with Hard Disk Manager interacts with the user in the console-like style. Such useful functions like *Browse disk contents* and *Search files* are not available. In case of the program asks a user to enter a filename, the user must enter a filename "blindly".
- The BlueScreen Component allows input only English letters.
  The text is in *lowercase*, only *BackSpace* editing key is supported.
- During the *Startup Bluescreen* phase, the network redirector does not functioning. For this reason, do not locate backup images on mapped network drives because their contents are unavailable at the *Startup Bluescreen* phase.

The detailed information about the interaction between Windows-based and Bluescreen components of Hard Disk Manager is discussed in the *Technician Manual*.

4.2.6.4 Backup locked partitions in Windows 95 and 98

Windows 95 and 98 include the limited version of the MS-DOS 7 as a part. The "true" DOS environment is available after booting to the DOS session (do not confuse with the *DOS prompt* in the Windows session).

1. Before starting the operation, the program checks whether partition is locked or not.
2. If the partition appears is the locked one, the program asks for reboot the system. Press **OK** button to reboot the system and complete the operation, press **Cancel** button to abort the operation.
3. The application passes silently the task to the DOS-based version of the Hard Disk Manager. Then the Windows-based application just runs the DOS-based program.
4. The DOS-based version of the Hard Disk Manager is configured (through the .PIF-file) to run in the true DOS session. Windows should reboot to the DOS session.
5. The DOS-based program becomes working in the unattended mode with displaying the operation progress and statistics.
6. When completing the operation, the program reboots the computer.

4.2.6.5 Limitations of the DOS-based version in the unattended mode

The DOS-based version of Hard Disk Manager can work either in the interactive mode or in the batch mode. In both cases, it provides the full functionality with the only exception of burning images on recordable CD/DVD discs.

There are some minor functionality limitations of the DOS-based version, which works in the unattended mode. These limitations come mostly from the unavailability some services in the DOS environment:

- Avoid using network mapped drives for placing backup images of locked partitions.
  The thing is that usually DOS environment is not configured for supporting the network, so that network resources will become unavailable in the DOS session.
  Otherwise, you should manually install and correctly configure the NIC drivers and the Network Client software for DOS.
- Do not use USB devices (ZIV, Flash cards etc.) for holding the backup images of locked partitions.
  DOS does not support USB (and there are no USB drivers for DOS on the market). So that after rebooting to DOS all USB devices become unavailable.
- In the unattended mode, the program unconditionally aborts the operation in case of detecting bad sectors, missing the image file or similar problems. To get the user-friendly behavior of the program, run Hard Disk Manager in the interactive mode.
- Take care of providing the DOS session with the device drivers for CD/DVD drives, SCSI and RAID controllers that have no own BIOS.
- Remember that drive letters associated with particular partitions may differ for the DOS and Windows sessions.
- Remember that DOS may fail to access to large partitions. For example, MS-DOS does not work with partitions that are larger than 8Gb.

The detailed information about interaction between Windows-based and DOS-based components of Hard Disk Manager is discussed in the *Technician Manual*.

4.2.6.6 Backup locked partitions in Windows ME

The diskette-based version of Hard Disk Manager can work either in the interactive mode or in the batch mode. In both cases, it provides the full functionality with the only exception of burning images on recordable CD/DVD discs.

The procedure of processing locked partitions and hard disks is described in the section Working with locked partitions/disks in Windows ME.

There are some functionality limitations of the diskette-based version of Hard Disk Manager. These limitations coincide with ones of the DOS-based version that works in the unattended mode (see Limitations of the DOS-based version in the unattended mode). These limitations come mostly from the unavailability some services in the DOS environment:

- Avoid using network mapped drives for placing backup images of locked partitions.
- Do not use USB devices (ZIV, Flash cards etc.) for holding the backup images of locked partitions.
- In the unattended mode, the program unconditionally aborts the operation in case of detecting bad sectors, missing the image file or similar problems. To get a friendly behavior of the Hard Disk Manager, run the program in the interactive mode.
- The diskette-based version may not work with SCSI and RAID controllers that have no own BIOS.
- Take care of providing the DOS session with the device drivers for CD/DVD drives. The Diskette Build Wizard utility simplifies the process of configuring the diskette for supporting CD & DVD drives.
- Remember that drive letters associated with particular partitions may differ for the DOS and Windows sessions.

- Remember that DOS may fail to access to large partitions. For instance, MS-DOS does not work with partitions that are larger than 8Gb.
- Your computer must have the ability to boot from floppy.

The detailed information about interaction between Windows-based and the diskette-based versions of Hard Disk Manager is discussed in the *Technician Manual*.

The preferred solution is to use the Paragon Recovery CD instead of using the bootable diskette.

4.3 Burn image of a partition on CD/DVD-R(W)

This chapter explains how to burn backup images of separate partitions directly on recordable and rewritable CD and DVD discs in Windows environment.

4.3.1 Overview

Hard Disk Manager supports the *burning* backup images to recordable CD/DVD media. This feature is available only in Windows environment. The function is isolated from the most generic *Backup* function because of there are some valuable specificities and limitations inhere that operation.

In Windows 95, 98 and ME this function is applicable to unlocked partitions only: Hard Disk Manager reboots into DOS to process locked partitions, but the burning hardware is generally unavailable in DOS environment.

In Windows NT, 2000 and XP Hard Disk Manager is able to burn images of both unlocked and locked partitions. In case of processing the locked partition, Hard Disk Manager reboots the system and uses the BlueScreen Component to complete the operation (see the section Backup system and locked partitions).

4.3.1.1 Media types and CD/DVD drives that are supported in the Burn image function

Hard Disk Manager uses the WinASPI service to access CD burning devices. For this reason the program can successfully use CD/DVD drives that are supported by the operating system. The program supports any media types that are supported by the used CD/DVD burning hardware.

The advantage of the program is that it can use both mapped and unmapped workable CD/DVD burning hardware.

The primary constraints of the Hard Disk Manager are those the program require blank recordable discs, it does not support Multisession CD/DVD, it places one volume per CD, it closes the CD session so that disc cannot be added. See the section Limitations of the built-in CD/DVD burning module for more details.

4.3.2 Initiating the operation in the Windows-based version

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a partition you want to store on CD/DVD-R(W).

Select some partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. Do not select *blocks of free space* because the function is disabled for blocks of free space.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

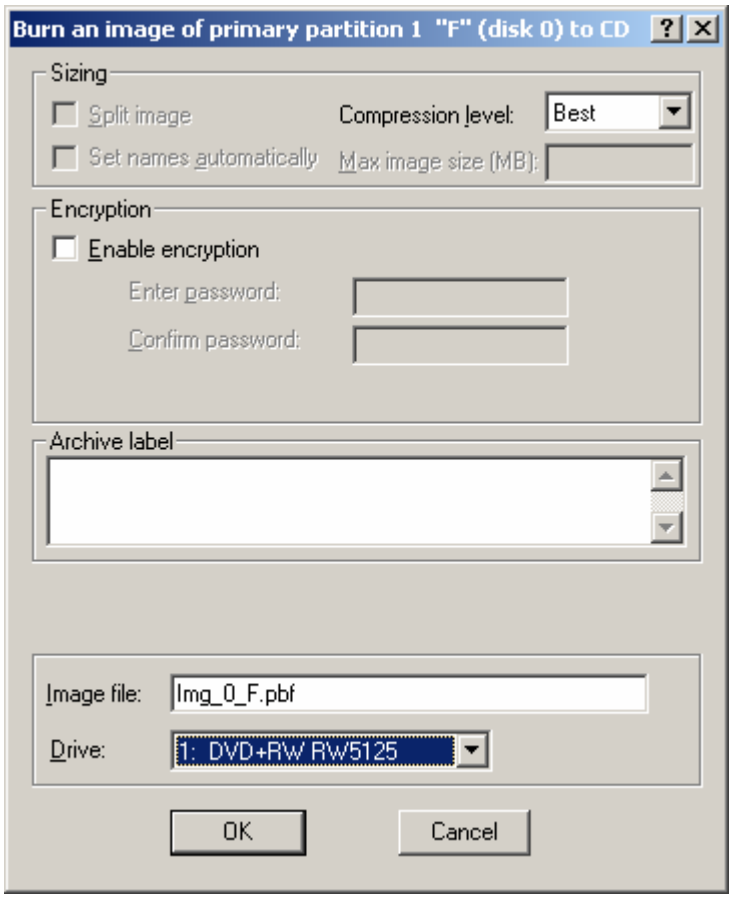
  Partition ➔ Burn an image of partition to CD

- call the *popup menu* for the selected partition and select the item:

  Burn an image of partition to CD

- Press **Burn** button on the Main Toolbar.

Step 3. Define parameters of the operation



Assign properties of the Burn image operation. You can control:

- compression level
- encryption
- archive description
- usable CD burning drive

Initially, the program suggests some consistent values for all parameters. In most cases, you just need to press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Burn Image of a Partition* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the Burn Image operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.3.3 Description of the Burn Image parameters

Parameters of the Burn Image operation are similar to the parameters of the backup operation (see Description of the Backup parameters). There are only two differences:

- The image should be located on some CD/DVD burning device
- The user cannot change the splitting parameters.

4.3.3.1 Common parameters of the Burn Image operation

Parameters of the backup operation are subdivided in four groups:

- *Encryption* settings manage the ability to cipher image contents and set the access password

- *Compression* settings manage an ability to squeeze image contents and the compression level
- *Archive label* allows to place custom notes to the image file for easy navigation through images
- *Location* settings allow selecting the CD/DVD recordable drive that will be used for burning the image.

Generally, the parameters for this operation coincide with the same parameters of the Backup Partition operation. See the chapter Description of the Backup parameters for the detailed explanation.

Encryption settings (see the chapter Description of the Backup parameters, section Encryption settings)	
Enable encryption	Set the checkmark ON to enable ciphering image contents and protect accessing to the image by the password.
Enter password **Confirm password**	With enabling the encryption, provide the password to be used to access the image. **Note:** the program does not allow "empty" passwords.
Compression settings (see the chapter Description of the Backup parameters, section Compression settings)	
Compression level	Change the compression level, in case of you wish to use the value other than the default one.
Archive label (see the chapter Description of the Backup parameters, section Archive label for more details)	
Archive label	You can associate a short descriptive text with the image. It may be helpful when selecting backup images.

4.3.3.2 Image Location

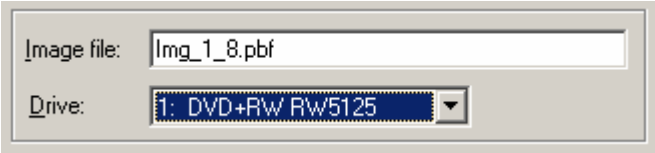


Image file

Here you can assign the desired filename of the backup image.

The program generates the default image filename by using the *disk number* and the *drive letter* (or the *partition number*) and suggests burning the image with the first registered CD/DVD burning device in the list.

Drive

You can select any available CD and DVD recordable drive from the pull-down list of CD/DVD burners.

Hard Disk Manager can use any CD burner that is accessible through WinASPI service, including devices that have no drive letters assigned. The program lists CD/DVD devices by their model names.

4.3.4 Running the Burn Image operation

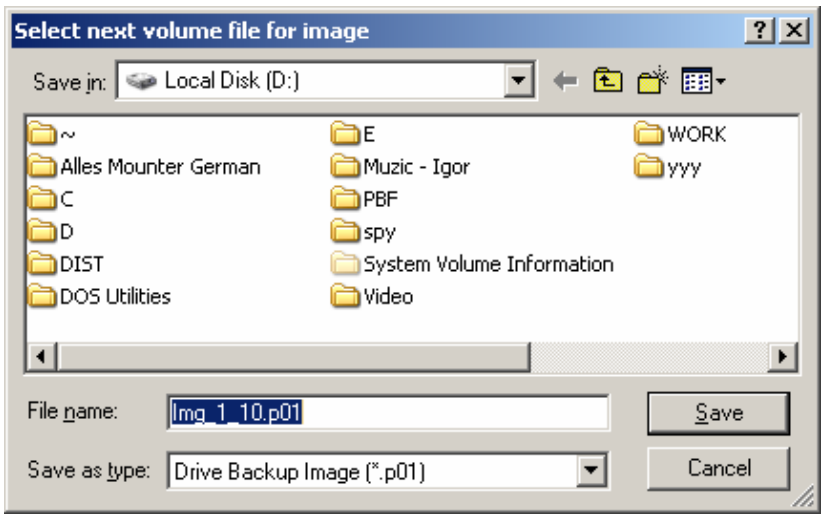
During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed and estimated remaining time for completion
- Estimated archive size and overestimated amount of CD/DVD discs required to fit the image.
- Averaged read & write speed and summarized data transfer rate

In case of a multivolumic image is created, the writing of each volume is treated as suboperation, and the program displays information about suboperation progress.

In case of a user does not allow automatic filename generation, the program pauses after completing a volume and asks a user for the next volume filename:



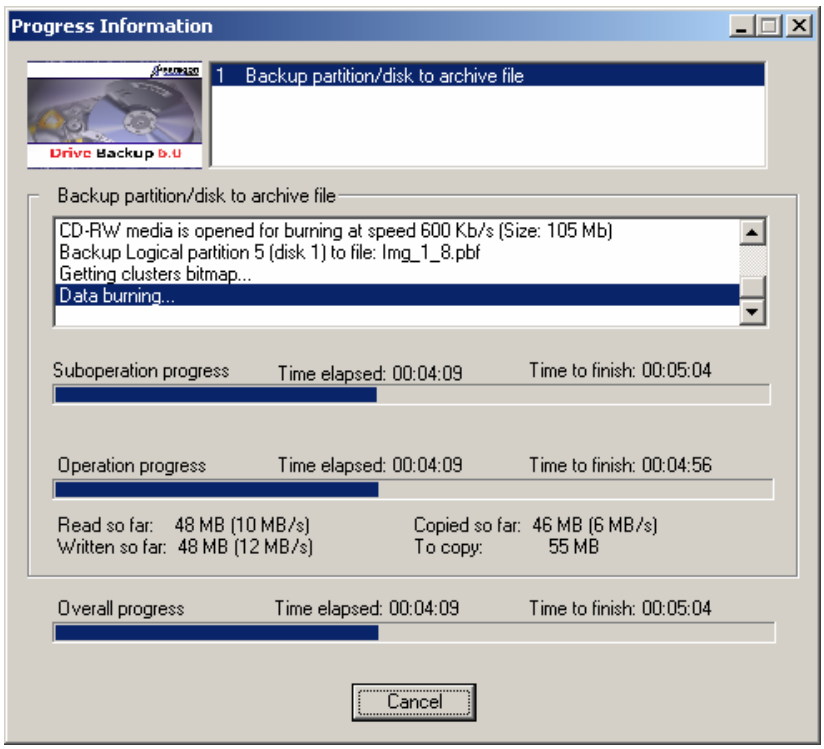
By default, Hard Disk Manager suggests the automatically generated new filename, but a user can customize this value. See the chapter How Hard Disk Manager generates filenames for multiple volumes for more details.

1. At the beginning of the real execution of the operation, Hard Disk Manager evaluates the total image size and the amount of recordable discs required to store the image. The resume is displayed in the *report console* that is located in the middle part of the window **Progress Information**.

The program makes the pessimistic evaluation of the amount of CD-R(W) discs of 700Mb capacity and DVD-R(W) of 4.7Gb discs capacity, which require to fit the backup image. Usually, this value is the overestimated one in case of using high levels of compression.

2. Next, the program starts the burning process.
   If no media is inserted in the CD/DVD drive, the program asks for inserting the new empty recordable disk.

3. Every media being inserted is tested for the cleanness.
   The non-empty once-recordable discs will be culled.
   The program allows to *erase* rewritable discs (CD/DVD-RW) that are non-empty:
   - ⇨ The *Quick Erase* assumes only the deletion of the existing formatting structures of the rewritable disc. The operation takes few minutes.
   - ⇨ The *Full Erase* assumes destroying and filling with zeroes the entire contents of the rewritable disc. The operation may take much time (dozens of minutes).



During the real execution of the operation, the **Operation Progress** window appears. The program displays the detailed statistics of the operation:

- The currently active operation is highlighted.
- Elapsed and estimated remaining times for completion of the operation are displayed.
- Averaged read speed, write speed, summarized speed of the operation are displayed.
- The complete information about suboperation progress is displayed.

4.3.5 Comments

The *BlueScreen Component* of Hard Disk Manager that is used in Windows NT, 2000 and XP supports the *Burn Image* operation. The operation is described in the chapter Burn Image of locked and system partitions.

4.3.5.1 Limitations of the built-in CD/DVD burning module

- The primary constraint to used media is that it should be blank (empty):
  - ⇨ The recordable discs (CD-R, DVD-R, DVD+R) must be blank.
  - ⇨ The rewritable discs (CD-RW, DVD+RW) that are not blank, will be erased before the use.
  - ⇨ Multisession CD/DVD are not supported.
- During the image burning, Hard Disk Manager creates the single-session, ISO-compliant disc (i.e. *Data CD* or *Data DVD*). The program makes the multivolumic backup image and places volumes on CDs, one file per disc. Every volume occupies the entire disc, the only exception of the most last volume.
- The current release of the Hard Disk Manager does not allow controlling the *write speed* of the CD/DVD burning drive. The program always uses the maximum speed available for the pair "CD burner – recordable disc".
- The current release of the Hard Disk Manager requires 64Mb of additional physical memory to initiate the CD-burner buffer. Totally, the program may require ~70Mb or more of physical memory during the *Burn Image* operation.

4.3.6 Burn Image of locked and system partitions

Generally, Hard Disk Manager needs to reboot the system in the single-tasking mode to complete the operation of *Burning an Image* of the locked partition. As it mentioned in the chapter How Hard Disk Manager executes operations, Hard Disk Manager uses the *Startup Bluescreen* service in Windows NT, 2000 and XP. In Windows 95 and 98, the program uses the *"true" DOS session*. In Windows ME, the program requires to reboot the computer from the DOS bootable diskette.

Unfortunately, Hard Disk Manager is unable to use CD/DVD burning hardware in DOS. For this reason, you are unable to use the *Burn Image* function on the locked and system partitions in

Windows 95, 98 and ME.

4.3.6.1 Burn Image of the locked partition in Windows NT, 2000 and XP

In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" service to operate the locked partitions:

1. Before starting the operation, the program checks whether partition is locked or not.
2. If the partition appears is the locked one, the program asks for reboot the system. Press **OK** button to reboot the system and complete the operation, press **Cancel** button to abort the operation.

By default, the program pauses the execution until the user makes the choice. For configuring the unattended working, see the section How to automate the system reboot for processing the locked partitions.

3. The program silently schedules the *BlueScreen Component* to run at next Windows startup and to execute the required operation.
4. Then the program reboots the computer.
5. At the next system boot-up, the Bluescreen Component executes the operation in place of the Windows-based version. The BlueScreen Component will display the operation progress in the console-like style.
6. If there is no disc inserted, the program displays the following message in the output console:

```
There is no media in CD drive. Insert new disk.
(Type exit to cancel operation.)
```

7. You should choose one of the following alternatives:
   - ⇨ insert the new recordable disc in the CD/DVD drive and press ENTER.
   - ⇨ type the word "*exit*" to abort the operation.
8. Then, Hard Disk Manager tests the disc for emptiness. In case of using rewritable media, the program suggests either to change media or erase existing data:

```
Rewritable media is not empty. Press Enter to try new media.
Type ERASE to erase media. Type EXIT to cancel operation.
```

9. You should choose one of the following alternatives:
   - ⇨ Exchange the disc with the new one and press ENTER to continue the operation.
   - ⇨ Type the word "*erase*" to erase the inserted media and continue the operation.
   - ⇨ Type the word "*exit*" to abort the operation.

Note: do not try to type in *UPPERCASE*. In fact you are able type in *lowercase* only, see the section Limitations of the BlueScreen Component for more details.

Text editing abilities in the BlueScreen Component:

- ⇨ Only printable characters and the *BackSpace* key are acceptable.
- ⇨ Other keys (including *CAPSLOCK*, *Ctrl*, *Alt*, *Shift*) are unacceptable.
- ⇨ Only English letters are available
- ⇨ All alpha letters are entered in the *lowercase*.

After the BlueScreen Component completes the operation, the Windows session begins and the Windows-based version of Hard Disk Manager comes up.

4.4 Backup & Burn the Extended Partition

4.4.1 Overview

The Extended Partition is actually the container the container of so-called *Logical Partitions* (see Glossary for more details). The fundamental feature of the Extended Partition is that it contains many partitions inside.

Hard Disk Manager provides the ability to backup the entire contents of all *logical partitions* within the single operation. This operation looks like an ordinary *backup operation* that is implemented to the Extended Partition.

Similarly, there is the ability to burn the integral backup image of all logical partitions within the single *Burn image* operation that is implemented to the Extended Partition.

Take into account that any changes in the layout of the Extended Partition and new data added to any logical partition after making the backup image, will be lost after the restoration of the

Extended Partition.

4.4.1.1 Where you can save backup images

Hard Disk Manager allows saving backup images on the following storage media:

- *Mounted* local drives and mapped network drives.
- Mounted removable media (ZIP, LS-120, MODD, ZIV etc).
- Mounted USB drives and Compact Flash cards (in Windows only).
- Any read-write media, which is mapped in the system.
- Unmounted partitions formatted to NTFS, Ext2 and Ext3.
  Hard Disk Manager uses the built-in *Universal Filesystem Driver* (UFSD) to access unmounted partitions.
- CD-R(W) and DVD-R(W) drives (in Windows only).

4.4.1.2 Restrictions

Hard Disk Manager is unable to place the image of the Extended Partition (or some of its volumes) on logical partitions that belong to the Extended Partition being imaged. One should place the image on a primary partition, on another hard disk, or on removable media.

4.4.1.3 Specific properties of images of the Extended Partition

The backup image of the Extended Partition consists of multiple volumes.

The only peculiar property of the Extended Partition image is that *it always consists of multiple files*. This feature affects on the real execution of the operations (see the section Running the operations).

In case of automatic generating of volume filenames, Hard Disk Manager uses *filename generating rules* that are described in the section How Hard Disk Manager generates filenames for multiple volumes:

- The primary file of the backup image has the filename extension **.PBF**. It contains information about the layout of logical partitions and information about volumes that are aggregated to the image of the Extended Partition.
- Logical partitions are separately stored in subordinate images. Subordinate images of logical partitions have the filename extension **.L00.** Exactly, subordinate images are valid images so that just primary volumes of them have the extension .L00 (continued with .Lnn).

For example, the user creates the image named "MyExtendedPartition" of the Extended Partition that contains three logical partitions. The image will consist of the following files:

"Single-volumic" archive	Multivolumic archive	
MyExtendedPartition.PBF MyEx0004.L00 MyEx0005.L00 MyEx0006.L00	MyExtendedPartition.PBF MyEx0004.L00 MyEx0004.L01 … MyEx0005.L00	MyEx0005.L01 … MyEx0006.L00 MyEx0006.L01 …

Switch off the automatic filename generation to manually set filenames of volumes. In this case, you are able to place volumes in different directories, on different drives and even on different media. After filling the current volume, the program will pause and ask you for the next volume name.

The side effect of the high image fragmentation is the probability of coincidence of auto-generated filenames for two different backup images. To avoid such a problem, put multivolumic images in separate directories.

4.4.2 Initiating the operations

As the matter of fact, the operation of *creating the backup image* of the Extended Partition is the same as with ordinary partitions. Similarly, the operation of *burning the backup image* of the Extended Partition to CD/DVD recordable discs is the same as with ordinary partitions.

4.4.2.1 How to Backup the Extended Partition

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Brief description of actions:

Step 1. Select the Extended Partition.

Select the Extended Partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Create an image of partition…

- Call the *popup menu* for the Extended Partition, and then select the menu item

  Create an image of partition…

- Press **Alt+B** keyboard combination
- Press **Backup** button on the Main Toolbar.

Step 3. Define parameters of the operation

The parameters of the operation are described in the section Description of the Backup parameters.

Brief description of parameters:

Parameters of the backup operation are subdivided in four groups:

- *Encryption* settings manage the ability to cipher image contents and set the access password
- *Compression* settings manage an ability to squeeze image contents and the compression level
- *Archive label* allows to place custom notes to the image file for easy navigation through images
- *Location* settings allow selecting the CD/DVD recordable drive that will be used for burning the image.

Generally, the parameters for this operation coincide with the same parameters of the Backup Partition operation. See the chapter Description of the Backup parameters for the detailed explanation.

Splitting settings (see the chapter Description of the Backup parameters, section Split settings for more details)	
Split image	Set this checkmark ON to enable automatic breaking of large images in multiple volumes. Concerning the image of the Extended Partition, this option affect on subordinate images of logical partitions.
Max image size	This value defines the maximum size of each volume of the image being created.
Set splitted file names automatically	The option forces the program to automatically generate filenames for next volumes of the image. Otherwise, the program will ask the user about the new volume filename. See Comments, section Using multivolumic images for more details.
Encryption settings (see the chapter Description of the Backup parameters, section Encryption settings)	
Enable encryption	Set the checkmark ON to enable ciphering image contents and protect accessing to the image by the password.
Enter password **Confirm password**	With enabling the encryption, provide the password to be used to access the image. **Note:** the program does not allow "empty" passwords.
Compression settings (see the chapter Description of the Backup parameters, section Compression settings)	
Compression level	Change the compression level, in case of you wish to use the value other than the default one.
Archive label (see the chapter Description of the Backup parameters, section Archive label for more details)	
Archive label	You can associate a short descriptive text with the image. It may be helpful when selecting backup images.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution for both *Backup Extended Partition* and *Burn Image of Extended Partition* operations (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the Backup operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.4.2.2 How to Burn image of the Extended Partition

The actions the user should make to initiate the operation are described in the chapter Initiating the Image Burning operation in the Windows-based version.

The parameters of the operation are described in the section Description of the Burn Image parameters.

Brief description of actions:

Step 1. Select the Extended Partition.

Select the Extended Partition in the *Tree Layout*, in the *UDP Layout* or in the *Partitions List*.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Burn an image of partition to CD

- call the *popup menu* for the selected partition and select the item:

  Burn an image of partition to CD

- Press **Burn** button on the Main Toolbar.

Step 3. Define parameters of the operation

The parameters of the operation are described in the section Description of the Burn Image parameters.

Brief description of parameters:

Parameters of the backup operation are subdivided in four groups:

- *Encryption* settings manage the ability to cipher image contents and set the access password
- *Compression* settings manage an ability to squeeze image contents and the compression level
- *Archive label* allows to place custom notes to the image file for easy navigation through images
- *Location* settings allow selecting the CD/DVD recordable drive that will be used for burning the image.

Generally, the parameters for this operation coincide with the same parameters of the Backup Partition operation. See the chapter Description of the Backup parameters for the detailed explanation.

Encryption settings (see the chapter Description of the Backup parameters, section Encryption settings)	
Enable encryption	Set the checkmark ON to enable ciphering image contents and protect accessing to the image by the password.
Enter password **Confirm password**	With enabling the encryption, provide the password to be used to access the image. **Note:** the program does not allow "empty" passwords.
Compression settings (see the chapter Description of the Backup parameters, section Compression settings)	
No compression	Set the checkmark to disable image compressing. By default, the compression is active. The default value is defined in the program settings (see General settings).
Compression level	Change the compression level, in case of you wish to use the value other than the default one.
Archive label (see the chapter Description of the Backup parameters, section Archive label for more details)	
Archive label	You can associate a short descriptive text with the image. It may be helpful when selecting backup images.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution for both the *Backup of the Extended Partition* and the *Burn Image of the Extended Partition* operations (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.

- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.4.3 Running the operations

The real execution of referring to above operations on the Extended Partition is similar to appropriate operations with ordinary partitions (see the sections Running the backup operation and Running the Burn Image operation for more details).

4.4.3.1 The Extended Partition image is the multivolumic one

The difference between images of ordinary partition and images of the Extended Partition is that Hard Disk Manager always creates the multivolumic image for the Extended Partition. The section How Hard Disk Manager generates filenames for multiple volumes describes the naming rules that Hard Disk Manager uses for auto-generated filenames.

In case of *burning the image* of the Extended Partition, the program always places the image on several discs.

The most first recordable disc will contain only the primary volume of the image, which in fact takes just several megabytes.

4.4.3.2 About processing locked Extended Partitions

The Extended Partition itself does contain neither files nor directories that can be used by an operating system or applications. However, if some of logical partitions appear locked ones, Hard Disk Manager is unable to process the Extended Partition without rebooting to the single-tasking environment.

When performing some operation on the entire Extended Partition, Hard Disk Manager checks the lock state of all logical partitions. In case of some logical partition is locked, the program reboots in the appropriate single-tasking environment and executes the operation (see How Hard Disk Manager executes operations).

The details of performance of locked partitions are described in the chapter Backup system and locked partitions.

4.4.4 Comments

4.4.4.1 How Hard Disk Manager works with logical partitions of different types

The Extended Partition may contain several logical partitions having different filesystem types. Hard Disk Manager allows creating an image of such "non-uniform" Extended Partition without any problems.

If the program works in the fast copying mode, it independently processes each logical partition. It analyzes the filesystem type and automatically switches to the appropriate copying mode. Remember that known filesystems are copied in the *fast copying* mode while the *unknown filesystems* are processed in the *sector-to-sector copying* mode.

If the program works in the sector-to-sector copying mode, it independently processes each logical partition, too. The only difference is that Hard Disk Manager saves all sectors of every existing partition.

In any of copying modes, Hard Disk Manager does not save blocks of unpartitioned space within the Extended Partition. So that the program will not save deleted logical partitions.

4.5 Backup Hard Disk

This chapter explains how to make backup images of entire hard disks on local and network drives, unmounted partitions and removable media under various conditions.

4.5.1 Overview

Hard Disk Manager provides the ability to backup not only separate partitions but also entire hard disks.

Generally, each hard disk contains the controlling records of a partitioning scheme, which is located on the disk (see Glossary), and the bootstrap code in addition to on-disk partitions. Each of mentioned above disk components is important to keep the availability of on-disk data.

Hard Disk Manager uses knowledge about the internal structure of hard disks in order to make workable images of all disk contents. In case of disk malfunction or corruption of data, the backup image of the hard disk can be used for complete restoration of the system workability.

Take into account that any changes in the layout of the hard disk and new data added to any on-disk partition after the making the backup image, will be lost after the disk restoration.

The implementation of the backup function varies in some cases:

1. In Windows, the operation differs for *unlocked* and *locked* (system) hard disks. To process locked hard disks, the program need to reboot the computer.
2. In turn, the processing of *locked disks* differs for Windows 95/98, Windows NT/2000/XP and Windows ME.

Locked hard disks are ones having locked partitions.

4.5.1.1 Where you can save backup images

Hard Disk Manager allows saving backup images on the following storage media:

- *Mounted* local drives and mapped network drives.
- Mounted removable media.
- Mounted USB drives and Compact Flash cards (in Windows only).
- Any read-write media that is mapped in the system.
- Unmounted partitions formatted to NTFS, Ext2 and Ext3. Hard Disk Manager uses the built-in *Universal Filesystem Driver* (UFSD) to access unmounted partitions.
- CD-R(W) and DVD-R(W) drives (in Windows only). This function is described in the chapter Burning the image of a hard disk.

4.5.1.2 Restrictions

Hard Disk Manager is unable to place the image of the hard disk (or some of its volumes) on any partition that belongs to the disk being imaged. One should place the image either on another hard disk or on removable media.

4.5.1.3 Specific properties of images of hard disks

Hard Disk Manager allows backup all the usable contents of the hard disk or only most important parts of the disk that contain the information about the disk layout.

The backup image of entire hard disk consists of multiple volumes.

The peculiar property of disk image is that it always consists of multiple files. This feature affects on the real execution of the operations (see the section Running the backup of the hard disk).

In case of automatic generating of volume filenames, Hard Disk Manager uses *filename generating rules* that are described in the section How Hard Disk Manager generates filenames for multiple volumes:

- The primary file of the backup image has the filename extension **.PBF**. It contains information about the layout of on-disk partitions and information about volumes that are aggregated to the image.
- *Primary partitions* are separately stored in subordinate images. Primary volumes of subordinate images of primary partitions have the filename extension **.P00**.
- The Extended Partition is stored in the subordinate image having the filename extension **.E00**.
- Logical partitions are separately stored in subordinate images. Primary volumes of subordinate images of logical partitions have the filename extension **.L00.**

For example, the user creates the image "MyDisk" of the hard disk that contains two primary and two logical partitions. The image will consist of the following files:

"Single-volumic" archive	Multivolumic archive	
MyDisk.PBF	MyDisk.PBF	MyDi0103.E00
MyDi0101.P00	MyDi0101.P00	MyDi0104.L00
MyDi0102.P00	MyDi0102.P01	MyDi0104.L01
MyDi0103.E00	…	…
MyDi0104.L00	MyDi0102.P00	MyDi0105.L00
MyDi0105.L00	MyDi0102.P01	MyEx0105.L01
	…	…

Switch off the automatic filename generation to manually set filenames of volumes. In this case, you are able to place volumes in different directories, on different drives and even on different media. After filling the current volume, the program will pause and ask you for the next volume name.

The side effect of the high image fragmentation is the probability of coincidence of auto-generated filenames for two different backup images. To avoid such a problem, put multivolumic images in separate directories.

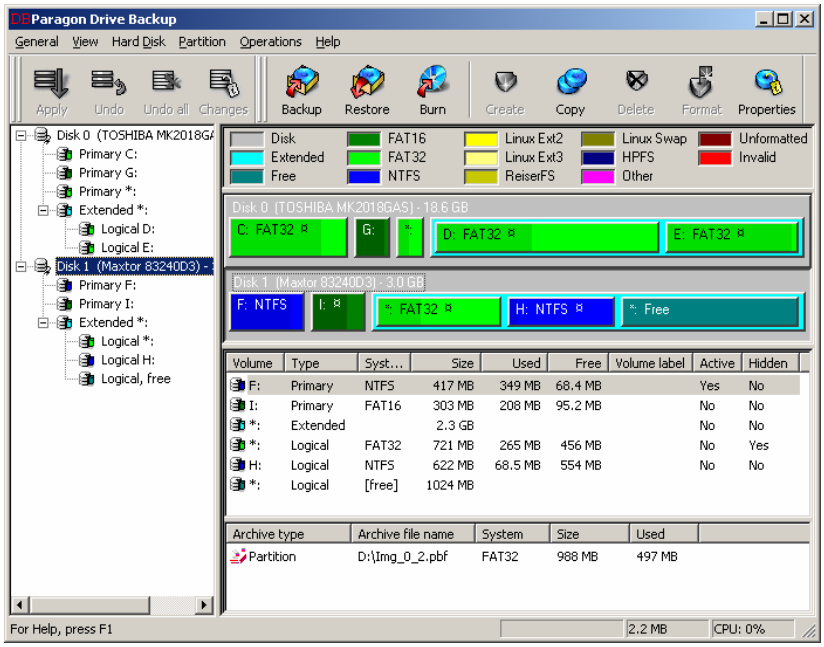
4.5.2 Initiating the operation

As the matter of fact, the operation of *creating the backup image* of the hard disk is similar to the operation of making the images of partitions.

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Brief description of actions:

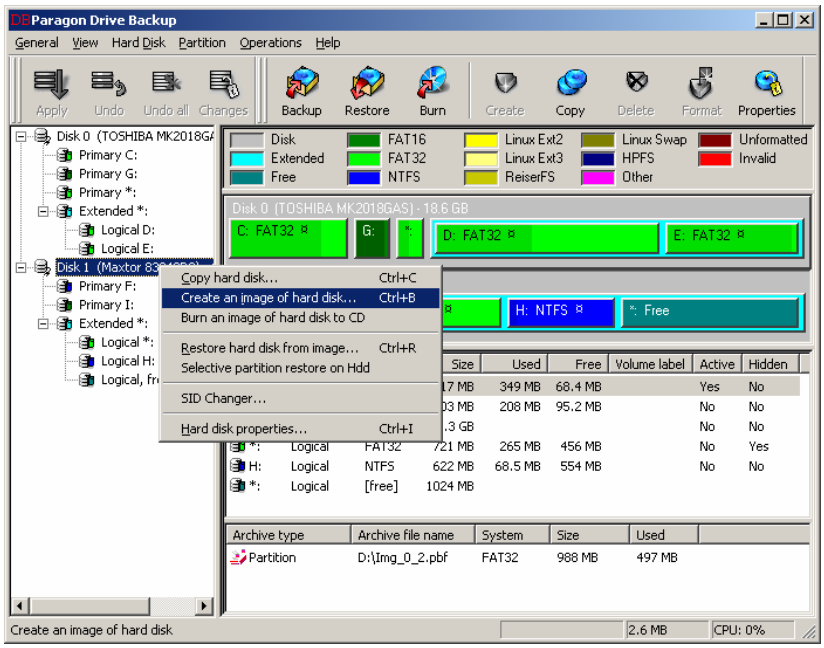
Step 1. Select the hard disk you want to backup.



There are two variants:

- select the hard disk in the Tree Layout panel or in the UDP Layout panel.
- select any partition that belongs the hard disk of interest.

Step 2. Select the operation to perform



Alternatives:

- Select in the main menu:

  Hard disk ➔ Create an image of hard disk…

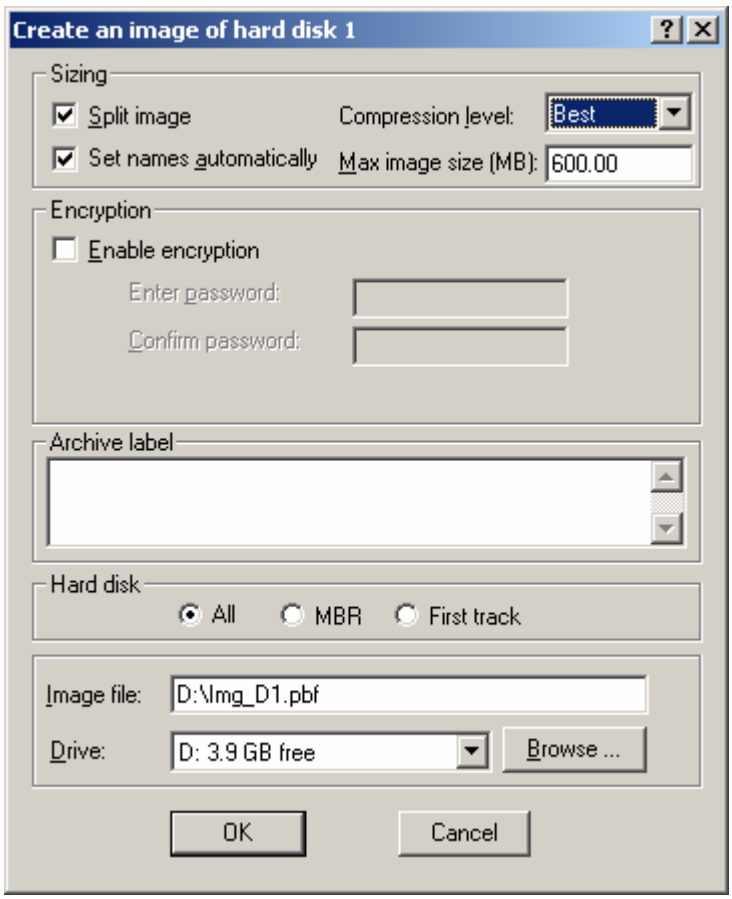
- In case of the hard disk is selected, Call the *popup menu* for the selected hard disk, and select the menu item:

  Create an image of hard disk…

- Press **Ctrl+B** keyboard combination
- Press **Backup** button on the Main Toolbar.

Step 3. Assign properties of the backup image

Generally, the parameters of the operation are similar to the same parameters of the Backup separate partitions operation.



Assign properties of the backup image. You can control:

- archive splitting
- compression level
- encryption parameters
- archive description
- image location
- the part of hard disk contents to be saved (see "Hard disk" section).

Initially the program suggests some consistent values for all parameters. In most cases, you just need to press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Backup Hard Disk* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the Backup operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.5.3 Description of parameters

Parameters of the Backup the Hard Disk operation are similar to the parameters of the Backup the Partition operation (see Description of the Backup parameters). There are only differences:

- There is the ability to backup either all contents or just bootable structures of the disk (see "Hard disk" section).
- The image must be located either on another hard disk or on removable media.

4.5.3.1 Common parameters of the Backup Hard Disk operation

Parameters of the backup operation are subdivided in six groups:

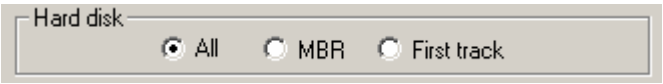
- *Split* settings manage the ability to subdivide the image in multiple volumes that can be located in different places.

- *Encryption* settings manage the ability to cipher image contents and set the access password
- *Compression* settings manage an ability to squeeze image contents and the compression level
- *Archive label* allows to place custom notes to the image file for easy navigation through images
- *Location* settings allow selecting the CD/DVD recordable drive that will be used for burning the image.
- *Hard disk* settings define the kind of on-disk data that must be stored in the image.

Generally, the parameters for this operation coincide with the same parameters of the Backup Partition operation. See the chapter Description of the Backup parameters for the detailed explanation.

Splitting settings (see the chapter Description of the Backup parameters, section Split settings for more details)	
Split image	Set this checkmark ON to enable automatic breaking of large images in multiple volumes. Concerning the image of the entire hard disk, this option affect on subordinate images of partitions.
Max image size	This value defines the maximum size of each volume of the image being created.
Set splitted file names automatically	The option forces the program to automatically generate filenames for next volumes of the image. Otherwise, the program will ask the user about the new volume filename. See Comments, section Using multivolumic images for more details.
Encryption settings (see the chapter Description of the Backup parameters, section Encryption settings)	
Enable encryption	Set the checkmark ON to enable ciphering image contents and protect accessing to the image by the password.
Enter password **Confirm password**	With enabling the encryption, provide the password to be used to access the image. **Note:** the program does not allow "empty" passwords.
Compression settings (see the chapter Description of the Backup parameters, section Compression settings)	
Compression level	Change the compression level, in case of you wish to use the value other than the default one.
Archive label (see the chapter Description of the Backup parameters, section Archive label for more details)	
Archive label	You can associate a short descriptive text with the image. It may be helpful when selecting backup images.

4.5.3.2 Hard disk section



Here one can select the part of the disk contents that will be placed to the backup image. Available alternatives are:

MBR	Hard Disk Manager will store only the Master Boot Record (MBR) of the hard disk. MBR contains the important information about the disk layout. See Glossary for more details.
First track	Hard Disk Manager will store only the 0th track of the hard disk. The 1st track may contain the executable code of the used boot manager or the disk overlay software that is essential for correct accessing contents of your hard disk. See Glossary for more details.
All	Hard Disk Manager will store all usable contents of the hard disk.

4.5.4 Running the backup of the hard disk

The real execution of the *Backup Hard Disk* operation is similar to the backup operations with partitions. The difference is that Hard Disk Manager always creates the multivolumic image for the hard disk. The section Specific properties of images of hard disks describes the naming rules that Hard Disk Manager uses for auto-generated volume filenames.

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Estimated archive size
- Averaged read & write speed and summarized data transfer rate

The program treats the creation of a subordinate archives of partitions as suboperations, and the program displays information about suboperation progress.

The operation is lengthy. Real performance fundamentally depends on hardware and an operating system being used.

4.5.5 Backup locked Hard Disks

The locked hard disks are ones containing *locked partitions* within (see Glossary). In particular, the hard disk that contains the Windows system partition is the locked one. Hard Disk Manager requires rebooting to the single-tasking environment to handle a locked partition or a locked hard disk:

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

At the beginning of the operation, Hard Disk Manager checks the lock state of all on-disk partitions. In case of any partition is locked, the program reboots in the appropriate single-tasking environment and executes the operation.

The performance of the *Backup of the Hard Disk* operation is similar to the *Backup of locked partitions* operation that is described in the chapter Backup system and locked partitions.

4.5.6 Comments

4.5.6.1 Supported partitioning schemes

Partitioning scheme is generally the format of disk partitioning information (see Glossary for more details).

The current version of Hard Disk Manager effectively supports only hard disks that use the *DOS partitioning scheme*. Still, the program can be configured to somehow support hard disks that contain other partitioning schemes.

Hard Disk Manager uses the knowledge of the structure of the DOS partitioning scheme to recognize partitions and their parameters.

4.5.6.2 How Hard Disk Manager works with partitions of different types

The hard disk may contain multiple partitions having different filesystem types. Hard Disk Manager allows creating an image of such "non-uniform" hard disk without any problems.

If the program works in the fast copying mode, it independently processes each on-disk partition. It analyzes the filesystem type and automatically switches to the appropriate copying mode: known filesystems are copied in the *fast copying* mode, and *unknown filesystems* are processed in the *sector-to-sector copying* mode.

4.5.6.3 How to make images of hard disks with unknown partitioning schemes

If Hard Disk Manager works in the *sector-to-sector copying* mode, it saves all sectors to the image of the hard disk. The program bypasses the analysis of used partitioning scheme, filesystem types and filesystem consistency of all partitions. The use of the *Compression* function will reduce the image size.

How to backup hard disks with unknown partitioning schemes:

1. Switch the program to the *sector-to-sector copying* mode:
   General ➔ Settings ➔ (page) General ➔ Copy all sectors 1:1
2. Initiate the *Backup Hard Disk* operation
   Hard Disk ➔ Create an image of hard disk…

4.5.6.4 Contents of the disk image

Hard Disk Manager uses the knowledge of the structure of the DOS partitioning scheme to recognize partitions and their parameters. In case of all contents are selected to be imaged, the program stores the entire 1st track of the hard disk, the complete information about the layout of all partitions and all contents of all partitions.

Partitions of unknown types are saved in the *sector-to-sector* copying mode. Partitions of known filesystem types are stored in the fast copying mode.

With each partition, Hard Disk Manager completely stores the bootable code (if exists), the structure of directories, contents of all files, information about files allocation, all file attributes including security information, encryption, compression and access quotas (if available).

4.6 Burn the image of a hard disk on CD/DVD-R(W)

This chapter explains how to burn backup images of hard disks directly on recordable and rewritable CD and DVD discs in Windows environment.

4.6.1 Overview

Generally, this function is similar to the function Burning images of partitions on CD/DVD-R(W). Hard Disk Manager supports the *burning* backup images to recordable CD/DVD media. This feature is available only in Windows environment.

Note that in Windows 95, 98 and ME this function is not practically applicable to the system and locked hard disks. The reason is that Hard Disk Manager requires to reboot into DOS to process locked partitions in Windows 95/98/ME, but the burning hardware is generally unavailable in DOS environment.

In Windows NT, 2000 and XP Hard Disk Manager is able to burn images of both unlocked and locked hard disks. In case of processing the system or locked hard disk, Hard Disk Manager uses the BlueScreen Component to complete the operation on the locked hard disks (see the section Backup locked Hard Disks).

4.6.1.1 Supported Media types and CD/DVD drives

Hard Disk Manager uses the WinASPI service to access CD burning devices. For this reason the program can successfully use CD/DVD drives that are supported by the operating system. The program supports any media types that are supported by the used CD/DVD burning hardware.

The advantage of the program is that it can use both mapped and unmapped workable CD/DVD burning hardware.

The primary constraints of the Hard Disk Manager are those the program require blank recordable discs, it does not support Multisession CD/DVD, it places one volume per CD, it closes the CD session so that disc cannot be added. See the section Limitations of the built-in CD/DVD burning module for more details.

4.6.2 Initiating the operation in the Windows-based version

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select the hard disk you want to store on CD/DVD-R(W).

There are two variants:

- select the hard disk in the Tree Layout panel or in the UDP Layout panel.
- select any partition that belongs the hard disk of interest.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

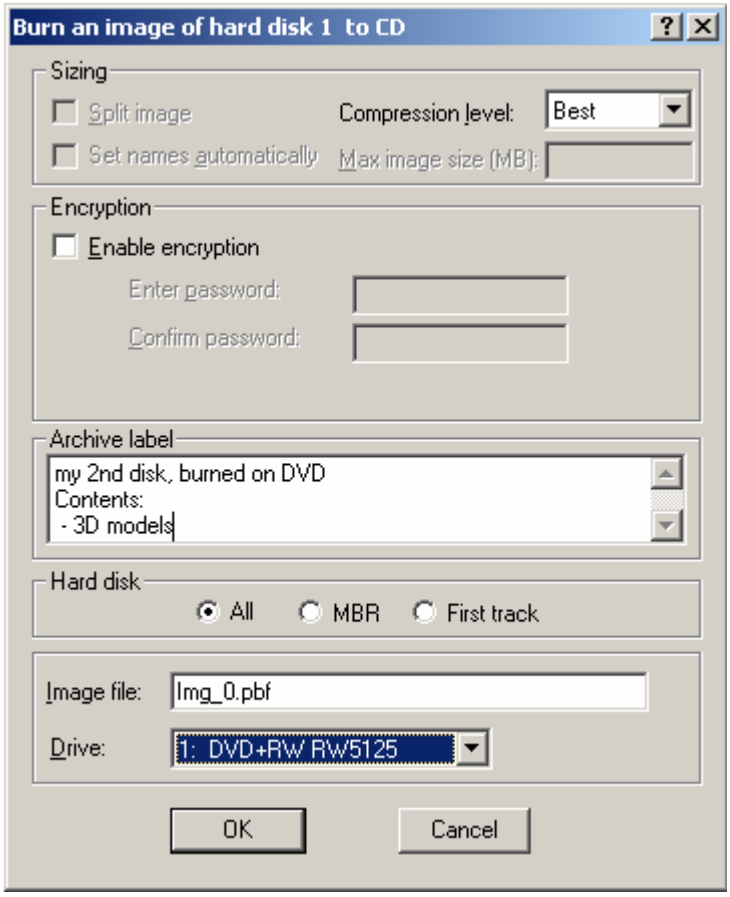
  Hard disk ➔ Burn an image of hard disk to CD

- In case of the hard disk is selected, call the *popup menu* for the selected partition in any of layout panels (click right mouse button)
  then select the menu item

  Burn an image of hard disk to CD

- In case of the hard disk is selected, press **Burn** button on the Main Toolbar.

Step 3. Define parameters of the operation



Assign properties of the Burn image operation. You can control:

- compression level
- encryption
- archive description
- usable CD burning drive
- the part of contents disk to be saved

Initially the program suggests some consistent values for all parameters. In most cases, you just need to press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Burn an image of Hard Disk* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.6.3 Description of the Burn Image parameters

Parameters of the Burn Image operation are similar to the parameters of the backup operation (see Description of parameters). There are only two differences:

- The image should be located on some CD/DVD burning device
- The user cannot change the splitting parameters.

4.6.3.1 Common parameters of the Burn Image operation

The general parameters of the Burn Image operation are described in details in the chapter Burning images of partitions on CD/DVD-R(W), section Description of the Burn Image parameters.

Parameters of the backup operation are subdivided in five groups:

- *Encryption* settings manage the ability to cipher image contents and set the access password
- *Compression* settings manage an ability to squeeze image contents and the compression level
- *Achieve label* allows to place custom notes to the image file for easy navigation through images
- *Location* settings allow selecting the CD/DVD recordable drive that will be used for burning the image.
- *Hard disk* section reveals the part of on-disk data to be stored in the image.

Generally, the parameters for this operation coincide with the same parameters of the Backup Partition operation. See the chapter Description of the Backup parameters for the detailed explanation.

Encryption settings (see the chapter Description of the Backup parameters, section Encryption settings)	
Enable encryption	Set the checkmark ON to enable ciphering image contents and protect accessing to the image by the password.
Enter password **Confirm password**	With enabling the encryption, provide the password to be used to access the image. **Note:** the program does not allow "empty" passwords.
Compression settings (see the chapter Description of the Backup parameters, section Compression settings)	
Compression level	Change the compression level, in case of you wish to use the value other than the default one.
Achieve label (see the chapter Description of the Backup parameters, section Achieve label for more details)	
Achieve label	You can associate a short descriptive text with the image. It may be helpful when selecting backup images.
Hard disk (see the chapter Description of parameters, section Hard disk section for more details)	
All	Save all contents of the hard disk.
MBR	Save only the Master Boot Record of the hard disk.
First track	Save only the 0th track of the hard disk.
Image location (see the chapter Description of the Burn Image parameters, section Image Location for more details).	
Image file	Enter the filename of the backup image. The default filename contains the *disk number* and the *drive letter* (or the *partition number*).
Drive	Select the recordable CD drive that should be used to burn discs. The pull-down list contains available CD/DVD-R(W) devices.

4.6.4 Running the Burn Image operation

Generally, the performance of the *Burn Image of hard disk* operation is same as the performance of the Burning images of partitions operation.

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed and estimated remaining time for completion
- Estimated archive size and overestimated amount of CD/DVD discs required to fit the image.
- Averaged read & write speed and summarized data transfer rate

In case of a multivolumic image is created, the writing of each volume is treated as suboperation, and the program displays information about suboperation progress.

1. At the beginning of the real execution of the operation, Hard Disk Manager evaluates roughly the total image size and the amount of recordable discs required to store the image. The resume is displayed in the *report console* that is located in the middle part of the window **Progress Information**.

The program makes the pessimistic evaluation of the amount of CD-R(W) discs of 700Mb capacity and DVD-R(W) of 4.7Gb discs capacity, which require to fit the backup image. Usually, this value is the overestimated one in case of using high levels of compression.

2. Next, the program starts the burning process.
   If no media is inserted in the CD/DVD drive, the program asks for inserting the new empty recordable disk.
3. Every media being inserted is tested for the cleanness. The non-empty once-recordable discs will be culled. The program allows to *erase* non-empty rewritable discs (CD/DVD-RW):
   ⇨ The *Quick Erase* assumes only the deletion of the existing formatting structures of the rewritable disc. The operation takes few minutes.

⇨ The *Full Erase* assumes destroying and filling with zeroes the entire contents of the rewritable disc. The operation may take much time (dozens of minutes).

During the real execution of the operation, the **Operation Progress** window appears. The program displays the detailed statistics of the operation:

- The currently active operation is highlighted.
- Elapsed and estimated remaining time for completion of the operation.
- Averaged read speed, write speed, summarized speed of the operation.
- The complete information about suboperations progress.

4.6.5 Comments

The *BlueScreen Component* of Hard Disk Manager that is used in Windows NT, 2000 and XP supports the *Burn Image of hard disk* operation. The operation is described in the chapter Burn Image of locked and system hard disks.

4.6.5.1 Limitations of the built-in CD/DVD burning module

The limitations of the current version of Hard Disk Manager are described in the chapter Burning images of partitions on CD/DVD-R(W), section Limitations of the built-in CD/DVD burning module.

4.6.6 Burn Image of locked and system hard disks

Generally, Hard Disk Manager needs to reboot the computer in the single-tasking mode to complete the operation of *Burning an Image* of the locked or system hard disk. As it mentioned in the chapter How Hard Disk Manager executes operations, Hard Disk Manager uses the *Startup Bluescreen* service in Windows NT, 2000 and XP. In Windows 95 and 98, the program uses the *"true" DOS session*. In Windows ME, the program requires to reboot the computer from the DOS bootable diskette.

Unfortunately, Hard Disk Manager cannot use CD/DVD burning hardware in DOS. For this reason, you are unable to use the *Burn Image* function for the locked and system hard disks in Windows 95, 98 and ME.

4.6.6.1 Burn Image of the locked hard disk in Windows NT, 2000 and XP

In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called *startup Bluescreen* service to operate the locked hard disks. Generally, the execution of the operation is very similar to the burning an image of the Extended Partition (see About processing locked Extended Partitions).

1. Before starting the operation, the program checks the lock state of all on-disk partitions.
2. If any partition on the disk appears locked, the program asks for reboot the computer. Press **OK** button to reboot the system and complete the operation, or press **Cancel** button to abort the operation.

By default, the program pauses the execution until the user makes the choice. For configuring the unattended working, see the section How to automate the system reboot for processing the locked partitions.

3. The program silently schedules the *BlueScreen Component* to run at next Windows startup and to execute the required operation.
4. Then the program reboots the computer.
5. At the next system boot-up, the Bluescreen Component executes the operation in place of the Windows-based version.
6. If there is no disc inserted, the program displays the following message in the output console:

```
There is no media in CD drive. Insert new disk.
(Type exit to cancel operation.)
```

7. You should choose one of the following actions:
   ⇨ insert the new recordable disc in the CD/DVD drive and press ENTER.
   ⇨ type the word "*exit*" to abort the operation.
8. Then Hard Disk Manager tests the disc for emptiness. In case of using rewritable media, the program suggests either to change media or erase existing data:

```
Rewritable media is not empty. Press Enter to try new media.
Type ERASE to erase media. Type EXIT to cancel operation.
```

9. You should choose one of the following actions:
   ⇨ exchange the disc with the new one and press ENTER to continue the operation.
   ⇨ type the word "*erase*" to erase the inserted media and continue the operation.

⇨ type the word "*exit*" to abort the operation.

Text editing abilities in the BlueScreen Component:

⇨ Only printable characters and the *BackSpace* key are acceptable.
⇨ Other keys (including *CAPSLOCK*, *Ctrl*, *Alt*, *Shift*) are unacceptable.
⇨ Only English letters are available
⇨ All alpha letters are entered in the *lowercase*.

After the BlueScreen Component completes the operation, the Windows session begins and the Windows-based version of Hard Disk Manager comes up.

4.7 Restore Partition from Backup Image

This chapter explains how to restore data from previously made backup images under various conditions.

4.7.1 Overview

Restoration from image is another primary function of the Hard Disk Manager that is complementary to the *Backup* operation (see Backup separate partitions).

The backup image includes contents of all files and *filesystem metadata*. The program keeps all information associated with files including the exact structure of directories, location of files on the disk, security information, access quotas and so on. After the restoration, the partition becomes in the pre-backup state.

Take into account that new data added to the partition after making the backup image, will be lost after the partition restoration.

The implementation of this function varies in some cases:

1. In Windows, the operation differs for *unlocked* and *locked* (system) partitions. To process locked partitions, the program need to reboot the computer.
2. The processing of *locked partitions* differs for Windows 95/98, Windows NT/2000/XP and Windows ME.

4.7.1.1 Where backup images can be taken from

Hard Disk Manager can restore partitions from backup images stored on the following media (see also the chapter Backup separate partitions, section Where you can save backup images):

- Mounted local drives and mapped network drives.
- CD/DVD-ROM drives. The equipment should be available through the system drivers.
- Mounted removable media including USB drives and Compact Flash cards (in Windows only).
- Any readable media that is mapped in the system.
- Unmounted partitions formatted to NTFS, Ext2 and Ext3.
  Hard Disk Manager uses the built-in *Universal Filesystem Driver* (UFSD) to access unmounted partitions.

4.7.1.2 Restrictions

1. Hard Disk Manager allows restoring the image over the existing partition. In this case, the following constraint should be taken into account: Hard Disk Manager is unable to use the image that is located (or holds some of its volumes) on the partition being restored.
2. The program does not allow restoring the image of the hard disk over the partition or over the block of free space.

To inspect the image type and the image contents, use the function Show Archive Info:

(menu) General ➔Show archive info…

4.7.2 Initiating the operation

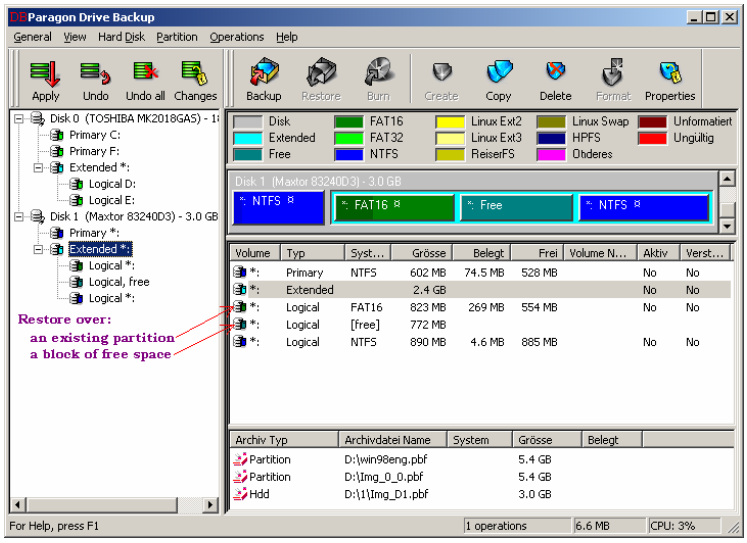
The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Previous versions of Hard Disk Manager traditionally expect that a user chooses a partition or a block of free space and then selects an operation for a chosen object.

Current version of Hard Disk Manager allows also initiating the *Restore* function as an operation over a backup image. Sections below describe both procedures.

4.7.2.1 A traditional procedure of restoration over a partition or free space

Step 1. Select a place for the partition to be restored.



Select some partition or a block of free space in the Tree Layout panel, or in the UDP Layout panel, or in the List of Partitions.

If the *block of free space* is selected, Hard Disk Manager can fit the restored partition within the bounds of the selected block only.

If the *partition* is selected, Hard Disk Manager can fit the restored partition within the bounds of the selected partition plus right and left adjacent blocks of free space.

See the section How Hard Disk Manager evaluates space available for restoration of the partition for more details.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Restore partition from image…

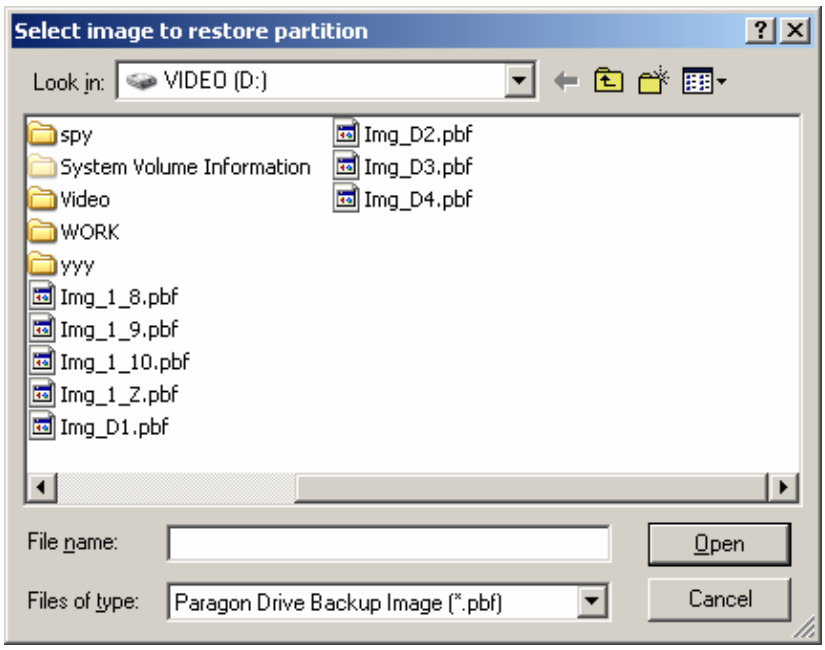
- Call the *popup menu* for the selected partition, then select the menu item:

  Restore partition from image…

- Press Alt+R key combination
- Press **Restore** button on the Main Toolbar.

Step 3. Select an image that contains the required partition

Just after calling the *Restore operation* the *Open File* dialog appears. Select some backup image to restore the partition from:



You are able to select any file from any *mounted local drive* or *mapped network drive*. In addition, you can select files on *unmounted partitions* on local hard drives (see the section Selecting unmounted partitions as the target location for saving backup images for more details).

In case of the image of hard disk is selected, the program generates the error message about the inappropriate image type (see Restrictions section). Use the Selective partition restore function to restore a single partition from the image of hard disk or the image of the Extended Partition. You can inspect the type and contents of the image chosen with using the Show Archive Info function.

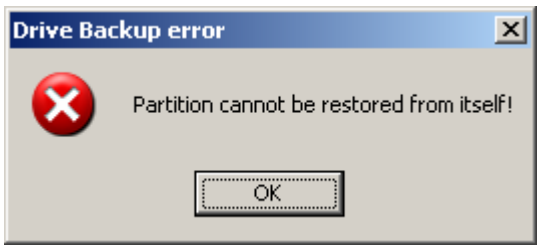
By default, the program displays only .PBF-files that are the primary files of the referential backup archives. Generally, there is the ability to restore selected partitions from the *multi-partition archives* (such as hard disk images and Extended Partition images). The preferred way is to use the function Selective partition restore. However, you are able to use the referring to above *Restore partition* function for this purpose:

1. Change the *filtered files type* in the **Files of type** pull-down list from "**Paragon** Hard Disk Manager **Image (*.pbf)**" to "**All files (*.*)**". The list of files should display all files.
2. Select files having the type (i.e. *file extension*) **.P00**, **.L00**, **.E00**. These files correspond to partitions that are included to the multi-partition archives:
   - ⇨ **.P00** refer to the stored *primary partition*.
   - ⇨ **.L00** refer to the stored *logical partition*.
   - ⇨ **.E00** refer to the stored *Extended partition*.

Step 4. Define more accurately the future location and size of the restored partition

1. First, Hard Disk Manager checks the image type. In this function, the program admits only images of single partitions and images of Extended Partitions. In case of incorrect image type, the error message appears:

2. Next, Hard Disk Manager checks that the selected image does not contain any of its volumes on the targeted hard disk. The error message appears in case of violation this restriction:

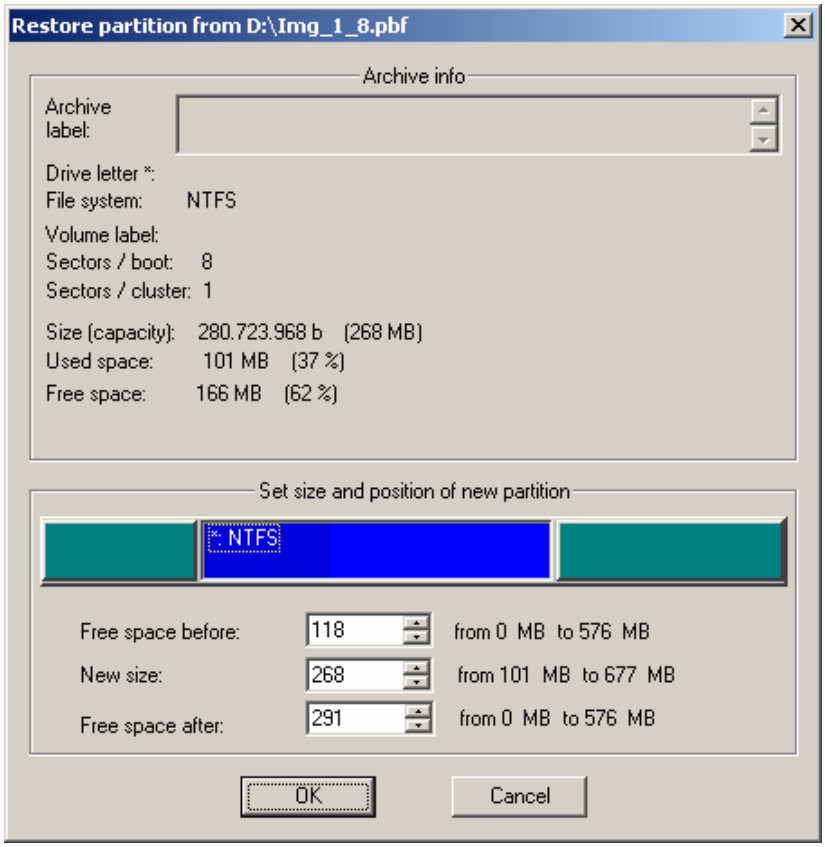


In this situation, one should either select another image or move volumes of the image to another partition, another hard disk or to removable media.

3. Next, Hard Disk Manager evaluates disk space available for restoration of the selected image (see How Hard Disk Manager evaluates space available for restoration of the partition). In case of the selected location and selected partition do not meet the requirements, Hard Disk Manager displays the error message:

You should select another image to restore on the selected partition.

4. Next, the program displays the *Restore partition* dialog. This window displays parameters of a partition being saved in the selected backup image.



In the **Archive info** section, the program displays: filesystem type, capacity, used & free space, archive label.

In the **Set size and position of new partition** section, the program displays the future location of the restored partition. Here, you are able to change the final location and size of the restored partition.

The functionality available in the *Restore partition* dialog is described in the section Defining the location and size of the restored partition. See also the section How Hard Disk Manager evaluates space available for restoration of the partition for more details.

Step 5. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Restore Partition* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.7.2.2 An alternative procedure of restoration of an image

An alternative way to initiate the function is to perform the Restore operation over a selected backup image. The program keeps the database of recently used backup images (see the chapter List of Backup Images).

Step 1. Select a backup image to be restored

Select some image of a partition in the List of Backup Images. The program displays a type of an image in the first column.

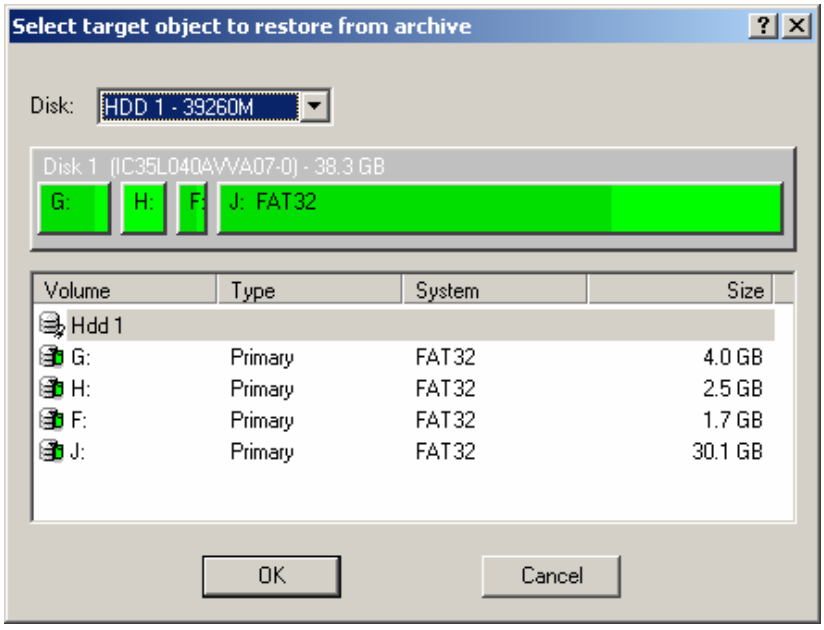
Step 2. Select the operation to perform

Call the *popup menu* for the selected image, and select the menu item:

Restore archive…

Step 3. Select an image that should be restored

The program displays the window "**Select target object to restore from archive**". This dialog allows selecting destination hard disk and partition for the restoration operation:



The **Disk** pull-down list allows selecting a hard disk where the image should be restored.

The *UDP control* placed below displays the current layout of the selected hard disk; it can be used for selecting a desired location for the restored partition. The *List of Partitions* on the bottom duplicates this functionality.

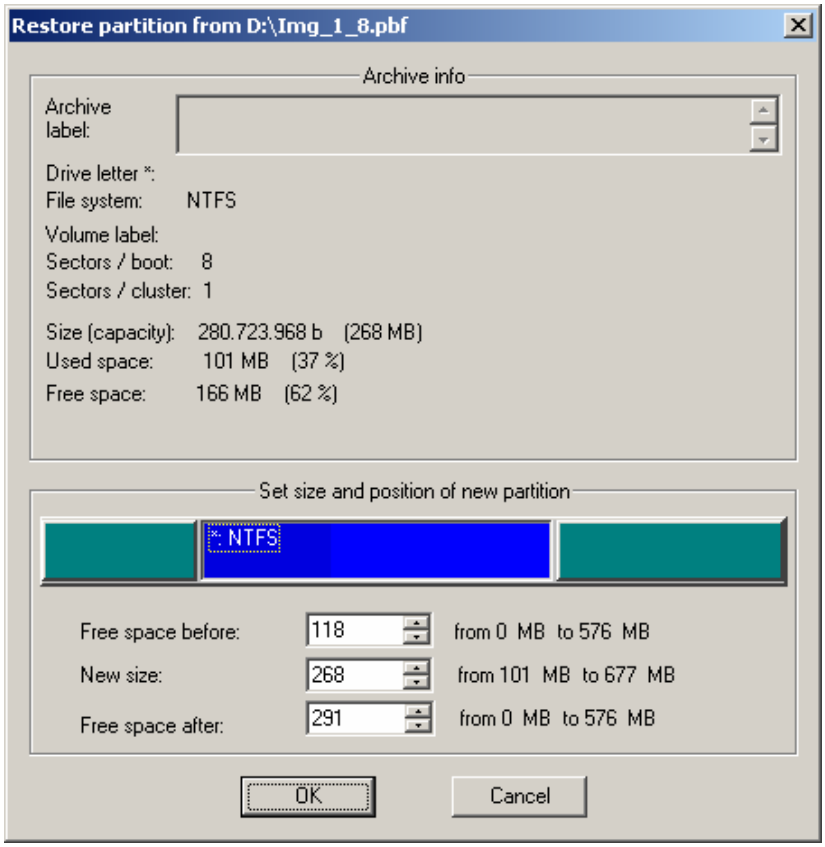
After selecting a location, press OK button to continue restoration.

Step 4. Define more accurately the future location and size of the restored partition

Step 5. Apply the operation

Last two steps are the same as in the first case. See the previous section A traditional procedure of restoration for the detailed explanation.

4.7.3 Defining the location and size of the restored partition



The *Restore partition* dialog provides the ability to fully control the resulting position and the size of the restored partition. The program determines the range of disk space available for placing the restored partition (see How Hard Disk Manager evaluates space available for restoration of the partition). Briefly:

- In case of a *block of free space* was targeted, Hard Disk Manager allows to locate the restored partition within the bounds of the selected block of free space.
- In case of a *partition* was targeted, Hard Disk Manager allows to locate the restored partition over the partition and the enveloping blocks of free space. In this case, the initial location of the restored partition coincides with the location of the targeted one.

In any case, the capacity of the restored partition must not exceed the size of the selected range of disk space.

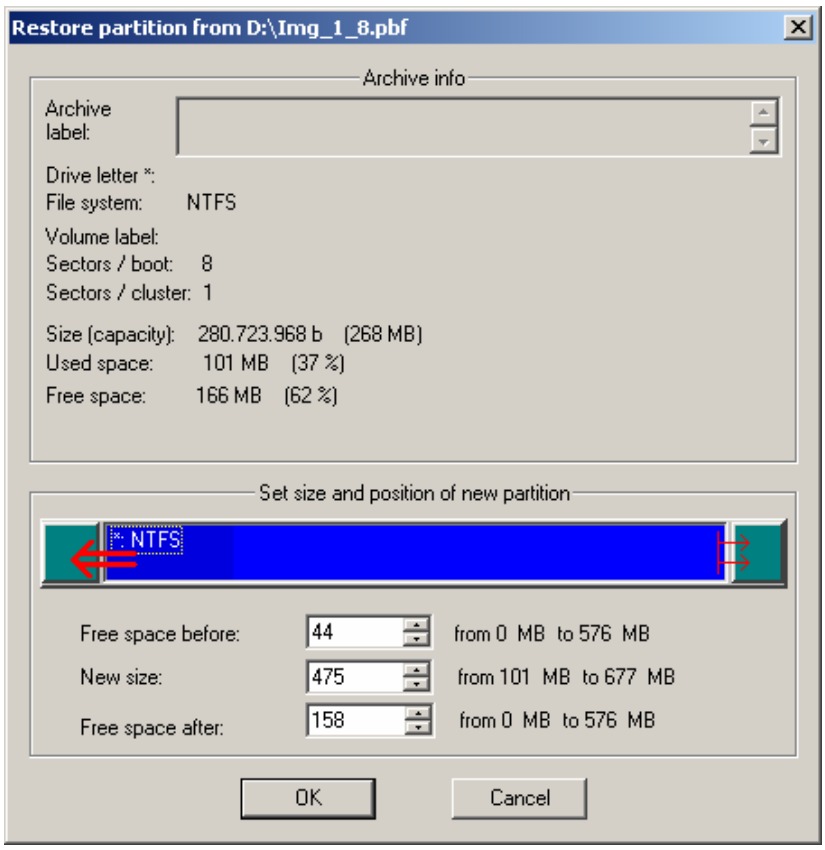
Free Space Before
New Size
Free Space After

You can move the restored partition and change its size either by using the UDP control that is located in the window or by using three *spinner* controls placed on the bottom:

- The **Free space before** spinner control defines the position (in Mb) of the restored partition relative to the beginning of the available range of disk space.
- The **New size** spinner control defines the size (in Mb) of the restored partition.
- The **Free space after** spinner control defines the amount of trailing free space (in Mb) at the end of the available range of disk space.

The *restrictions* the program takes into account are those:

- The partition must totally fit in the range of available disk space.
- The partition size must be greater than the amount of used space.



The UDP control and spin controls are synchronized, the changing of any of these elements affects on all other ones.

How the spin controls behave:

Free space before	Moves the beginning of the partition (left edge), preferably with keeping the partition size.
New Size	Changes the size of the partition, preferably with keeping the starting position (left edge).
Free space before	Moves the end of the partition (right edge). On increasing the value, it (preferably) keeps the partition size. On decreasing the value, it (preferably) keeps the starting position (left edge) so that the partition expands.

There are rules that take effect in the partition resizing. See the section Comments for more details.

4.7.4 Running the restore operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed and summarized data transfer rate

The restoration of each archive volume is treated as suboperation, and the program displays information about suboperation progress.

In case of a multivolumic image is used, the program may pause and display the *Open File* dialog to ask the user for the next volume filename and location.

Note: this is not the usual behavior of Hard Disk Manager. The program pauses the restoration only in the following cases:

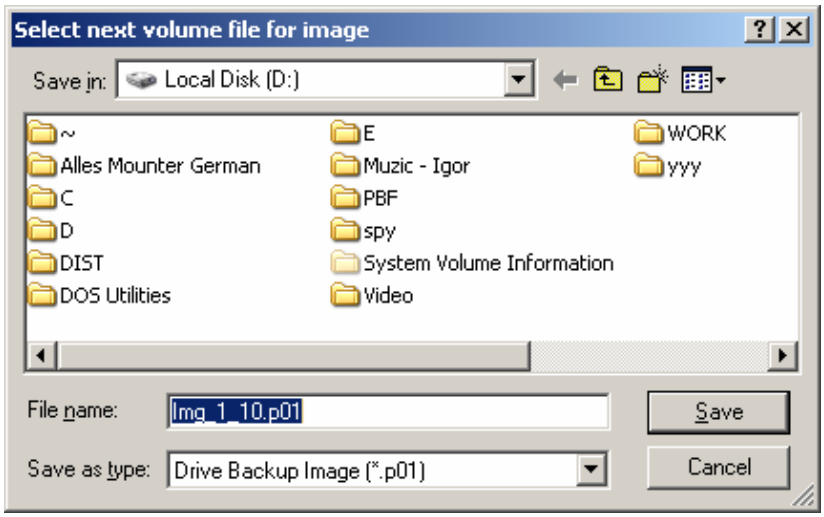
⇨ some volumes of the image were moved or renamed
⇨ volumes are located on removable media

4.7.4.1 Restoring multivolumic archives

Multivolumic images consist of multiple files, which can be placed in different directories, on different disks and removable media.

When creating an image, the program saves information about location of volume of an image. Alternatively, during the restoration from a multivolumic image, the program searches volumes on their original location for completing the process without user's intervention.

In case of some files were renamed or moved to another location, the program pauses working and displays the *Open File* dialog in order to a user points next volume location manually:



Note: this is not a usual behavior of Hard Disk Manager. The program pauses the restoration only in the following cases:

- ⇨ some volumes of the image were moved or renamed
- ⇨ volumes are located on removable media

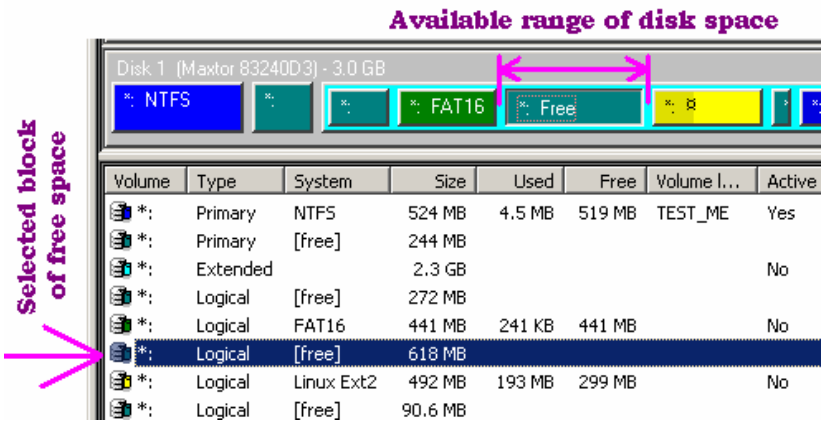
The program will automatically restore multivolumic image in "unattended" fashion, in the following cases:

- All volumes of the image reside on their original locations, and all of them are placed on local hard disk(s).
- All volumes of the image are picked up in the same directory on a local hard disk.

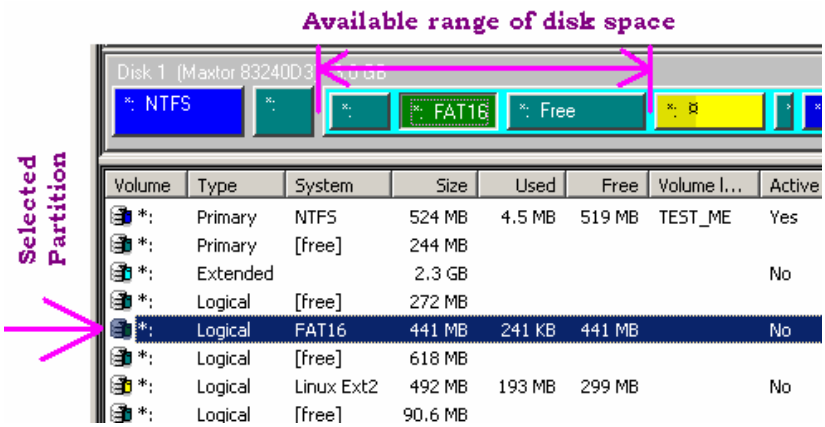
4.7.5 Comments

4.7.5.1 How Hard Disk Manager evaluates space available for restoration of the partition

The *Restore partition* dialog provides the ability to fully control the resulting position and the size of the restored partition. The program determines the range of disk space available for placing the restored partition:



1. In case of a *block of free space* was aimed for placing the restored partition, Hard Disk Manager allows to locate the target partition within the bounds of the selected block of free space only. The capacity of the restored partition must not exceed the size of the selected block of free space. In addition, in this case the program selects the beginning of the free block as the initial location of the restored partition.



2. In case of an *existing partition* was aimed, Hard Disk Manager includes to the available range:
   - ⇨ the selected partition
   - ⇨ the left adjacent block of free space (if exists)
   - ⇨ and the right adjacent block of free space (if exists).

   The capacity of the restored partition must not exceed the size of the summary range of disk space. The initial location of the restored partition coincides with the beginning of the selected partition (if the restored partition can fit to the range). Otherwise, Hard Disk Manager moves forward the initial location.
3. The range of available disk space should be greater than the initial size of the restored partition. It is the *primary condition* of the ability of restoration.

4.7.5.2 How to avoid bad sectors during the restoration

In case of restoring data to the bad sectors, the resulting partition may become corrupted. To avoid loosing data because of bad sectors, select the *surface test* to the value other than **None**.

(menu) General ➔ Settings… ➔ (page) General ➔ Surface test

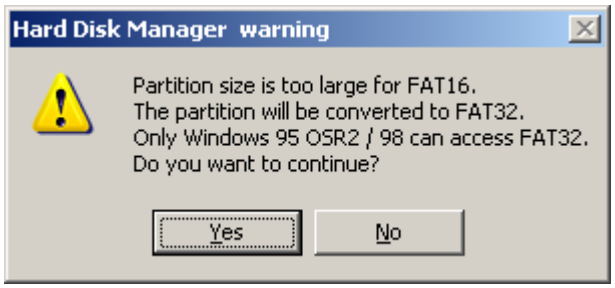
In this case, the program performs the surface test prior to the data restoration over the range of disk space that will be occupied by the restored partition. In case of detecting bad or unreliable sectors, the program will mark them *bad* and will not place data to those sectors.

The additional surface test noticeably increases the overall time required to complete the operation, also the algorithm that bypasses bad sectors significantly slows down the performance. Use this feature only in the case you suspect that there are bad sectors in the location you have chosen for the partition restoration.

4.7.5.3 Converting FAT16 to FAT32 during the automatic resizing

FAT16 partitions are limited to the value of 2Gb (see the section Converting FAT16 to FAT32 during the automatic resizing for more details).

Nevertheless, Hard Disk Manager allows expanding FAT16 partition over the 2Gb boundary by converting FAT16 filesystem to FAT32. In this case, the program displays the following warning message:



There is the ability to inhibit this warning (see the chapter Settings overview, section General Page):

(menu) General ➔ Settings… ➔ (page) General ➔ (checkmark) Convert FAT16 to FAT32 automatically

In this case, the program automatically converts FAT16 partitions to FAT32 without warnings.

4.7.6 Restoration of locked partitions

Hard Disk Manager allows restoring the image over the existing partition. As the matter of fact, Hard Disk Manager deletes the existing partition prior to the real restoration of data. Still, there could be a situation when the targeted partition is modified by another software, e.g. some file is edited. In this case, Hard Disk Manager is unable delete partition or uniting the old partition structure being restored with the new data.

To avoid the damage of data consistency, Hard Disk Manager reboots the system to a single-tasking environment in order to eliminate the interference of other programs.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

4.7.6.1 Restoration over locked partitions in Windows NT, 2000 and XP

In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase to operate the locked partitions:

1. Before starting the operation, the program checks whether partition is locked or not.
2. If the partition appears is the locked one, the program asks for reboot the system. Press **OK** button to reboot the system and complete the operation, press **Cancel** button to abort the operation.

By default, the program pauses the execution until the user makes the choice. For configuring the unattended working, see the section How to automate the system reboot for processing the locked partitions.

3. The program silently schedules the *BlueScreen Component* to run at next Windows startup and to execute the required operation.
4. Then the program reboots the computer.
5. At the next system boot-up, the Bluescreen Component executes the operation in place of the Windows-based version. The BlueScreen Component will display the operation progress in the console-like style.
6. In case of using the multivolumic image having some volumes:
   - ⇨ moved to another location
   - ⇨ placed on removable media (such as CD/DVD-ROM)

   the program may fail to find automatically the sequel of the multivolumic image. In this case, the program asks the user for the filename of the next volume. The most bottom lines of the console output will be the following:

```
Get new filename of the file for subsequent reading.
(Type exit to cancel operation.)
D:\Img_1_8.p01
```

7. You should choose one of the following actions::
   - ⇨ change the filename to proceed with another one.
   - ⇨ clear the filename and type the word "*exit*" to abort the operation.
8. By default, Hard Disk Manager suggests the filename that is registered in the archive. The problem is that the program has failed to open the volume (otherwise it would continue working without pausing). At this moment, the user must *guess right* the filename to continue with the restoration.
9. After the BlueScreen Component completes the operation, the Windows session begins and the Windows-based version of Hard Disk Manager comes up.

The text editing abilities in the BlueScreen Component:

See the section The text editing abilities in the BlueScreen Component.

4.7.6.2 Limitations of the BlueScreen Component

There are some minor functionality limitations of the BlueScreen Component that come from the inaccessibility of some Windows services during the Startup Bluescreen phase:

- The version of the BlueScreen Component provided with Hard Disk Manager interacts with the user in the console-like style. Such useful functions like *Browse disk contents* and *Search files* are not available. In case of the program asks a user to enter a filename, the user must enter a filename "blindly".
- The BlueScreen Component allows input only English letters.
  The text is in *lowercase*, only *BackSpace* editing key is supported.
- During the *Startup Bluescreen* phase, the network redirector does not functioning.
  For this reason, it is impossible to restore backup images located on mapped network drives because their contents are unavailable at the *Startup Bluescreen* phase.

The detailed information about the interaction between Windows-based and Bluescreen components of Hard Disk Manager is discussed in the *Technician Manual*.

4.8 Selective partition restore

The function of *Selective partition restore* provides the ability to restore one or several partitions from *multi-partition images* in one operation (see Glossary).

4.8.1 Overview

The *Selective partition restore* is only the extended variant of the function Restore Partition from an Image function. The only difference is that it provides a more convenient mechanism of multiple partitions restoration from multi-partition images: a user can restore several partitions in one operation, and each restored partition can be resized during the restoration. In fact, the *Selective Restore* function consists of multiple restoration suboperations.

Take into account that Hard Disk Manager does not combine new data with restored ones. It irrevocably destroys the existing partition that is targeted for the selective restoration of saved partitions.

The implementation of this function varies in some cases:

1. In Windows, the operation differs for *unlocked* and *locked* partitions. To process locked partitions, the program need to reboot the computer.
2. The processing of *locked partitions* differs for Windows 95/98, Windows NT/2000/XP and Windows ME.

4.8.1.1 Where backup images can be taken from

Hard Disk Manager can restore partitions from backup images stored on the following media (see also the chapter Backup separate partitions, section Where you can save backup images):

- Mounted local drives and mapped network drives.
- CD/DVD-ROM drives. The equipment should be available through the system drivers.
- Mounted removable media including USB drives and Compact Flash cards (in Windows only).
- Any readable media that is mapped in the system.
- Unmounted partitions formatted to NTFS, Ext2 and Ext3.
  Hard Disk Manager uses the built-in *Universal Filesystem Driver* (UFSD) to access unmounted partitions.

4.8.1.2 Restrictions

Hard Disk Manager allows restoring the image over the existing partition. In this case, the following constraint should be taken into account: Hard Disk Manager is unable to use the image that is located (or holds some of its volumes) on the partition being restored.

To inspect the image type and the image contents, use the function Show Archive Info:

(menu) General ➔Show archive info…

4.8.2 Initiating the Selective Restore operation in the Windows-based version

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a block of free space to restore partitions to.

For the *Selective Restore* function, either a *block of free space* or an existing partition can be targeted as the future location of restored partitions. The targeted space can be selected either in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. See the section How Hard Disk Manager evaluates space available for restoration of the partition to understand the rules Hard Disk Manager follows to evaluate the available disk space for the restoration of partitions.

Step 2. Select the operation to perform

Variants:

- Select from the main menu:

Partition ➔ Selective partition restore

- Select from the main menu:

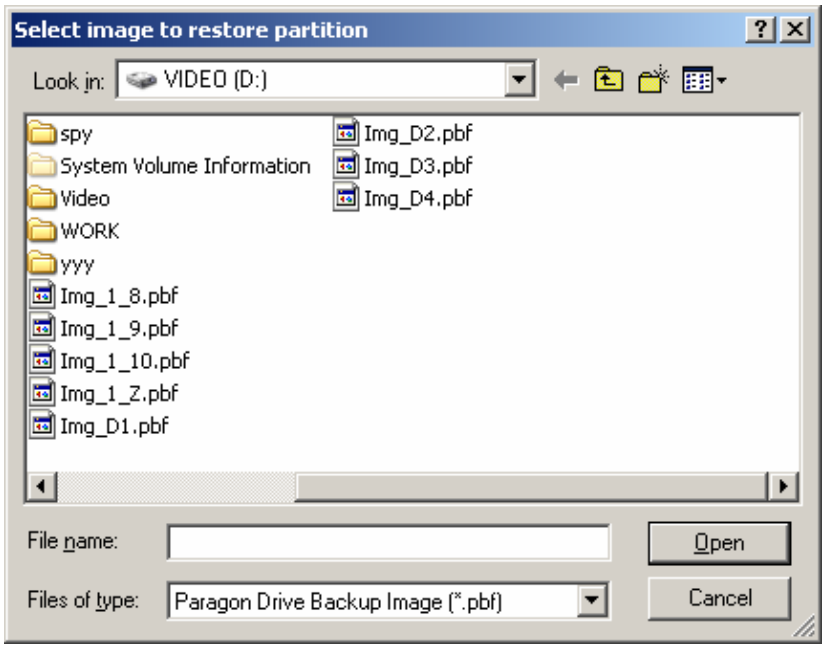
  Hard disk ➔ Selective partition restore on Hdd

- Call the *popup menu* for the targeted partition and select the menu item:

  Selective partition restore

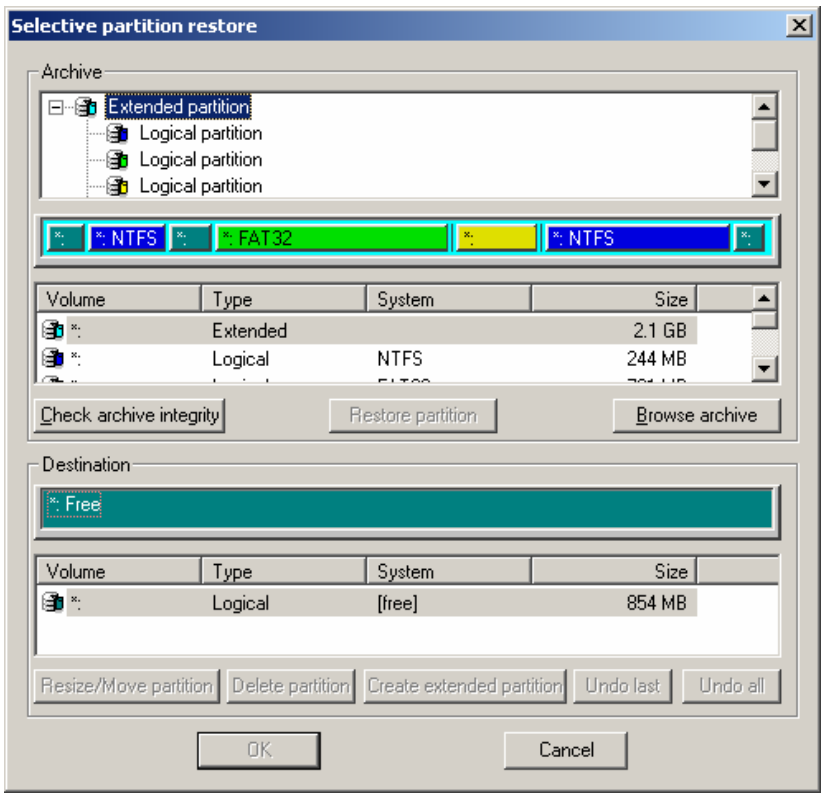
Step 3. Select an image that contains required partitions

After calling the *Selective Restore* operation, the *Open File* dialog appears. Select some backup image to restore a partition.



The *Selective Restore* function is applicable to images of hard disks, Extended Partitions and single partitions. You are able to select backup images from mounted local drives, mapped network drives and unmounted partitions on local hard drives (see the section Selecting unmounted partitions as the target location for saving backup images for more details).

After the image has been selected, the program opens the *Selective Partition Restore* dialog. It displays the layout of partitions included in the selected image, indicates most common parameters of partitions (type, size and filesystem) and provides the functionality of image checking and exploring, restoring partitions, changing size and location of restored partitions. The functionality of the dialog is described in the section Description of the Selective Partition Restore dialog functionality.



Step 4. Select a partition being restored

Select the desired partition in the Archive section (either in the Tree Layout, or in the UDP Layout, or in the List of Partitions). The Archive Tree Layout and Archive List of Partitions display only partitions and skip blocks of free space. The UDP Layout demonstrates the real structure of stored partitions.

Step 5. Select a desired location for restored partition

First, select a *block of free space* in the Destination section (either in the UDP Layout or in the List of Partitions).

If the selected partition can be restored to the selected block of free space, the button **Restore partition** becomes *enabled*. Otherwise, this button remains *disabled* (the button label remains grayed and recessed).

Press the **Restore partition** button to schedule the restoration of the selected partition in the List of Pending Operations.

Step 6. Change the size and location of the restored partition

The program allows resizing and moving restorable partitions in two ways:

- By using the *drag-&-drop* technique in the Destination UDP Layout.
- By using the **Resize/Move Partition** dialog.
  One should select a partition in the Destination List and then press the button **Resize/Move Partition** in the bottom of the window.

This functionality is described in the section Description of the Selective Partition Restore dialog functionality.

Step 7. Repeat two previous steps for other restored partitions

The program allows restoring multiple partitions within the single operation. The steps 5 & 6 should be repeated for each partition to be restored.

Note: At each iteration, you must definitely select both the partition to be restored (in the Archive panel) and the targeted block of free space (in the Destination panel).

The limitations are explained in the section Comments.

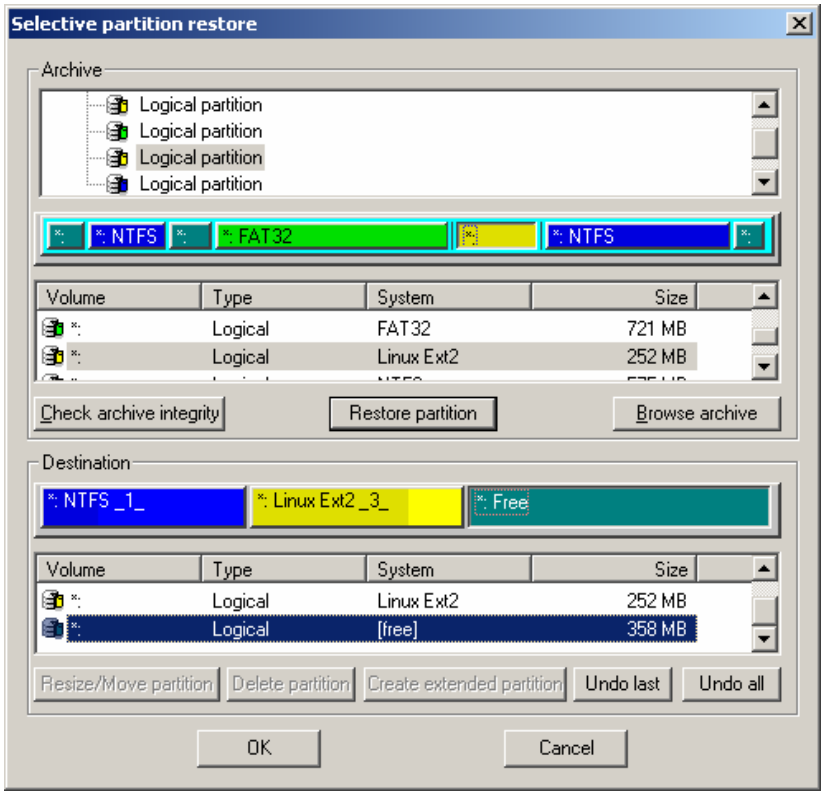
Step 8. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Selective Partition Restore* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.8.3 Description of the Selective Partition Restore dialog functionality

The *Selective Partition Restore* dialog displays the layout of partitions included in the selected image and indicates most common parameters of partitions such as type, size and filesystem type:



The window is divided in two sections:

The Archive section reveals the layout of stored partitions with three panels: the Archive Tree Layout, the Archive UDP Layout and the Archive List of Partitions. You can check the archive integrity, browse contents of imaged partitions and select partitions to be restored.

The Destination section displays the future layout of restored partitions with its own Destination UDP Layout and Destination List of Partitions. The section provides re-partitioning functionality that may be appropriate: the restored partitions may be deleted, resized and moved within the available range of the disk space.

In fact, the *Selective Partition Restore* function consists of multiple suboperations, so it supports the own list of suboperations. The Destination section provides the ability to edit this list.

4.8.3.1 The functionality of the *Selective Partition Restore* dialog

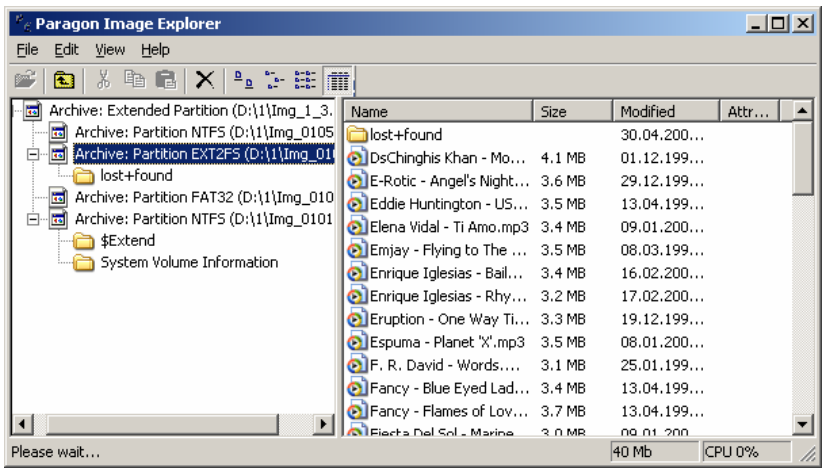
All the dialog functionality is available through operation buttons (the redistribution of restored partitions functionality is also available through *drag-&-drop* technique, see UDP control activity):

Archive panel: Check archive integrity

Press this button to validate the backup image. The function is described in the section Check archive integrity of the chapter Additional functions.

Archive panel: Browse archive

Press this button to run the built-in tool named *Paragon Image Explorer*. The utility allows browsing the backup image, including the browsing contents of separate partitions having types FATxx, NTFS or Ext2/3 with the ability to extract separate files and directories from the backup image.



The features and functionality of the *Paragon Image Explorer* utility is described in the chapter Supplementary functions.

Archive panel: Restore partition

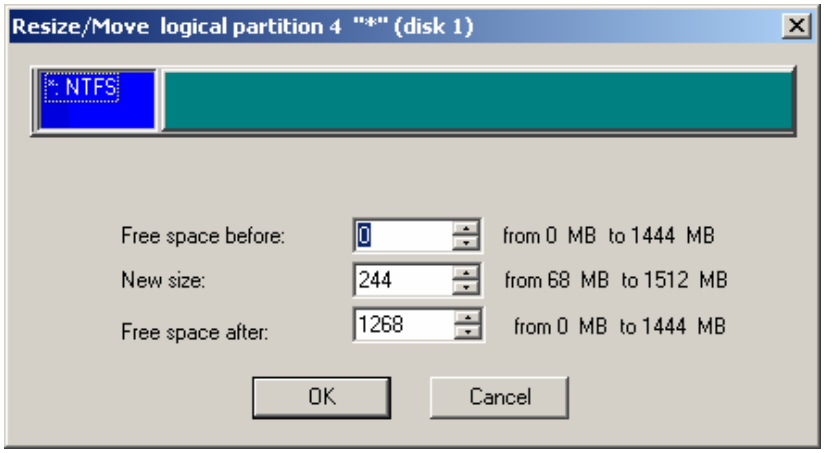
Press this button to schedule the restoration of the partition that is selected in the **Archive** panel to the block of free space that is selected in the **Destination** panel.

Note: this button is enabled only in case of the user really selects in both **Archive** panel and **Destination** panel.

Destination panel: Resize/Move partition

Press this button to open the *Resize/Move dialog* for the selected partition.

Note: the button is enabled only for restored partitions in the Destination panel. The partition must have the known filesystem type (see Known filesystems for more details).



The functionality of the *Resize/Move* dialog is similar to the functionality of the *Restore partition* dialog (see the section Defining the location and size of the restored partition). This dialog provides the ability to fully control the resulting position and the size of the selected restored partition.

The final capacity of the restored partition cannot exceed the size of the selected range of disk space and cannot be less than the amount of used space.

The program determines the range of disk space available for placing the restored partition (see How Hard Disk Manager evaluates space available for restoration of the partition).

Destination panel: Delete partition

Press this button to delete the restored partition.

Note: The button becomes enabled only in case of selecting some restored partition in the Destination panel.

As the matter of fact, this function simply removes the deleted partition from the list of restorable partitions – the program won't make idle actions of restoration and further deletion of a partition.

Destination panel: Create Extended Partition

Press this button to create the Extended Partition within the range of available disk space.

Note: the function is available only under the following conditions:

- A user has selected a block of free space in the Destination panel.
- There is no Extended Partition on the targeted hard disk.
- The total amount of Primary partitions is less than four (4).

If the button **Create Extended Partition** was pressed, the program creates the Extended Partition in the selected block of free space.

The last two conditions are the common limitations on the ability to create the Extended Partition (see *Technician Manual*).

The total amount of primary partitions is fundamentally limited to 4, and the creation of the Extended Partition allows placing more partitions on the hard disk.

Destination panel: Undo last

Press this button to cancel the latest operation that was scheduled in the *Selective Partition Restore* dialog. The function is similar to the **Undo** operation for the global List of Pending Operations (see the section Virtual operations).

Destination panel: Undo all

Press the button to cancel the all operations that was scheduled in the *Selective Partition Restore* dialog. The function is similar to the **Undo All** operation for the global List of Pending Operations (see the section Virtual operations).

4.8.4 Running the Selective Restore operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed and summarized data transfer rate

The restoration of each archive volume is treated as suboperation, and the program displays information about suboperation progress.

In case of a multivolumic image is used, the program may pause and display the *Open File* dialog to ask the user for the next volume filename and location.

Note: this is not the usual behavior of Hard Disk Manager. The program pauses the restoration only in the following cases:

- ⇨ some volumes of the image were moved or renamed
- ⇨ volumes are located on removable media

4.8.5 Comments

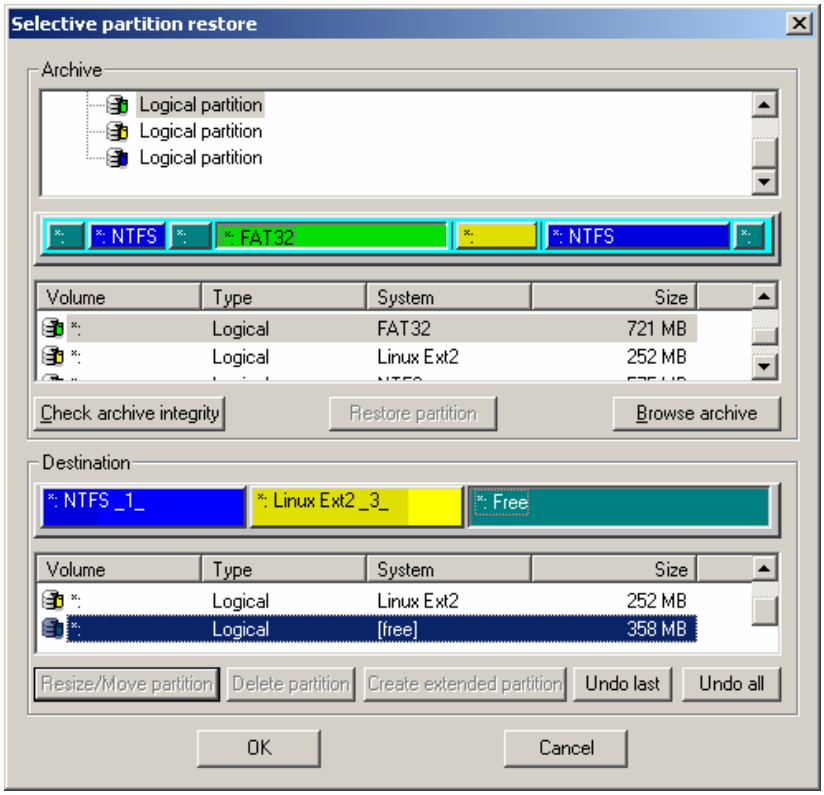
4.8.5.1 Conditions on the size of the block of free space

As it was mentioned in the section How Hard Disk Manager evaluates space available for restoration of the partition, Hard Disk Manager requires the partition being restored must fit the targeted range of disk space.

Concerning the *Selective Partition Restore* operation, in every iteration, each restored partition must have the initial size less than the size of the targeted block of free space.

The picture placed below demonstrates such a kind of mistaken situations:

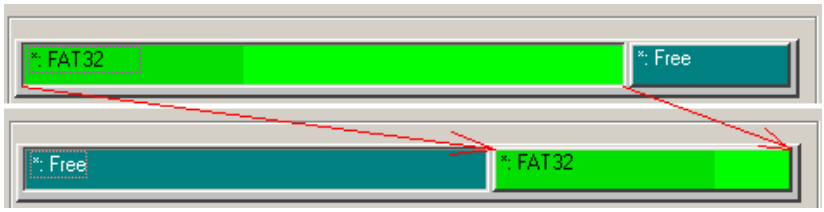
The selected FAT32 partition is almost empty, and it is less than the overall size of the range of available space. Still, its size exceeds the size of currently selected block of free space.



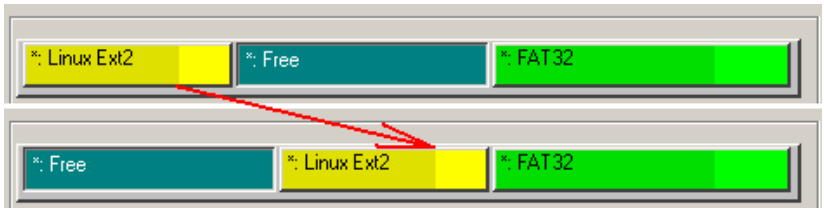
In this situation, Hard Disk Manager cannot restore the selected FAT32 partition.

To bypass the limitation, one can change the order of restoration of partitions:

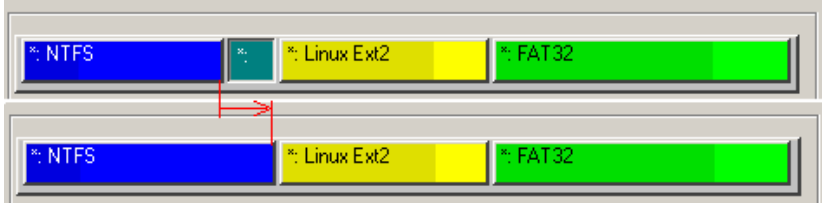
1. The FAT32 partition should be restored first. It also must be squeezed and moved to the end of the range of disk space.



2. Next, restore and move the Ext2 partition (its initial size meet the restoration conditions).



3. Restore and resize the NTFS partition (its initial size meet the restoration conditions).



4.9 Restore Hard Disk from Backup Image

This chapter explains how to restore hard disks from previously made backup images under various conditions.

4.9.1 Overview

In terms of Hard Disk Manager, a "restoration of hard disks" means retrieving all informational components of a hard disk: all partitions, the Partition Table and bootstrap code. During the operation, the program destroys all previous disk contents and replaces them with saved ones.

Any changes in the disk layout will be irreversibly lost after the disk restoration.
New data, which are added to any partition after making the backup image of the hard disk, will be irreversibly lost after the disk restoration.

Note, if your aim is only add partitions from a backup image to the currently used hard disk, with keeping some of its contents, you should use Selective Partition Restore function instead of *Restore Hard Disk*.

Hard Disk Manager supports several kinds of backup images:

- Images of single partitions.
- Images of multiple partitions,

  e.g. images of the Extended Partition.
  Such images include multiple subordinate images.

- Images of the 1st track of a hard disk,

  which include only 1st track saved (most OS-independent boot managers occupy 1st track; with this option, it's possible duplicating boot managing software).

- Images of MBR sector,

  which include only MBR sector saved.

- Images of entire hard disk.

  These images include MBR, 1st track, Partition Table and all on-disk partitions. These images consist of multiple subordinate images.

To inspect the image type and the image contents, use the function Show Archive Info:

(menu) General ➔Show archive info…

In addition, the program displays type of contents for backup images, which are included in the *List of Backup Images*.

Hard Disk Manager can use an image of any type for restoration over a hard disk:

- In case of using hard disk image, the program destroys all previous disk contents and then restores ones saved in the image.
- In case of using images of partitions, the program destroys previous partitions and Partition Table, with replacing them by restored ones.
  A bootable code placed in MBR, or boot manager's code in the 1st track, are kept intact.
- In case of using MBR image or 1st track image, the program restore just MBR or the 1st track contents, with keeping intact current partitions and the Partition Table; so that disk layout remains unchanged.

The implementation of the Restore Hard Disk function varies in some cases:

1. In Windows, the operation differs for *unlocked* and *locked* (system) hard disks. To process locked hard disks, the program need to reboot the computer.
2. The processing of *locked hard disks* differs for Windows 95/98, Windows NT/2000/XP and Windows ME.

4.9.1.1 Where backup images can be taken from

Hard Disk Manager can restore partitions from backup images stored on the following media (see also the section Where you can save backup images):

- Mounted local drives and mapped network drives.
- CD/DVD-ROM drives. The equipment should be available through the system drivers.
- Mounted removable media including USB drives and Compact Flash cards (in Windows only).
- Any readable media that is mapped in the system.
- Unmounted partitions formatted to NTFS, Ext2 and Ext3.
  Hard Disk Manager uses the built-in *Universal Filesystem Driver* (UFSD) to access unmounted partitions.

4.9.1.2 Restrictions

In case of restoration the hard disk, the following constraint should be taken into account:

- Hard Disk Manager is unable to use the image that is located (or holds some of its volumes) on partitions, which belong the targeted hard disk.

4.9.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Previous versions of Hard Disk Manager traditionally expect that a user chooses a hard disk and then selects an operation for a chosen object.

Current version of Hard Disk Manager allows also initiating the *Restore* function as an operation over a backup image. Sections below describe both ways.

4.9.2.1 A traditional procedure of restoration over a hard disk

With this way, the program allows restoring following objects:

- It can restore all disk contents from images of a hard disk;
- It can restore 1st track from images of 1st track;
- It can restore MBR bootstrap code from images of MBR

In the traditional restoration procedure, the program does not allow using images of partitions, in order to exhibit the behavior compatible with old versions of Hard Disk Manager.

Step 1. Select the hard disk to be restored

There are two variants:

- select a hard disk in the Tree Layout panel or in the UDP Layout panel.
- select any partition that belongs a hard disk of interest.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

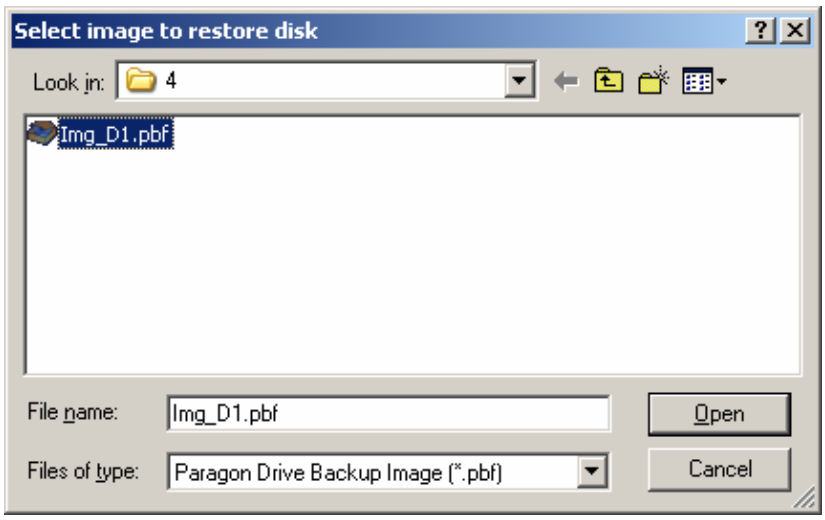
  Hard disk ➔ Restore hard disk from image…

- Press the **Ctrl+R** key combination.
- In case of a hard disk is selected, Press **Restore** button on the Main Toolbar.
- In case of a hard disk is selected, call the *popup menu* for the selected partition in any of layout panels (click right mouse button)
  then select the menu item

  Restore hard disk from image…

Step 3. Select an image that should be used for hard disk restoration

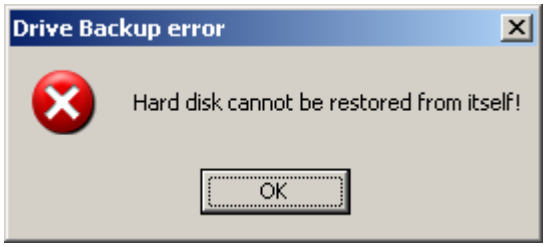
The program displays the *Open File* dialog in order that a user selects an image to be used for restoration:



It is allowed selecting files from any *mounted local drive*, on *mapped network drive* and on *unmounted partitions* from local hard disks (see the section Selecting unmounted partitions as the target location for saving backup images for more details).

At this stage, the program can generate following error messages:

- In case of the selected image is located on the hard disk that was selected for restoration, the program displays the error message:

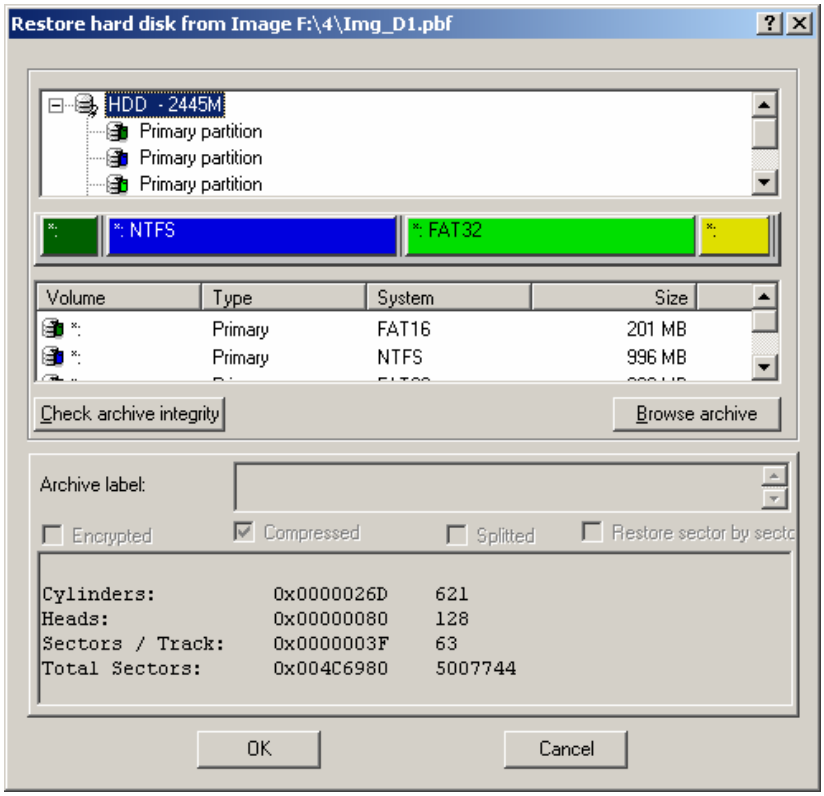


- In case of the selected image contains too much usable data, which cannot went into the selected hard disk, the program displays the error message:

In any of these cases, one should choose another image for disk restoration.

Step 4. Inspect contents of the hard disk image

The program displays the *Restore Disk from Image* dialog:



A user can see properties of saved hard disk configuration:

- ⇨ amount of partitions
- ⇨ type, filesystem and size for each partition
- ⇨ image properties (compression, encryption, label etc)
- ⇨ information about hard disk geometry.

Step 5. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Restore Hard Disk* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the Restore Disk operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.9.2.2 An alternative procedure of restoration of an image

An alternative way is to perform the *Restore* operation over a selected backup image. The program keeps the database of recently used backup images (see the chapter List of Backup Images).

With this way, the program allows restoring following objects:

- It can restore all or selective partitions from images of a hard disk;
- It can restore 1st track from images of 1st track;
- It can restore MBR bootstrap code from images of MBR;
- It can replace current disk layout by a single partition restored from images of partitions.
- It can replace current disk layout by selective partitions restored from multi-partition images

Step 1. Select a backup image to be restored

Select an image in the List of Backup Images. The program displays a type of an image in the first column.

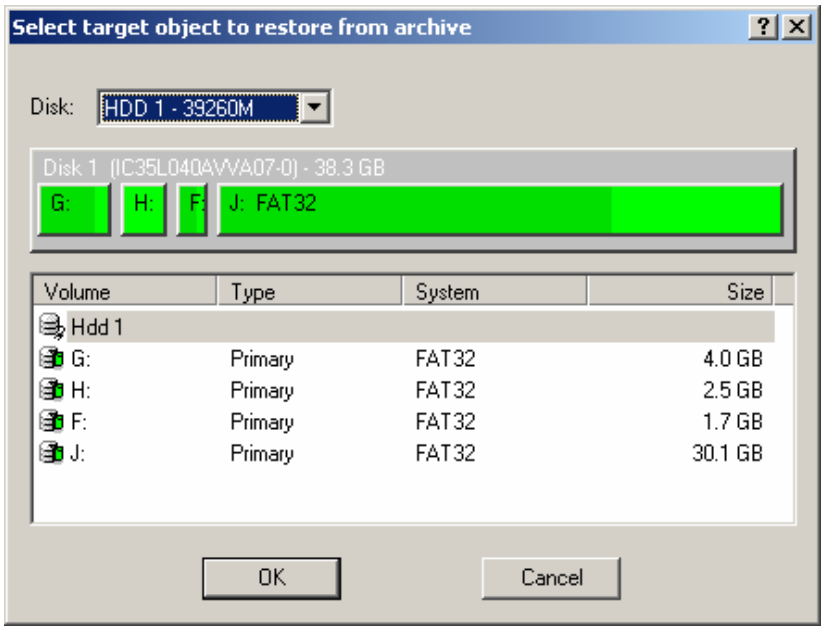
Step 2. Select the operation to perform

Call the *popup menu* for the selected image, and select the menu item:

Restore archive…

Step 3. Select an object that should be restored from the image

The program displays the window "**Select target object to restore from archive**":



This dialog allows selecting destination object for the restoration. The detailed description is given in the next section Description of the dialog "Select target object to restore".

Step 4. Inspect image contents and select ones to be restored

The program displays the **Selective Partition Restore** dialog in order a user is able customizing disk layout that should be restored.

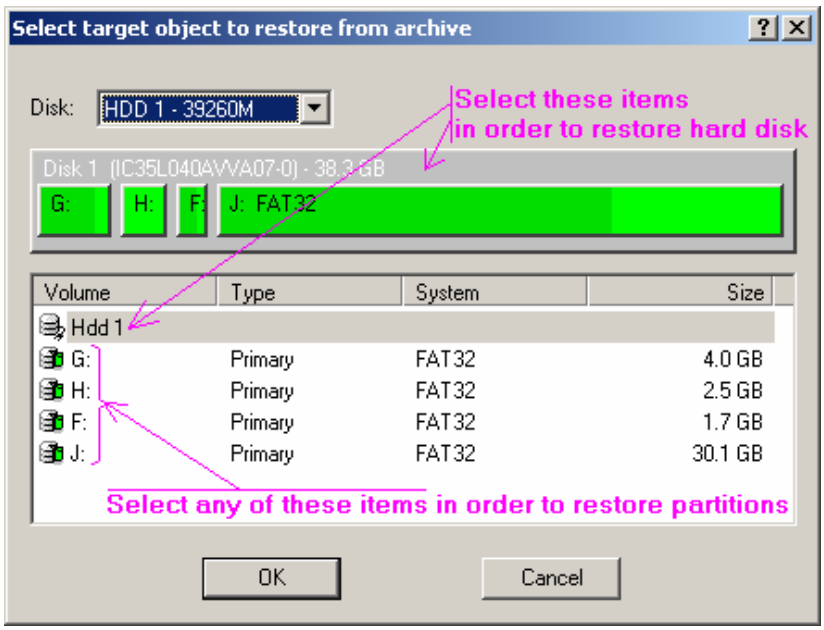
The functionality of this dialog is same with the Selective partition restore function.

Step 5. Apply the operation

The final step is common for all operations (see the section Applying operations):

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Restore Hard Disk* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the Restore Disk operation in the List of Pending Operations.

4.9.2.3 Description of the dialog "Select target object to restore"



Disk

This pull-down list allows selecting a hard disk, which contains a targeted object.

The *UDP control* and the *List of Partitions*, which are placed below, display contents of the selected hard disk that can be restored from the selected backup image. Either of these controls can be used for selecting a restorable object.

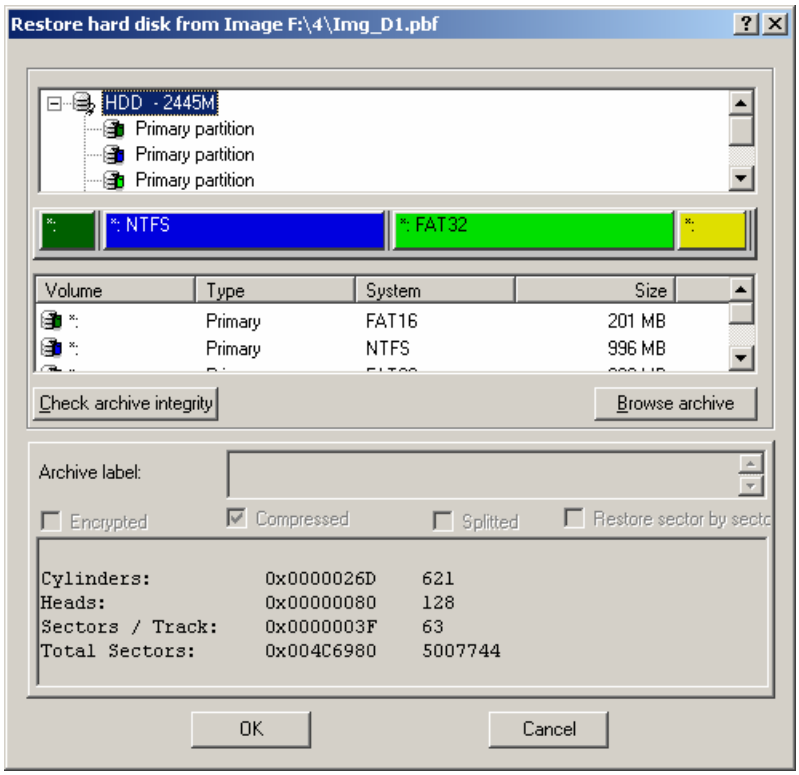
The table below exhibits rules, which Hard Disk Manager follows:

Image type	Which disk components:	
	Are displayed	Can be restored
HDD	• Hard disk • Partitions	• Entire hard disk • Selective partitions
Partition	• Hard disk • Partitions	• A disk with single partition • A selected partition
First Track	Hard disk	1st track of the selected hard disk
MBR	Hard disk	MBR of the selected hard disk

4.9.3 Description of Restore Hard Disk dialog

In fact, the *Restore Hard Disk* dialog is very similar to the Show Archive Info one. It provides the following functionality:

- Confirm or cancel the operation
- Explore properties of saved objects
- Check integrity of the selected image
- Explore contents of saved partitions



Hereafter the brief description of available functionality:

Check archive integrity

Press this button for immediate performing the Check Archive Integrity operation.

This option allows making certain about usability of the selected backup image prior the real execution of the operation.

Browse archive

Press this button to browse the archive contents with using the Paragon Image Explorer built-in tool.

Image Explorer allows exploring files and directories from partitions, which are saved in the image, opening files in the read-only mode and extracting separate directories and files from the backup image.

Archive label and archive options

The grayed controls exhibit information about the selected archive: the descriptive text (Archive Label), whether or not Compression, Encryption, Splitting were implemented to the archive.

The upper part of the window

contains the group of panels, which represent the layout of disk contents saved in the selected image: List of Partitions, UDP controls, Tree-Layout. Each of these panels can be used for selecting disk contents in order to inspect its properties.

The lower part of the window

displays properties of the selected object:

- In case of a "hard disk" object is selected, the program displays properties of a saved hard disk (commonly named *hard disk geometry* see Glossary for more details):

Cylinders	Amount of *Cylinders* on the saved hard disk
Heads	Amount of *Heads* on the saved hard disk)
Sectors/Track	Amount of *Sectors per Track* on the saved hard disk
Total Sectors	The total amount of sectors on the saved hard disk

- In case of a "partition" object is selected, the program displays properties of a selected partition:

File system	File system that is placed on the selected partition
Drive letter	--- not available for partitions from backup images ---
Volume label	Volume label of the selected partition; can be used for distinguishing partitions in case of *Selective Partition Restore*.
Sectors/boot	Amount of sectors reserved for the bootable code on the selected partition. In fact, this information of least importance for most users.
Sectors/Cluster	*Cluster Size* value expressed in Sectors. Halve this value to get the *Cluster Size* value in Kbytes.
Size (capacity)	Capacity of the selected partition in bytes and Mbytes
Used space	Amount of usable data on the selected partition, percentage and the real value in Mbytes
Free space	Amount of free space on the selected partition, percentage and the real value in Mbytes

4.9.4 Running the Restore Hard Disk operation

Let remind once again that this operation irrevocably destroys *all* contents of *all* partitions that are located on the targeted hard disk!

The most first action at the real disk restoration is the overwriting of existing Partition Table, which causes to the immediate removal of all partitions. Even the "fastest" interruption of the operation is too slow for the aborting the first stage of the operation.

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed and summarized data transfer rate
- Currently used volume filename

The restoration of each archive volume is treated as suboperation, and the program displays information about suboperation progress.

4.9.4.1 Restoring multivolumic archives

Hard Disk Manager always creates hard disk images consisting in multiple volumes. Volumes can be placed in different directories, on different disks and removable media. During the restoration from a multivolumic image, the program

searches volumes on their original location for completing the process without user's intervention (see aso the section Backup Partition➔Running the Backup operation)**.**

In case of some files were renamed or moved to another location, the program pauses working and displays the *Open File* dialog in order to a user points next volume location manually.

Note: this is not a usual behavior of Hard Disk Manager. The program pauses the restoration only in the following cases:

- ⇨ some volumes of the image were moved or renamed
- ⇨ volumes are located on removable media

The program will automatically restore multivolumic image in "unattended" fashion, in the following cases:

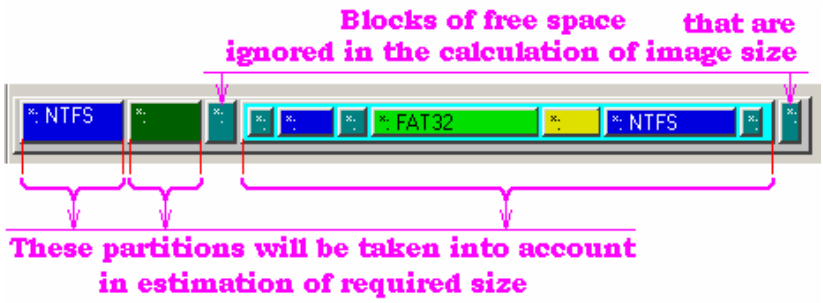
- All volumes of the image remain on their original locations, and all of them are placed on local hard disk(s).
- All volumes of the image are picked up in the same directory on a local hard disk.

4.9.5 Comments

4.9.5.1 How Hard Disk Manager evaluates the disk space required to restore the image of the hard disk

Hard Disk Manager uses the following rules of image appropriateness in the *Restore Hard Disk* function:

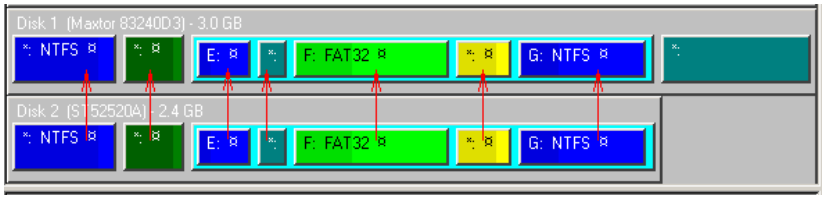
1. The program calculates the overall size of all primary partitions and the Extended Partition (if exists) that are saved in the image of the hard disk. Let's name it *image size* in this section.



2. The primary constraint is that the summary *image size* must not exceed the capacity of the targeted hard disk. Otherwise the program immediately cancels the operation:

3. If the targeted hard disk is larger than the one saved in the image, the program restores the hard disk from the image with keeping intact the position and size of all on-disk partitions. The residual disk space remains unpartitioned.

See the illustration below: the larger disk (Disk #1, 3.0Gb) is restored from the image of the smaller disk (Disk#2, 2.4Gb). The last 600Mb of disk space remains unpartitioned (i.e. this space does not belong to any partition):



4. If the targeted hard disk is less than the one saved in the image, but greater than the summary *image size*, the program restores the hard disk from the image with keeping intact only the relative order and size of all on-disk partitions. The blocks of free space will be proportionally squeezed.

See the illustration below: the smaller disk (Disk#2, 2.4Gb) is restored from the image of the larger disk (Disk #1, 3.0Gb). Primary partitions and the Extended Partition keep their sizes. The structure of the Extended Partition remains.

Blocks of free space between primary partitions and the Extended Partition are squeezed with keeping the size proportion:

4.9.5.2 How to avoid bad sectors during the restoration

In case of restoring data to the bad sectors, the resulting partition(s) may become corrupted. To avoid loosing data due to bad sectors, select the *surface test* to the value other than **None**.

(menu) General ➔ Settings… ➔ (page) General ➔ Surface test

In this case, the program performs the surface test prior to the data restoration over the disk space that will be occupied by the restored partition(s). In case of detecting bad or unreliable sectors, the program will mark them *bad* and will not place data to those sectors.

The additional surface test noticeably increases the overall time required to complete the operation, also the algorithm that bypasses bad sectors significantly slows down the performance. Use this feature only in case of you suspect that there are bad sectors on the chosen hard disk.

4.9.6 Restoration of locked hard disks

Hard Disk Manager allows restoring the image over the partitioned hard disk. Generally, this operation is similar to the restoration of locked partitions (see the section Restoration of locked partitions). The program deletes all existing partitions prior to the real restoration of data. Still, there could be a situation when some of partitions on the targeted hard disk are used by another software. In this case, Hard Disk Manager is unable to complete the data restoration.

To avoid the damage of data consistency, Hard Disk Manager reboots the system to a single-tasking environment in order to eliminate the interference of other programs.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

4.9.6.1 Restoration over locked hard disks in Windows NT, 2000 and XP

In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase to operate the locked partitions:

1. Before starting the operation, the program checks whether partition is locked or not.
2. If the partition appears is the locked one, the program asks for reboot the system. Press **OK** button to reboot the system and complete the operation, press **Cancel** button to abort the operation.

By default, the program pauses the execution until the user makes the choice. For configuring the unattended working, see the section How to automate the system reboot for processing the locked partitions.

3. The program silently schedules the *BlueScreen Component* to run at next Windows startup and to execute the required operation.
4. Then the program reboots the computer.
5. At the next system boot-up, the Bluescreen Component executes the operation in place of the Windows-based version. The BlueScreen Component will display the operation progress in the console-like style.
6. In case of using the multivolumic image having some volumes:
   - ⇨ moved to another location
   - ⇨ placed on removable media (such as CD/DVD-ROM)

   the program may fail to find automatically the sequel of the multivolumic image. In this case, the program asks the user for the filename of the next volume. The most bottom lines of the console output will be the following:

```
Get new filename of the file for subsequent reading.
(Type exit to cancel operation.)
D:\Img_1_8.p01
```

7. You should choose one of the following alternatives:
   - ⇨ change the filename to proceed with another filename.

⇨ clear the filename and type the word "*exit*" to abort the operation.

8. By default, Hard Disk Manager suggests the filename that is registered in the archive. The problem is that the program has failed to open the volume (otherwise it would continue working without pausing). At this moment, the user must *guess right* the filename to continue with the restoration.
9. After the BlueScreen Component completes the operation, the Windows session begins and the Windows-based version of Hard Disk Manager comes up.

The text editing abilities in the BlueScreen Component:

See the section The text editing abilities in the BlueScreen Component.

4.9.6.2 Limitations of the BlueScreen Component

There are some minor functionality limitations of the BlueScreen Component that come from the inaccessibility of some Windows services during the Startup Bluescreen phase:

- The version of the BlueScreen Component provided with Hard Disk Manager interacts with the user in the console-like style. Such useful functions like *Browse disk contents* and *Search files* are not available. In case of the program asks a user to enter a filename, the user must enter a filename "blindly".
- The BlueScreen Component allows input only English letters.
  The text is in *lowercase*, only *BackSpace* editing key is supported.
- During the *Startup Bluescreen* phase, the network redirector does not functioning.
  For this reason, it is impossible to restore backup images located on mapped network drives because their contents are unavailable at the *Startup Bluescreen* phase.

The detailed information about the interaction between Windows-based and Bluescreen components of Hard Disk Manager is discussed in the *Technician Manual*.

4.10 Copy Partition

This chapter explains how to copy single partitions that are placed on local hard disks, under various conditions.

4.10.1 Overview

Copy is another primary function of Hard Disk Manager. The copying of partitions can be used for the purposes of:

- cloning "sample" partitions
- making backup copies of working partitions

One can duplicate partitions to protect oneself from downtime in case of system malfunction. The partition can be copied back to the original place within few minutes, or it can be used for copying just separate files.

Hard Disk Manager allows to copy partitions of any type. Partitions of unknown filesystem types are always copied in the *sector-to-sector* copying mode, and both copy and original have the same size.

As to partitions of known filesystem types, the program provides an ability to copy them either in the fast copying mode or in the *sector-to-sector* mode. In addition, there is the ability to automatically change the partition size during the copying.

Hard Disk Manager duplicates all usable partition data including files, the exact structure of directories and *filesystem metadata*: location of files, security information, access quotas and so on.

The implementation of the backup function varies in some cases:

1. In Windows, the operation differs for *unlocked* and *locked* (system) partitions. To process locked partitions, the program need to reboot the computer.
2. In turn, the processing of *locked partitions* differs for Windows 95/98, Windows NT/2000/XP and Windows ME.

4.10.1.1 Where original and duplicate partitions can be located

Currently, Hard Disk Manager allows to copy partitions that are located on non-removable local hard disks only. Hard disks should be connected to IDE, SCSI or RAID controllers and should be somehow available in the operating system (see Comments for more details).

The program allows to copy partitions within the single hard disk and between local hard disks of types mentioned above, in any combination. In particular, it is possible to copy partitions from IDE to SCSI disks and vice-versa.

The program allows to copy primary partitions into the Extended Partition (so that a primary partition becomes a logical one). In addition, logical partitions can be copied outside the Extended Partition (so that a logical partition becomes a primary one).

4.10.1.2 Restrictions

Limitations in supported media

- The current version of Hard Disk Manager does not support removable hard disks such as USB, PCMCIA or LPT hard disks or FLASH cards.
- The program cannot copy partitions on remote hard disks. LAN, USB, LPT/COM and other type connections are not supported.
- Finally, the program does not allow to copy contents of formatted removable media as separate partitions on hard disks.

In addition, the significant restriction of the current version of Hard Disk Manager is that it supports only DOS partitioning scheme (see Glossary).

Functionality limitations

Hard Disk Manager allows to copy partitions only to the blocks of free space.

To copy some partition over another one, use the following method:

1. delete the targeted partition
2. select the partition that should be copied
3. copy the selected partition to the block of free space.

4.10.2 Initiating the operation

The *Copy Partition* operation can be initiated in several ways that differ in their user friendliness and the flexibility. According to your aims, you can choose either the fast method based on the *drag-&-drop* technique or the accurate method based on entering parameters in the *Copy Partition* dialog.

4.10.2.1 Using the drag-&-drop technique

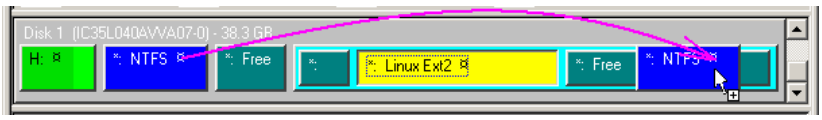
The *drag-&-drop* technique is available only in case of enabling virtual execution of operations, in the *Smart* and *Virtual Execution* modes (see the chapter Virtual operations).

The *drag-&-drop* technique for the *Copy Partition* operation is supported in all layout panels (the Tree Layout panel, the UDP Layout panel and the List of Partitions):

- select the partition you want to copy
- press and hold the primary (=left) mouse button
- then press and hold the **Ctrl** keyboard key
- drag a partition to a block of free space
- and drop it to the desired position.

During the dragging a partition, the program indicates the availability of the operation for the current position:

The program is able to copy the partition to this place:



The program cannot copy the partition to this place:

See the Comments for more details about copying rules and constraints.

4.10.2.2 Drag-&-drop copying limitations

1. The *drag-&-drop copying* is enabled only in the *Virtual Execution mode*. See the section Virtual operations for more details. Remember, you can switch on/off the *Virtual Execution mode* from the program's Settings:

   (menu) General ➔ Settings… ➔ (page) Virtual operations ➔ Enable virtual operations

2. The *drag-&-drop copying* allows only copy a partition without changing its size (within the single operation).

4.10.2.3 Using the Copy Partition dialog

The *Copy Partition* dialog provides the full functionality of the Copy function: one can copy a partition to any available disk, set accurately any admittable position for the duplicate partition and define precisely its final size.

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select the partition you want to copy.

Select the partition you want to copy, in the Tree Layout panel, or in the UDP Layout panel, or in the List of Partitions. In any case, the partition will be highlighted in all three panels. The function is not available for blocks of free space.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

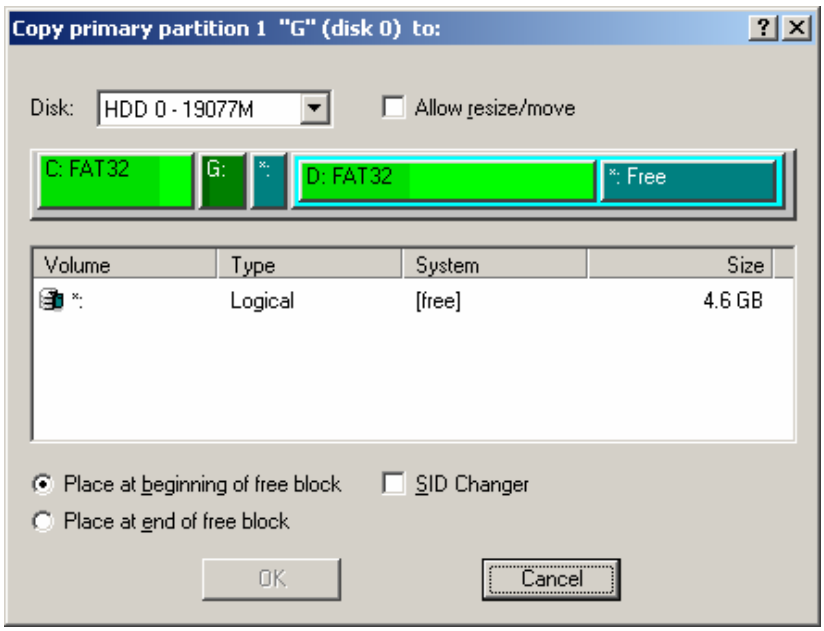
  Partition ➔ Copy partition…

- Call the *popup menu* for the selected partition in any of layout panels (click right mouse button) then select the menu item:

  Copy partition…

- Press **Alt+C** keyboard combination
- Press **Copy** button on the Main Toolbar.

Step 3. Define parameters of the operation



In the *Copy Partition* dialog, you can define the properties of the duplicate partition: targeted disk, position and size of the duplicate partition. The section Description of the parameters contains the detailed description of the dialog's functionality.

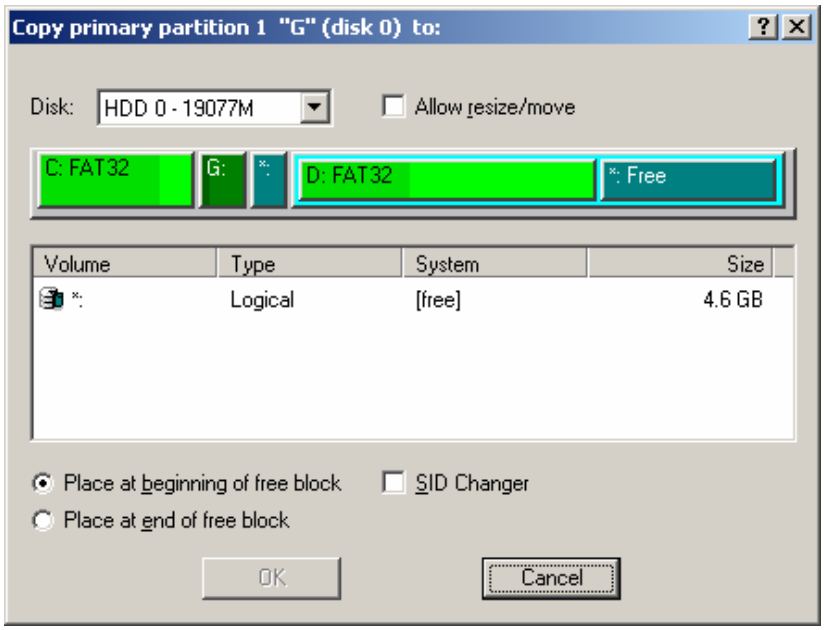
Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Copy Partition* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the Copy operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.10.3 Description of the parameters

4.10.3.1 Copy partition dialog



Disk

Select the hard disk, where the duplicate partition should be placed.

After selecting the targeted hard disk, the program displays the current disk layout in the UDP control that is located in the *Copy Partition* dialog. The *List of Available Places* placed below contains only the list of free blocks (on the selected disk) that match the constraints on the disk space for copying partitions.

Initially, the *List of Available Places* includes only blocks of free space that are larger or equal in size the original partition.

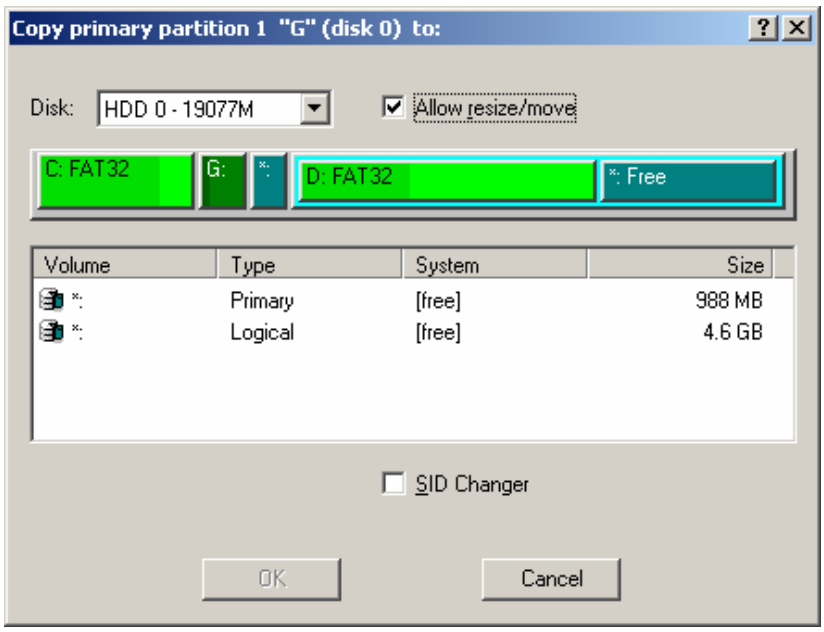
If the *List of Available Places* remains empty, there is no appropriate place to copy the selected partition. Choose another hard disk for duplicating the partition.

Allow resize/move

Set this checkmark to allow the program simultaneously changing the size of the duplicate partition during the copying. In terms of Hard Disk Manager project, this feature is named *autoresize*.

The state of the checkmark affects the *List of Available Places* and on the available functionality at executing the operation (see Comments for more details).

If the checkmark is marked, the *List of Available Places* includes blocks of free space that are larger than the amount of data on the selected partition (see the picture below). The partition can fit additional free blocks in case of performing the partition shrinking.



On the picture: the 2nd partition was selected to copy. Initially, it can be copied only to the logical block of free space. After *resize/move* enabling, the selected partition can also be copied to the primary block of free space (because this free block larger than the amount of on-partition data).

Place at beginning of free block
Place at end of free block

These radio buttons allow fast aligning the partition to the right or the left edge of the selected block of free space.

In case of the **Allow resize/move** checkmark is turned OFF, this is the only option for selecting the position of the duplicate partition.

In case of the **Allow resize/move** checkmark is marked, the program ignores the state of these radio buttons but uses the *Resize/Move Partition* dialog for defining the position and size of the duplicate partition.

SID Changer

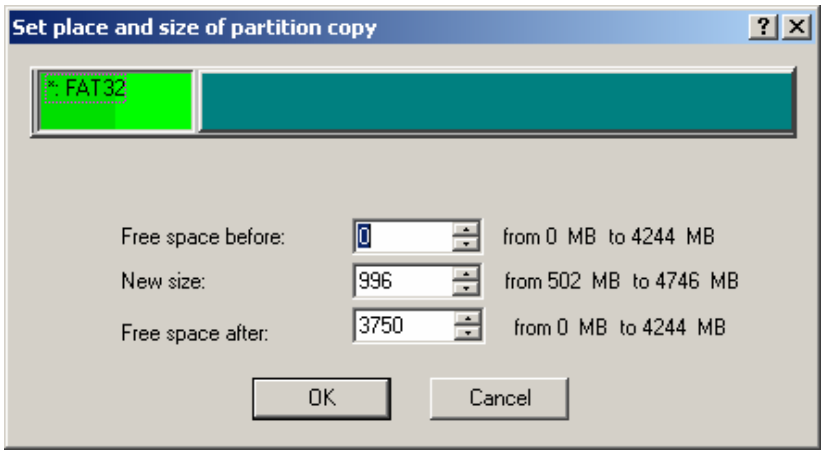
Set this checkmark so that the program runs the built-in SID Changer at the end of the copying this partition. The option can be potentially useful in case of copying the Windows system partition.

The *List of Available Places*

Finally, select some item in the *List of Available Places*. The program will copy the partition to the selected block of free space.

4.10.3.2 Resizing & moving the duplicate partition

In case of the **Allow resize/move** checkmark is marked, the program automatically runs the Resize/Move Partition dialog for accurate defining the position and size of the duplicate partition:



Use three spinner controls or the drag-&-drop technique over the UDP control to locate and change the size of the duplicate partition (see the section Resize&Move Partition ➔ Description of parameters or Restore Partition ➔ Defining the location and size of the restored partition for more details).

The UDP control and spin controls are synchronized, the changing of any of these elements affects on all other ones.

How the spin controls behave:

Free space before	Moves the beginning of the partition (left edge), preferably with keeping the partition size.
New Size	Changes the size of the partition, preferably with keeping the starting position (left edge).
Free space before	Moves the end of the partition (right edge). On increasing the value, it (preferably) keeps the partition size. On decreasing the value, it (preferably) keeps the starting position (left edge) so that the partition expands.

There are rules that take effect in the partition resizing. See the section How Hard Disk Manager evaluates space available for partition copying for more details.

4.10.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed and summarized speed of the operation

The operation is lengthy. Real performance fundamentally depends on hardware and an operating system being used.

4.10.5 Comments

4.10.5.1 How Hard Disk Manager evaluates space available for partition copying

In case of copying a partition without changing its size, the program uses the following rules for placing the duplicate partition:

For partitions that are copied in the *sector-to-sector* mode:

1. The program always keeps the partition size. The resizing is not available.
2. The starting position of the partition (the left edge) is aligned to the boundary of the corresponding cylinder, in accordance with the rules of the DOS partitioning scheme.
3. The ending edge remains "as is". In particular, it can become unaligned to the boundary of the corresponding cylinder (for example, after copying the primary partition to the Extended Partition or after copying the logical partition outside the Extended Partition).

For partitions of *known types* that are copied in the *fast copying* mode:

1. The beginning and ending edges of the duplicate partition are aligned in accordance with the rules of the DOS partitioning scheme:
   - ⇨ Ending positions of all partitions are aligned to the end of corresponding cylinders.
   - ⇨ Starting positions of logical partitions are aligned to the second track (track #1) of corresponding cylinders.
   - ⇨ The beginning of the most first primary partition, which begins from the cylinder#0, is aligned to the track #1.
   - ⇨ Starting positions of other primary partitions are aligned to beginning of corresponding cylinders.
2. In case of copying a partition without changing its size, the program keeps the amount of cylinders that partition holds. So that:
   - ⇨ Primary partitions are slightly reduced when they have been copied to the Extended Partition.
   - ⇨ Logical partitions are slightly expanded when they have been copied outside the Extended Partition.

In case of copying a partition with simultaneous changing its size, the program uses the following rules for placing the duplicate partition:

1. The partition boundaries are aligned in accordance with the rules of the DOS partitioning scheme.
2. The size of the partition can be selected between the following values:
   - ⇨ It cannot exceed the size of the selected block of free space
   - ⇨ It must be larger than the on-partition used space.

⇨ The used space is estimated as the summary size of clusters that are used by files, directories and filesystem metadata.

3. The program tries to keep intact important partition parameters such as the *Filesystem Type* and the *Cluster Size*, if it's possible.

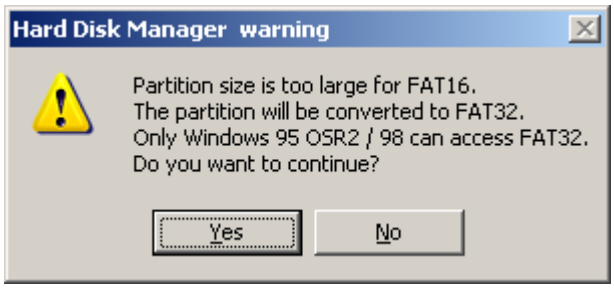
Thus, the program does not change these partition properties for NTFS and FAT32 partitions within the very large range of partition size values.

It is not so with FAT16 partitions: the doubling partition size usually requires the *Cluster Size* doubling. Hard Disk Manager automatically performs changing the cluster size when changing the partition size.

In addition, the size of FAT16 partitions is limited to the value of 2Gb. Windows NT 4.0 supports FAT16 partitions up to 4Gb in size (in this case the *Cluster Size* value is 64K). Unfortunately other operating systems do not support 64K cluster size. Hard Disk Manager supports the 64K cluster size (see Settings overview).

4.10.5.2 Converting FAT16 to FAT32 during the automatic resizing

FAT16 partitions are limited to the value of 2Gb (see comments above). Nevertheless, Hard Disk Manager allows expanding FAT16 partition over the 2Gb boundary. In fact, the program can automatically convert the FAT16 filesystem to FAT32. In case of such conversion is required, the program displays the warning message:



There is the ability to inhibit this warning (see the chapter Settings overview, section General Page):

(menu) General ➔ Settings… ➔ (page) General ➔ (checkmark) Convert FAT16 to FAT32 automatically

In this case, the program automatically converts FAT16 partitions to FAT32 without warnings.

4.10.6 Copying locked and system partitions

Hard Disk Manager allows to copy locked and system partitions. In case of other software or the operating system modifies contents of the partition being copied, the program stops the operation as it cannot make the valid duplicate. To complete the copying of a locked/system partition, Hard Disk Manager reboots the system to a single-tasking environment in order to eliminate the interference of other programs.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

Generally, the copying of locked partitions is mostly like the creating of backup image of the locked partition (see the section Backup system and locked partitions for more details).

4.10.6.1 Copying locked partitions in Windows NT, 2000 and XP

In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase to operate the locked partitions:

1. Before starting the operation, the program checks whether partition is locked or not.
2. If the partition appears is the locked one, the program asks for reboot the system. Press **OK** button to reboot the system and complete the operation, press **Cancel** button to abort the operation.

By default, the program pauses the execution until the user makes the choice. For configuring the unattended working, see the section How to automate the system reboot for processing the locked partitions.

3. The program silently schedules the *BlueScreen Component* to run at next Windows startup and to execute the required operation.
4. Then the program reboots the computer.
5. At the next system boot-up, the Bluescreen Component executes the operation in place of the Windows-based version. The BlueScreen Component will display the operation progress in the console-like style.

The detailed information about the interaction between Windows-based and Bluescreen components of Hard Disk Manager is discussed in the *Technician Manual*.

4.10.6.2 Copying locked partitions in Windows 95 and 98

Windows 95 and 98 include the limited version of the MS-DOS 7 as a part. The "true" DOS environment is available after booting to the DOS session (do not confuse with the *DOS prompt* in the Windows session).

1. Initially, the program tries to copy the partition without rebooting to DOS. The operation continues until the end or until another software writes to the original partition.
2. In case of the original partition is modified, the program asks for reboot the system. Press **OK** button to reboot the system and complete the operation, press **Cancel** button to abort the operation.
3. The application passes silently the task to the DOS-based version of the Hard Disk Manager. Then the Windows-based application just runs the DOS-based program.
4. The DOS-based version of the Hard Disk Manager is configured (through the .PIF-file) to run in the true DOS session. Windows should reboot to the DOS session.
5. The DOS-based program becomes working in the unattended mode with displaying the operation progress and statistics.
6. When completing the operation, the program reboots the computer.

4.10.6.3 Using the Hard Disk Manager diskette in Windows ME

The diskette-based version of Hard Disk Manager can work either in the interactive mode or in the batch mode. In both cases, it can copy partitions like the Windows-based version does.

There are some functionality limitations of the diskette-based version of Hard Disk Manager. These limitations coincide with the ones concerning the DOS-based version that works in the unattended mode (see Limitations of the DOS-based version in the unattended mode). These limitations come mostly from the unavailability some services in the DOS environment that are available in Windows:

- In the unattended mode, the program unconditionally aborts the operation in case of detecting bad sectors, missing the image file or similar problems. To get a friendly behavior of the Hard Disk Manager, run the program in the interactive mode.
- The diskette-based version may not work with SCSI and RAID controllers that have no own BIOS.
- Remember that DOS may fail to access to large partitions. For instance, MS-DOS does not work with partitions that are larger than 8Gb.
- Your computer must have the ability to boot from floppy.

The detailed information about interaction between Windows-based and the diskette-based versions of Hard Disk Manager is discussed in the *Technician Manual*.

The preferred solution is to use the Paragon Recovery CD instead of using the bootable diskette.

4.11 Copy Hard Disk

This chapter explains how to copy contents of local hard disks to another ones, under various conditions.

4.11.1 Overview

Hard Disk Manager provides the ability to copy not only separate partitions but also entire hard disks.

During hard disk copying, the program moves controlling records of used *partitioning scheme*, the bootstrap code and on-disk partitions. The *Hard Disk Copying* operation cannot be substituted with only copying all on-disk partitions.

Generally, Hard Disk Manager allows copying hard disks that are partitioned with any partitioning scheme, in the *sector-to-sector copying mode* (see Fast copying algorithm). For disks, which use *DOS partitioning scheme*, the program provides many functional extensions for the disk copying operation:

- Hard Disk Manager supports the fast copying mode for partitions of known filesystem types. This functionality significantly reduces elapsed time for the disk copying because the program transfers usable data only.
- In case of the disk contains partitions of both *known* and *unknown* filesystem types, the program automatically switches between fast copying and sector-to-sector modes.
- Hard Disk Manager provides an ability of automatic proportional *resizing* of all partitions, which have known filesystem types, during the operation. This functionality extension may be very useful in upgrading hard disk(s) to larger ones.

- Finally, Hard Disk Manager supports the disk copying with selective copying of partitions. Some partitions can be skipped, other ones can be copied multiple times, with changing their size, relative order and position.

The way Hard Disk Manager duplicates disks, allows successfully migrating some boot managing software to another hard disks. For example, standard bootstrap code, Paragon Boot Manager and LILO can successfully work on cloned hard disks without re-installation.

With *Copy Hard Disk* operation, the program irreversibly destroys old contents of the targeted hard disk, replacing them with the new data and new layout of partitions.

The implementation of the Copy Disk function varies in some cases:

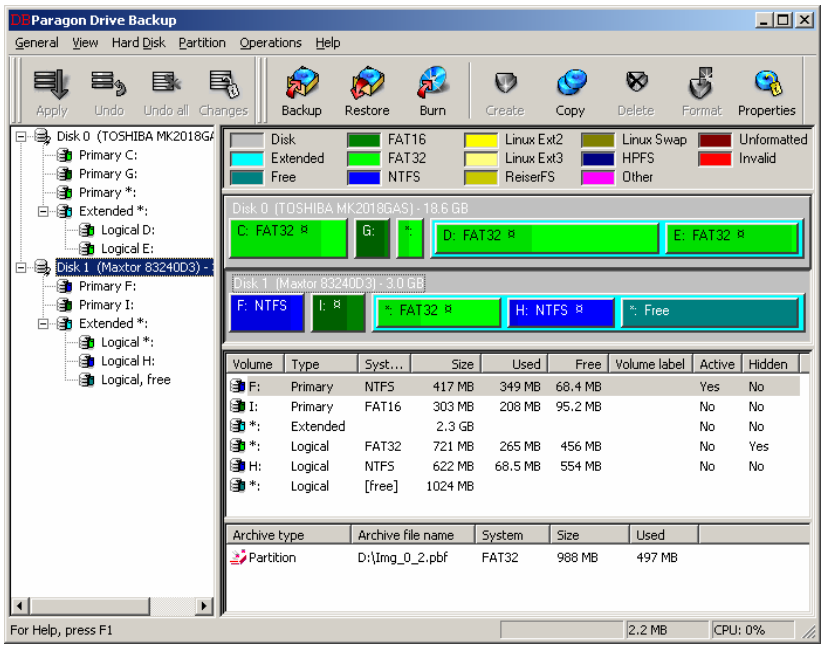
1. In Windows, the operation differs for *unlocked* and *locked* (system) hard disks; this mention concerns both source and targeted disks. To process locked hard disks, the program need to reboot the computer.
2. The processing of *locked disks* differs for Windows 95/98, Windows NT/2000/XP and Windows ME.

Note: locked hard disks are ones having locked partitions.

4.11.2 Initiating the operation in the Windows-based version

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

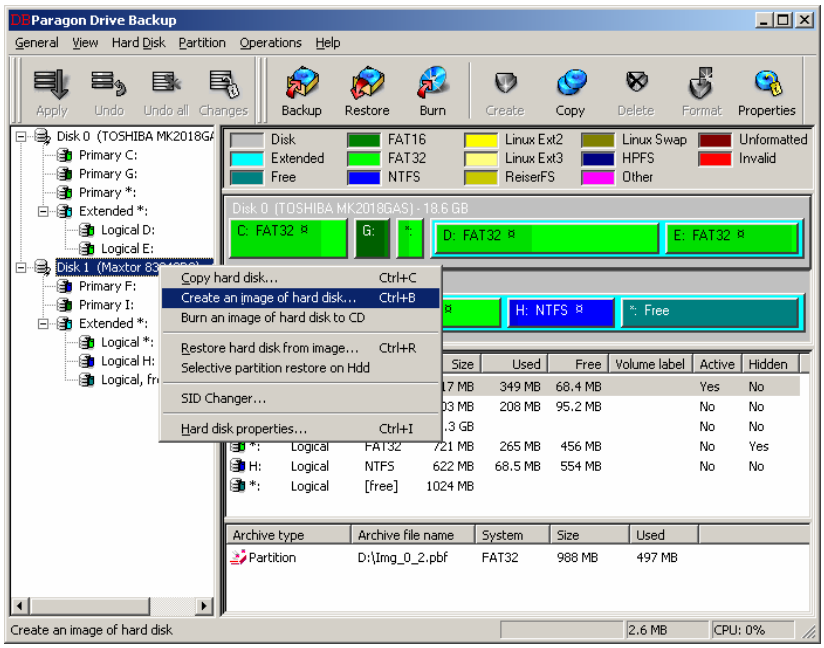
Step 1. Select the hard disk you want to backup.



There are two variants:

- select the hard disk in the Tree Layout panel or in the UDP Layout panel.
- select any partition that belongs the hard disk of interest.

Step 2. Select the operation to perform



Alternatives:

- Select in the main menu:

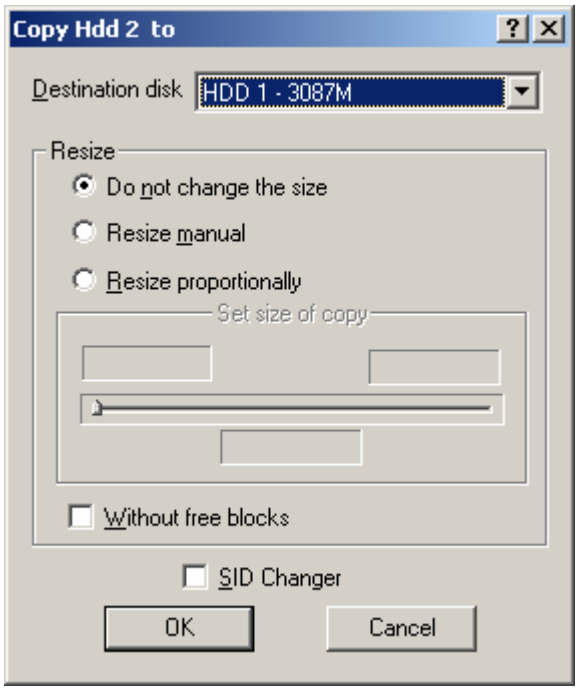
  Hard disk ➔ Copy hard disk…

- In case of the hard disk is selected, call the *popup menu* for the selected hard disk in any of layout panels (click right mouse button)
  then select the menu item:

  Copy hard disk…

- Press **Ctrl+C** keyboard combination
- Press **Copy** button on the Main Toolbar.

Step 3. Assign parameters of the operation



The program allows:

- Select a target hard disk

- Select a copying method
- Run SID Changer utility after completing the disk copying

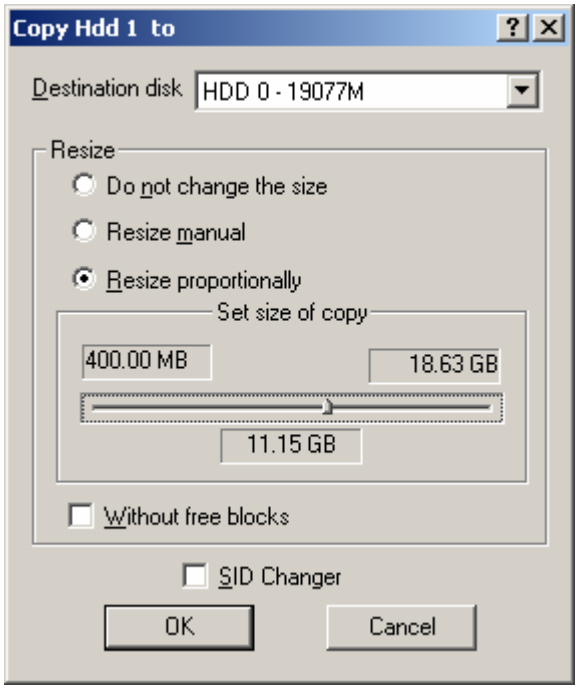
Initially the program suggests the simplest method of copying disk contents without changing size. In most cases, you just need to press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Copy Hard Disk* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.11.3 Description of the parameters



4.11.3.1 Common parameters

Destination disk

The pull-down list includes legal hard disks, which can be targeted for the operation. It contains all fixed hard disks available in the system, with the only exception of the source hard disk.

By default, the 1st hard disk in the list is selected.

If the source disk is any but the Disk#0 (Disk#0 is the 1st disk in the system), the Disk#0 is present in the top of the list of legal hard disks; so that it will be the default destination disk.

Take care of changing the targeted hard disk, if you do not intend overwriting all contents of the Disk#0.

SID Changer

Select this checkmark to force the program running SID Changer utility after completing disk copying. The feature is useful in case of cloning Windows system hard disks.

Without free blocks

This option takes effect in the "**Do not change the size**" and "**Resize proportionally**" copying methods (see the section Disk copying methods for more details).

With this option being active, the program does not leave blocks of free space between partitions on the targeted hard disk.

4.11.3.2 Disk copying methods

Do not change the size

Select this radio button to apply the method of copying without changing partitions size.

In this mode, Hard Disk Manager moves partitions with keeping their relative order, absolute size and position. See more details in the section Copying without changing partitions size.

Resize manual

Select this radio button to apply the method of *selective partitions copying*. The additional dialog window "**Copy HDD with manual resizing partitions**" will appear after pressing **OK** button.

In this mode, Hard Disk Manager allows changing size, position and relative order of partitions, skipping some partitions, or multiple duplicating other ones. The functionality is described in the section of Selective copying of partitions.

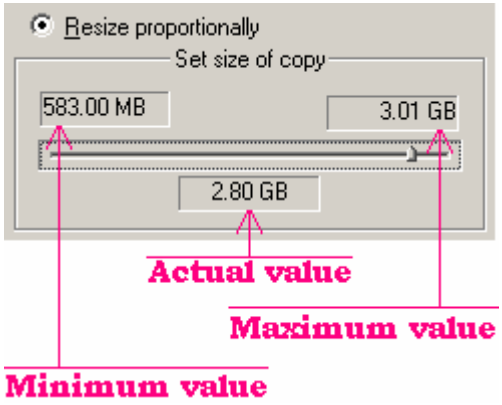
Resize proportionally

Select this radio button to apply the method of Copying with proportional changing partitions size.

In this mode, Hard Disk Manager changes the size of partitions in the same proportion, with keeping intact their relative order. This option can be useful in upgrading a hard disk to a larger one. See more details in the section Copying with proportional changing partitions size.

Set size of copy

The slider control allows assigning the range of the disk space that will be occupied on the destination disk with copied partitions.



- The *minimum value* (shown on the left side) is equal to the amount of the data currently allocated on the source hard disk.
- The *maximum value* (on the right side) corresponds the capacity of the destination disk.
- The *selected size* of the used range is displayed bottom the slider.

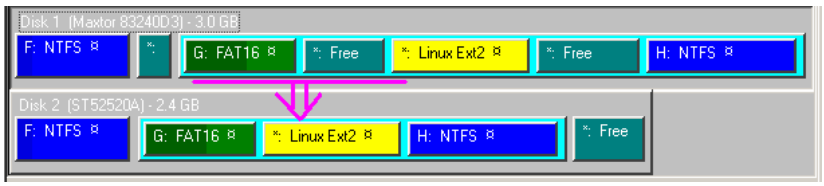
Rules the program use are described in following sections; in the *Virtual Execution mode* Hard Disk Manager allows to evaluate the resulting state of hard disks prior to real performing the operation.

4.11.3.3 Copying without changing partitions size

In this mode, Hard Disk Manager moves partitions with keeping their relative order, absolute size and position.

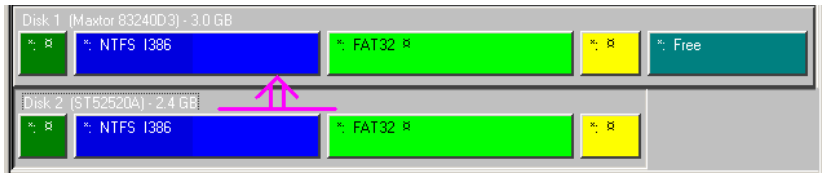
Removing free space between partitions

If the option "**Without free blocks**" is active, the program allocates copied partitions one by one, with removing free space between them. The rest of the destination disk may remain unpartitioned:



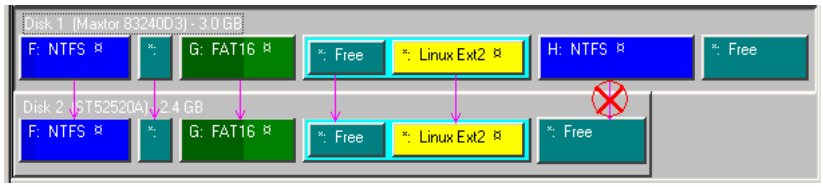
In case of the targeted disk is larger than the source one

Hard Disk Manager copies all source partitions with keeping their size, position and order. The rest of the disk space on the destination hard disk remains unpartitioned (free space):



In case of the targeted disk is smaller than the source one

Hard Disk Manager copies only those source partitions that fit the targeted disk, with keeping their size, position and order:



Copying the Extended Partition

As to Logical Partitions, the program will copy them only in the case the entire Extended Partition can fit to the targeted hard disk. Otherwise, the program skips copying the Extended Partition.

4.11.3.4 Copying with proportional changing partitions size

In this mode, Hard Disk Manager changes the size of partitions in the same proportion, with keeping intact their relative order.

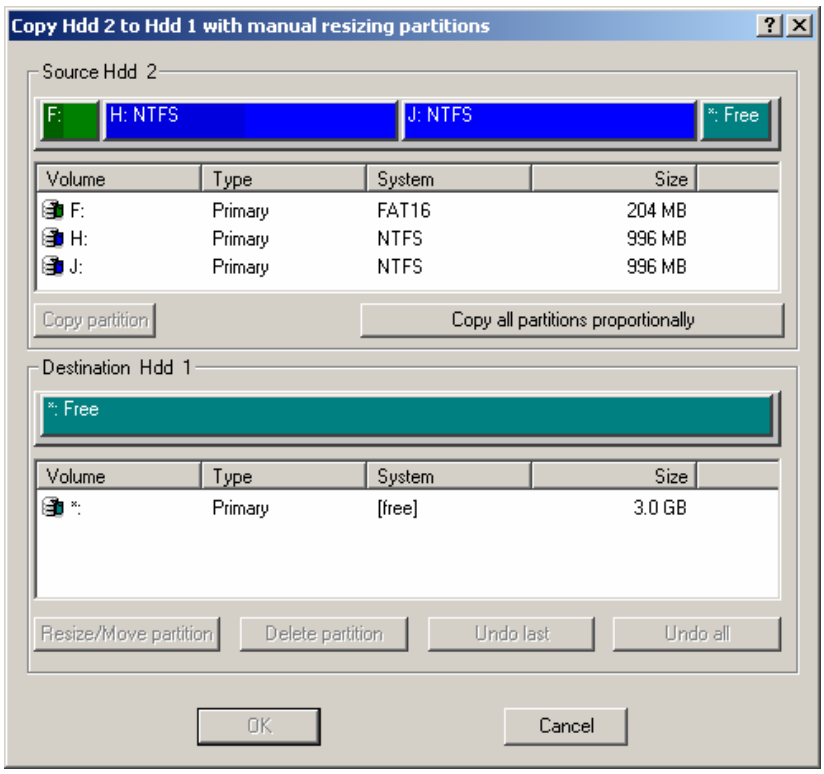
The slider control "**Set size of copy**" allows selecting the range of the disk space that will be occupied on the destination disk with copied partitions. This assigns implicitly the proportion of changing size of partitions.

If the option "**Without free blocks**" remains inactive, blocks of free space between partitions will be changed in the same proportion.

4.11.4 Selective copying of partitions

In this mode, Hard Disk Manager allows changing size, position and relative order of partitions, skipping some partitions, or multiple duplicating other ones.

If this copying method is selected, the additional dialog window "**Copy HDD with manual resizing partitions**" will appear after pressing **OK** button in the "**Copy HDD**" dialog:



This dialog is functionally similar to the "**Selective Partition Restore**" dialog (see the chapter Description of the Selective Partition Restore dialog functionality).

The dialog maintains own list of virtual operations: some or all partitions can be virtually copied to the destination disk in any order; at any time, any of copied partitions can be resized, moved or deleted from the destination disk. Changes in the layout of the destination disk can be cancelled.

After a user has confirmed changes, the program analyzes the final layout of the destination disk and generates the optimal sequence of actions, with ignoring unnecessary ones.

4.11.4.1 Description of parameters

"Source HDD" and "Destination HDD" groups of controls

In the **Source HDD** group, the *UDP control* and the *List of Partitions* represent the layout of the source hard disk. Any of these elements can be used for selecting source partitions.

In the **Destination HDD** group, the *UDP control* and the *List of Partitions* represent the future layout of the targeted hard disk. Any of these elements can be used for selecting blocks of free space and partitions on the destination disk.

Copy partition

Press this button to copy a single partition from the source disk to the destination one. The button is enabled only in case of selecting the source partition in the **Source HDD** group (above) and the block of free space in the **Destination HDD** group (below).

The action can be cancelled with the **Undo last** button (see below).

Destination partitions are marked with symbolic labels like "_1_", "_2_" etc. These labels only display a number of a corresponding source partition.

Copy all partitions proportionally

Press this button to copy all contents of the source disk with keeping size proportions for partitions of known types and blocks of free space.

This action cancels all changes previously made. The destination disk becomes in the state it would be after executing the copying with proportional resizing. Next, you are able to delete or resize some destination partitions.

This action is very useful, if you need copy entire hard disk having many partitions, with only minor changes of the disk layout.

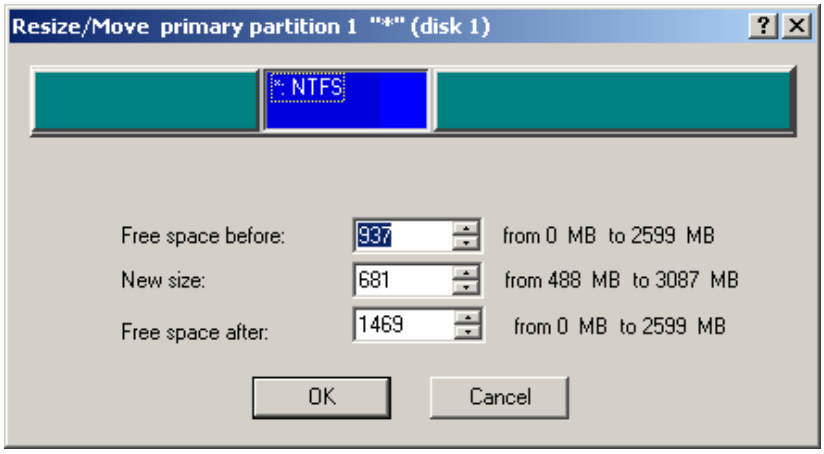
The action can be cancelled with the **Undo last** button (see below).

Resize/Move partition

Initially, a source partition that is copied to a block of free space on the destination disk, is aligned to the beginning of the selected free block, and has the original size.

Press the button "**Resize/Move partition**" to change the size and/or position of a partition on the destination disk. The button is enabled only in case of selecting some partition in the **Destination HDD** group (below).

In this case, the program will display the *Resize/Move partition* dialog:



The functionality of this window is the same with similar window in functions Restore Partition, Selective Partition Restore, Copy Partition.

Use three spin controls or the drag-&-drop technique over the UDP control to change size and position of a destination partition (see Defining the location and size of the restored partition for more details).

The UDP control and spin controls are synchronized, the changing of any of these elements affects on all other ones.

How the spin controls behave:

Free space before	Moves the beginning of the partition (left edge), preferably with keeping the partition size.
New Size	Changes the size of the partition, preferably with keeping the starting position (left edge).
Free space before	Moves the end of the partition (right edge). On increasing the value, it (preferably) keeps the partition size. On decreasing the value, it (preferably) keeps the starting position (left edge) so that the partition expands.

There are rules that take effect in the partition resizing. See the section How Hard Disk Manager evaluates space available for partition copying for more details.

Delete partition

Press this button to delete a partition from the destination disk. The button is enabled only in case of selecting some partition in the **Destination HDD** group (below).

The action can be cancelled with the **Undo last** button (see below).

Undo last

Press this button to cancel the last change in the layout of the destination disk.

Undo all

Press this button to cancel all changes in the layout of the destination disk. After this action, the destination disk becomes empty.

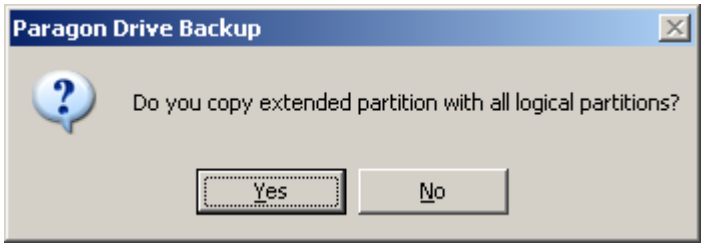
Edit the destination disk layout with using drag-&-drop technique

The *UDP control* in the **Destination HDD** group supports the drag-&-drop technique for changing size and position of partitions on the destination disk.

4.11.4.2 Copying the Extended Partition

Hard Disk Manager allows copying the Extended Partition as a whole.

In case of copying the Extended Partition, the program asks for copying all Logical Partitions together with the Extended Partition:

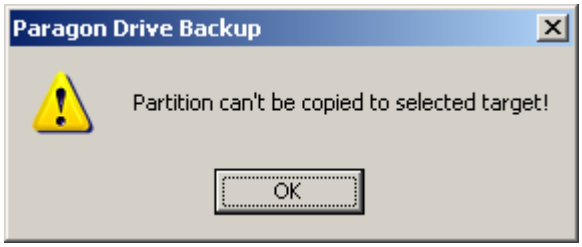


If **Yes** answer is selected, the program copies the Extended Partition with all Logical Partitions inside. Otherwise, the empty Extended Partition will be created on the destination disk; later, you can populate it with selective partitions.

Hard Disk Manager allows copying *primary partitions* to the Extended Partition, so that they become *logical* ones, and vice versa.

4.11.4.3 Partition copying rules

- The source partition can be copied to the destination free block only in case of the size of the free block exceeds the amount of used space for the selected partition.
  Otherwise, the error message appears:



- The source Extended Partition can be copied to the destination free block only in case of the size of the free block exceeds the summary amount of used space for all logical partitions.
- If the destination free block is less than the original size of the source partition, the program squeezes the source partition so that it fits the free block. In case of the Extended Partition is selected, all Logical Partitions will be squeezed proportionally.

4.11.5 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed and summarized speed of the operation

The program treats copying of each partition as a suboperation, and the program displays suboperation statistics.

The operation is lengthy. Real performance fundamentally depends on hardware and an operating system being used.

4.11.6 Comments

4.11.6.1 The source disk contains partitions of unknown types

Hard Disk Manager can only copy partitions of *unknown types* in the *sector-to-sector mode*, with keeping their size.

When using the *copying with proportional resizing* to a hard disk, which has *unknown* partitions, Hard Disk Manager moves unknown partitions without changing their size. The program re-calculates the resizing rate for partitions of known types and blocks of free space.

4.11.6.2 Cloning the system hard disk in Windows 2000 and XP

Hard Disk Manager is frequently used for cloning or distributing system hard disks. In case of performing this action with Windows 2000/XP system disk, the specific procedure is required in order to avoid problems with the duplicate hard disk.

1. Connect both source and destination disks to the computer. Boot the computer and run Hard Disk Manager (any version, for any platform).
2. With using Hard Disk Manager, clone the source disk to the destination one.
3. **(!)** Shutdown the computer.
4. **(!)** Disconnect (physically) the source hard disk.
5. **(!)** Boot the computer from the destination hard disk. No problems should appear during the startup process.

After completing these actions, you can use both hard disks separately or together on the same computer without problems.

The thing is that Windows 2000 and XP keep information about all mounted and dismounted partitions ever on all hard disks ever connected to the system in a special database. Partitions are identified by the hard disk serial number and the relative partition's order. The database of partitions is periodically updated. During cloning hard disks, this database is duplicated on the destination disk.

Usually, users do not perform two last steps, i.e. they remain the source hard disk connected at the most first startup from the destination disk. At the startup, Windows finds the old hard disk, retrieves drive letters to its partitions from the old records, adds new drive letters for partitions on the destination disk, and updates the database.

After the source disk has been disconnected, the single destination disk becomes unusable: Windows is unable to boot from this disk because of some important system data become associated with absent drives.

4.11.7 Working with locked and system hard disks

The *Copy Hard Disk* operation requires both source and destination hard disks should be unlocked ones (see Glossary):

- The program needs to destroy contents of the destination disk in the beginning of the operation. If some partitions on the destination disk are locked, that means that an operating system or other applications still use this partition. The untimely deletion of disk contents may violate working of other software.
- The program requires contents of the source disk remain unchanged during data copying. If some partitions on the source disk are locked, that means that other software can modify disk contents. In this case, the program cannot securely copy disk contents.

To complete the disk copying of/to a locked/system hard disk, Hard Disk Manager reboots the system to a single-tasking environment in order to eliminate the interference of other programs.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

Generally, the copying of locked hard disks is mostly like the copying of locked partitions (see the section Backup system and locked partitions for more details).

4.11.7.1 Copying locked hard disks in Windows NT, 2000 and XP

In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase to operate the locked partitions:

1. Before starting the operation, the program checks whether any locked partitions on both source and destination hard disks.
2. In case of some partition(s) appears locked, the program asks for reboot the system. Press **OK** button to reboot the system and complete the operation, press **Cancel** button to abort the operation.

By default, the program pauses the execution until the user makes the choice. For configuring the unattended working, see the section How to automate the system reboot for processing the locked partitions.

3. The program silently schedules the *BlueScreen Component* to run at next Windows startup and to execute the required operation.
4. Then the program reboots the computer.
5. At the next system boot-up, the Bluescreen Component executes the operation in place of the Windows-based version. The BlueScreen Component will display the operation progress in the console-like style.

The detailed information about the interaction between Windows-based and Bluescreen components of Hard Disk Manager is discussed in the *Technician Manual*.

4.11.7.2 Copying locked hard disks in Windows 95 and 98

Windows 95 and 98 include the limited version of the MS-DOS 7 as a part. The "true" DOS environment is available after booting to the DOS session (do not confuse with the *DOS prompt* in the Windows session).

1. Before starting the operation, the program checks whether any locked partitions on both source and destination hard disks.
2. In case of some partition(s) appears locked, the program asks for reboot the system. Press **OK** button to reboot the system and complete the operation, press **Cancel** button to abort the operation.
3. The application passes silently the task to the DOS-based version of the Hard Disk Manager. Then the Windows-based application just runs the DOS-based program.
4. The DOS-based version of the Hard Disk Manager is configured (through the .PIF-file) to run in the true DOS session. So that Windows reboots to the DOS session.
5. The DOS-based program becomes working in the unattended mode with displaying the operation progress and statistics.
6. When completing the operation, the program reboots the computer.

4.11.7.3 Copying locked hard disks in Windows ME

The diskette-based version of Hard Disk Manager can work either in the interactive mode or in the batch mode. In both cases, it can copy partitions like the Windows-based version does.

The procedure of processing locked partitions and hard disks is described in the section Working with locked partitions/disks in Windows ME.

There are some functionality limitations of the diskette-based version of Hard Disk Manager. These limitations are listed in the section Backup locked partitions in Windows ME.

The preferred solution is to use the Paragon Recovery CD instead of using the bootable diskette.

4.12 Create Partition

This chapter explains how to create new partitions with using Hard Disk Manager; in addition, it contains the description of partitioning rules and limitations.

4.12.1 Overview

Hard Disk Manager provides the ability to create new partitions on hard disks that are partitioned with using the *DOS partitioning scheme* (see Glossary).

4.12.1.1 Limitations

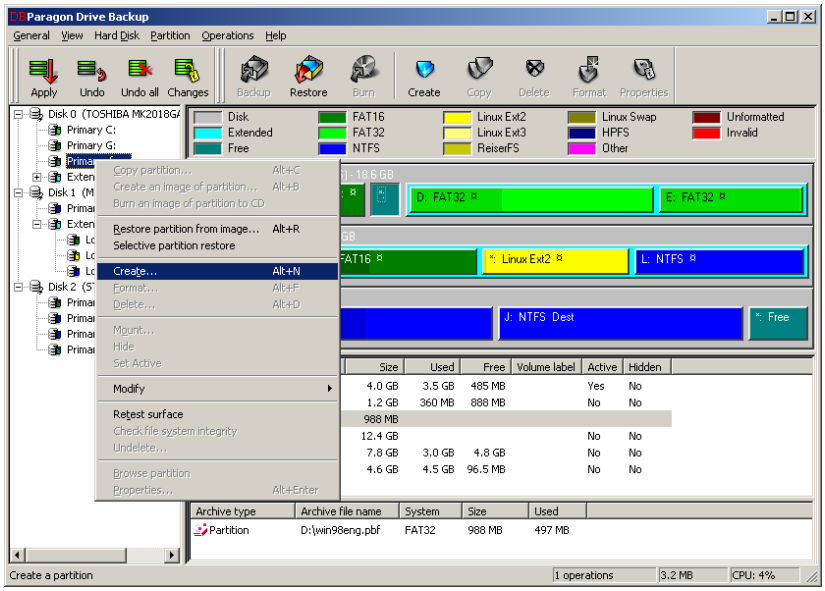
1. The program allows creating new partitions only within blocks of unpartitioned space. It cannot "convert" a free space on an existing partition to a new partition!
   The procedure of creating a new partition from a free space on an existing partition is described in the section How to make a new partition from a free space in another partition.
2. Do not use the *Create Partition* function in order to undelete the just deleted partition. Use the Undelete Partition function instead.
3. The program cannot create new partitions on *Dynamic Disks*. Current version of the program supports only hard disks that use the *DOS partitioning scheme* (in Windows 2000 and XP these disks are named *Basic Disks*).
4. Empty hard disks are automatically become partitioned with the *DOS partitioning scheme*, in case of creating a partition of any type.
5. New partitions are aligned to the boundaries of the beginning and ending cylinders in accordance with the rules of the *DOS partitioning scheme* (see Partition alignment rules).

4.12.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a location of a new partition

Select a block of free space (unpartitioned disk space) in the Tree Layout panel, or in the UDP Layout panel, or in the List of Partitions. The newly created partition will be located within the bounds of the selected free block.



The program differentiates between *Primary* free space and *Logical* free space:

- Within the *Logical* free space, only *Logical* partitions can be created
- Within the *Primary* free space, *Primary* partitions or the Extended Partition can be created, only in case of the partitioning rules are not violated (see Limitations of the DOS partitioning scheme.)

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Create …

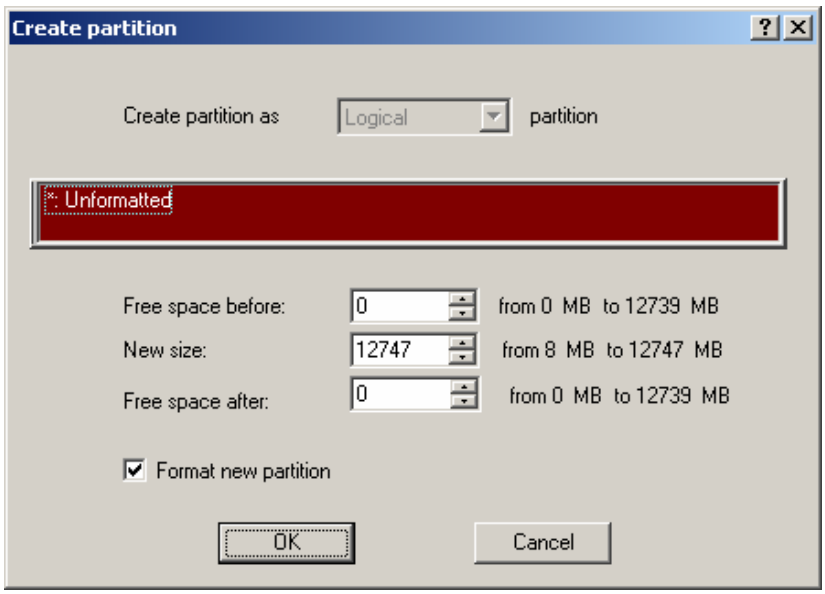
- Call the *popup menu* for the selected block of free space in any of layout panels (click right mouse button), then select the menu item:

  Create…

- Press **Alt+N** keyboard combination
- Press **Create** button on the Main Toolbar.

Step 3. Assign properties of a new partition

After selection a free block, the *Create Partition* dialog appears:



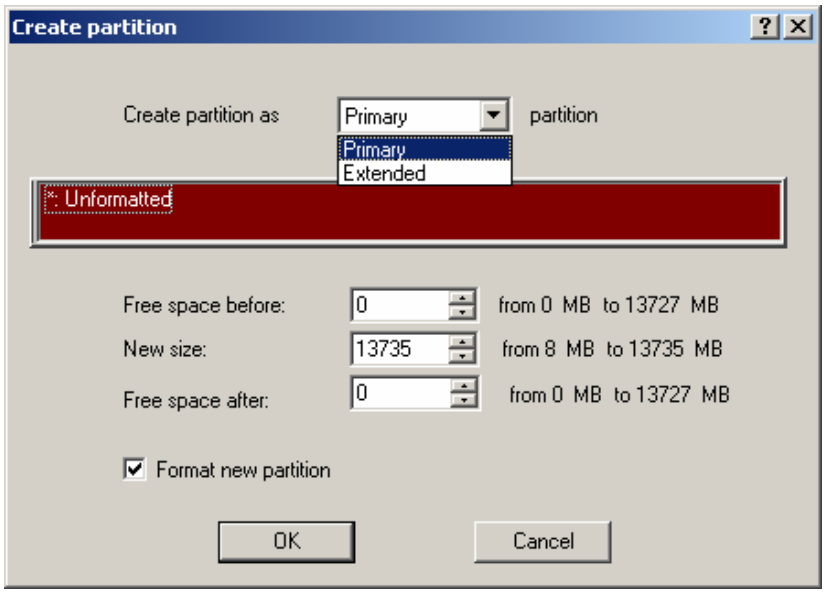
The *Create Partition* dialog allows assigning the accurate position and size of a new partition. In addition, there is the ability to format the new partition just after creation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Create Partition* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.12.3 Description of the parameters



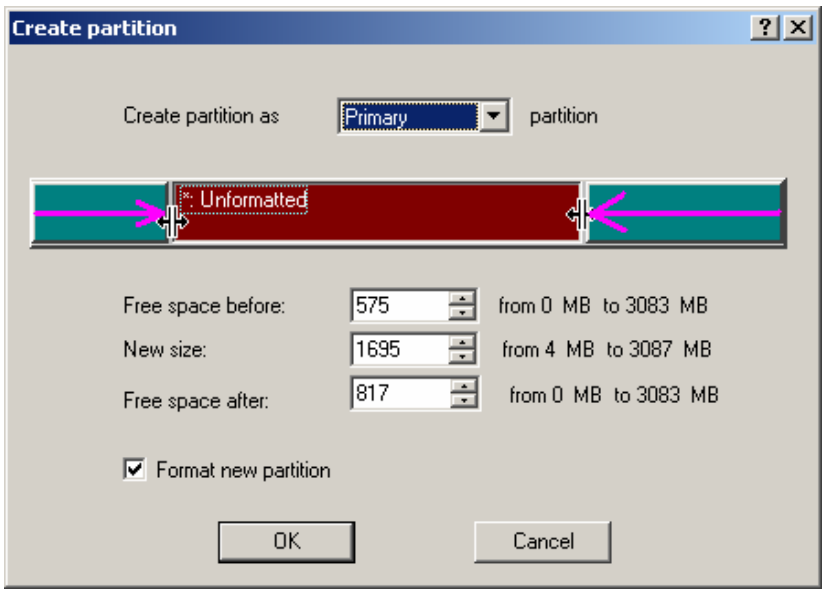
Create partition as …

Select the desired kind of a new partition in this pull-down list. As the matter of fact, the available alternatives fundamentally depend on the type of the selected block of free space:

- Within the *Logical* free space, only *Logical* partitions can be created
- Within the *Primary* free space, *Primary* partitions or the Extended Partition can be created (in case of the partitioning rules are not violated, see Limitations of the DOS partitioning scheme.)

New Size, Free Space Before, Free Space After

The capacity of the new partition must not exceed the size of the selected block of free space. You can freely define the position and size of the new partition within the targeted block of free space by using the UDP control that is located in the window:



In addition, you can use three *spinner* controls placed on the bottom:

- The **Free space before** spinner control defines the position (in Mb) of the new partition relative to the beginning of the block of free space.
- The **New size** spinner control defines the size (in Mb) of the new partition.
- The **Free space after** spinner control defines the amount of trailing free space (in Mb) at the end of the new partition.

How the spin controls behave:

Free space before	Moves the beginning of the partition (left edge), preferably with keeping the partition size.
New Size	Changes the size of the partition, preferably with keeping the starting position (left edge).
Free space before	Moves the end of the partition (right edge). On increasing the value, it (preferably) keeps the partition size. On decreasing the value, it (preferably) keeps the starting position (left edge) so that the partition expands.

The *UDP control* and *spinners* are synchronized, the changing of any of these elements affects on all other ones.

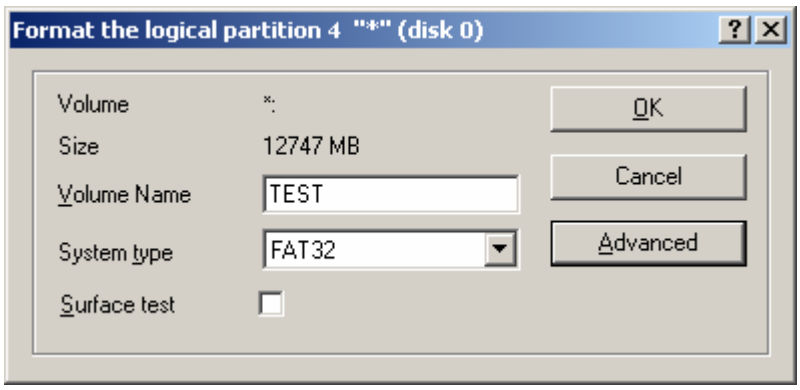
At the real partition creation, the program will round these values to accommodate user-defined values to the partitioning rules that are effective for the *DOS partitioning scheme*. So that the final values may slightly differ from the entered ones.

Format new partition

The program allows to populate the new partition with some filesystem.

Set this checkmark ON in order to immediately format the newly created partition. Otherwise, the partition will remain unformatted (so that it will not be ready for use).

The program allows defining parameters of a filesystem that will occupy the new partition by calling the *Format Partition* dialog (see the next section).

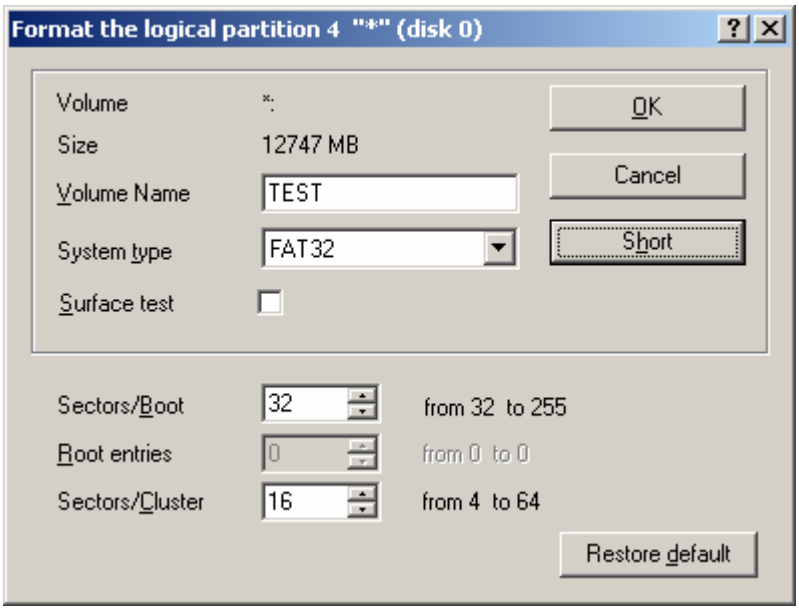


4.12.3.1 Formatting a new partition

The appearing *Format Partition* dialog is identical to the one described in the chapter Format Partition.

The simple form of the dialog allows selecting the Filesystem Type and defining the so-called *Volume Label* for the partition.

The advanced form of the dialog allows defining important filesystem properties such as *Cluster Size* (**Sector/Cluster** spinner control) and amount of sectors per boot code (**Sector/Boot** spinner control). For FAT16 filesystem, the amount of *Root Entries* can also be changed:



4.12.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time

The program treats the updating of system information as a suboperation, and the program displays suboperation statistics.

The creation of a partition takes just a fraction of a second. Still, the program waits until Windows accommodates changing of the disk layout. This operation may take 5-20 seconds in Windows 2000 and XP.

4.12.5 Comments

4.12.5.1 How to make a new partition from a free space in another partition

Hard Disk Manager does not provide a function like "pick-up free space from existing partition(s) and create one more partition". One should manually define a sequence of resize/move operations on existing partitions in order to free disk space and collect it into a single block of unpartitioned space.

Example

Suppose you have one primary and one logical partition on the hard disk, and you need to create one more primary partition for the isolated installation of another operating system, at the expense of unused space on the Logical Partition.

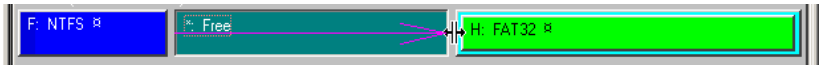
The original disk layout is shown below:

Step 1. Resize&Move the Logical Partition

Move "forward" the starting edge of the Logical Partition. With this action, you are releasing some disk space:

Step 2. Resize&Move the Extended Partition

Move "forward" the starting edge of the Extended Partition. The block of unpartitioned space is moved outside the Extended Partition, so that a newly created partition will become a primary one:



Step 3. Create the new Primary Partition

Create the new Primary Partition in the block of unpartitioned space:

4.12.5.2 Limitations of the DOS partitioning scheme

Current version of Hard Disk Manager fully supports only the *DOS partitioning scheme*. The program follows limitations of the DOS partitioning scheme at the creating of new partitions.

Practical limitations

- Only one Extended Partition can be created on a hard disk.
- Only four Primary Partitions can be created on a hard disk, or three Primary Partitions and one Extended Partition.
- The amount of Logical Partitions within the Extended Partition is not limited.
- All newly created partitions are aligned in accordance with rules of the DOS partitioning scheme and the current disk geometry (see the section Partition alignment rules of the DOS partitioning scheme).
- The program does not change the order of *entries* in the *Partition Table* for existing partitions (see the section Order of Partitions).

4.12.5.3 General structure of the DOS partitioning scheme

The DOS partitioning scheme keeps the information about partitions located on the disk, in the *Partition Table*. The *Partition Table* consists of elements describing partitions: location, size, filesystem type and "Active" flag. These elements are usually named *entries* or *slots* of the Partition Table.

The first sector of the hard disk (sector #0) is occupied with the *Master Boot Record* (MBR). It contains the primary part of the Partition Table and the so-called *bootstrap code* – a very small program responsible for initiation of the boot-up process.

Primary Partitions

The primary part of the Partition Table contains only 4 entries. The partitions that are registered in these records are named *Primary Partitions*. There can be only four primary partitions on the disk.

The specific property of primary partitions is that the standard *bootstrap* is able to start operating systems from them.

Extended Partition

One of the primary partitions can be marked as the *Extended Partition*.

The Extended Partition is used for expanding the Partition Table to provide the ability of allocating many partitions. The Extended Partition begins with the extension of the Partition Table, which sometimes is named *Extended Partition Table* (EPT).

Logical Partitions

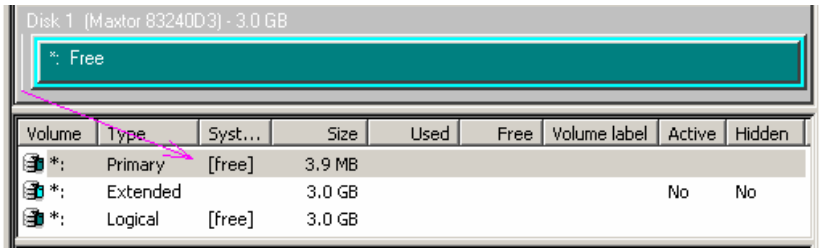
Partitions that are registered in the Extended Partition Table are named *Logical Partitions*. All Logical Partitions must be nested within the Extended Partition. The amount of logical partitions is not limited.

4.12.5.4 Partition alignment rules of the DOS partitioning scheme

When creating new partitions, Hard Disk Manager follows common *partition alignment rules* that are compatible with all operating systems and disk management utilities.

Alignment rules

1. The *ending edge* of every partition is aligned to the end of the appropriate cylinder.
2. The Primary Partition, which is located at the beginning of the hard disk, starts from the 1st Track of the 0th Cylinder. The 0th Track is occupied by the *MBR* and boot managing software.
3. Other Primary Partitions start from the beginning of appropriate cylinders.
4. Logical Partitions start from 1st Tracks of appropriate cylinders. 0th Tracks of these cylinders are occupied by the *EPT* records.
5. The Extended Partition should start from the beginning of the first cylinder it occupies. In case of the Extended Partition is located at the beginning of the disk, it starts from the 1st Cylinder (see the picture below) because the beginning of the 0th Cylinder is occupied by MBR.



An unused part of the 0th Cylinder potentially can be used for creation of a small Primary Partition.

4.12.5.5 Order of Partitions

The *order of partitions* may affect on the partition accessibility and on the assignment of drive letters in various families of operating systems: DOS, Windows and Linux are among them. The creation of a new partition can change their relative order and violate references to partitions.

Partitions can be enumerated by their natural location on a disk, or by the index of respective *entries* in the *Partition Table*. Generally, these enumerations are not same: the order of entries coincides with the natural order of partitions for Logical Partitions, but this is not true for Primary Partitions.

From this point, problems come up:

- Linux enumerates partitions according to the order of entries. The creation of a new partition in front of existing partitions changes references to sequent partitions on the disk. These changes take effect at the system reboot.

Disk	Partition	Symbolic name in Linux
1st	1st primary	/dev/hda1
	2nd primary	/dev/hda2
	1st logical	/dev/hda5
2nd	1st primary	/dev/hdb1
	1st logical	/dev/hdb5

- Windows 95, 98 and ME use rather intricate algorithm for assigning drive letters to partitions of supported type (see the chapter Mount Partition➔Comments). A creation of a new FAT16/FAT32 partition on any of hard disks may shift drive letters, after the system reboot. This operation does not affect drive C: only.
- Generally, Windows NT, 2000 and XP save the physical location of all partitions in the Registry and use this information for assigning drive letters. The creation of a new partition does not violate drive letters of *mounted partitions*.

There is another problem with Windows NT, 2000 and XP. The OS loader (NTLDR in these systems) enumerates primary partitions by the number of respective entry in the Partition Table. Partitions numbers are used by NTLDR for

locating the *Windows System Partition* during the startup. The creation of a new primary partition in rare situations may lead to inability to start Windows NT/2000/XP. It is impossible to observe all probable scenarios, but the creation a new primary partition **before** the system partition is potentially dangerous operation!

Hard Disk Manager provides the functionality of controlling the order of entries in the Partition Table (see the function Change Primary Slot). In addition, the problem of changing the order of primary partitions can be solved by editing the BOOT.INI system file that is responsible for configuring NTLDR (see Glossary).

Hard Disk Manager does not change the order of entries in the Partition Table. Such logic allows avoid the problem mentioned above in many cases.

The other side of the problem is that *Windows Disk Administrator* system utility automatically re-sorts entries in the Partition Table according to the natural order of partitions, without making corrections in system files. In case of creating new primary partitions with using Windows Disk Administrator, and in case of alternate use of both Hard Disk Manager and Windows Disk Administrator, Windows bootable files may become disorganized.

4.13 Format Partition

This chapter explains how to format existing or newly created partitions with using Hard Disk Manager; in addition, it discusses some filesystem limitations related to the *Format* function.

4.13.1 Overview

A partition should contain some *filesystem* to be used for keeping data; a partition itself is only a continuous range of disk space that is marked as "exploitable" in the *Partition Table* (see Glossary).

The process of installing of a filesystem is commonly named *format*. There is a large variety of filesystems developed. Hard Disk Manager can format partitions to the following filesystems:

- FAT12 & FAT16
- FAT32
- NTFS
- Ext2
- Ext3
- ReiserFS
- Linux Swap v. 2
- HPFS

4.13.1.1 Limitations

The current version of Hard Disk Manager creates NTFS ver. 1.2 (it corresponds to Windows NT 4.0 abilities). For this reason, *Access Quotas* feature is not available on NTFS partitions formatted with Hard Disk Manager. To get *Access Quotas* feature available, use Windows 2000/XP FORMAT tool instead.

4.13.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a partition to be formatted

Select an existing *Primary* or *Logical* partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. The operation is not enabled for the Extended Partition and for blocks of free space.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Format…

- Call the *popup menu* for the selected partition, then select the menu item:

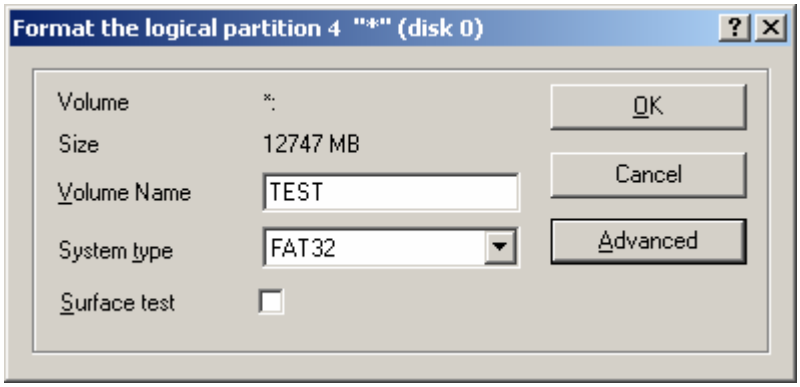
  Format…

- Press **Alt+F** keyboard combination

- Press **Format** button on the Main Toolbar.

Step 3. Assign format properties

After selection, the *Format Partition* dialog appears:



The *Format Partition* dialog allows defining the label of the partition and choosing the filesystem type, which should be placed on the partition.

In addition, there is the ability to control important parameters of the installed filesystem such as *cluster size*, amount of entries in the *Root* directory and size of the on-partition boot area.

Step 4. Apply the operation

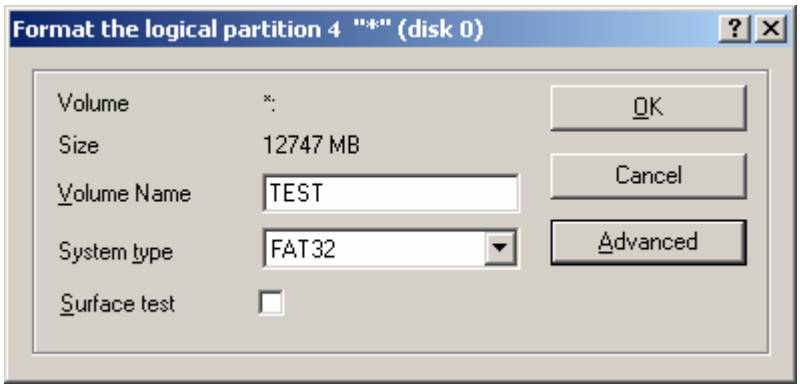
Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Format Partition* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.13.3 Description of the parameters

The program provides the *Simple* and the *Advanced* forms of the *Format Partition* dialog.

The *Simple* form allows only selecting the filesystem type and define the volume name:



Volume Name

Enter the *Volume Name* for the selected partition in this textual field. The Volume Name an unimportant parameter of a logical drive that can be used for drive identifying. See the chapter Glossary for more details about *Volume Name*.

System type

Select the desired filesystem type from this pull-down list. By default, the program suggests to keep the Filesystem Type and the Volume Name.

The program allows creating following filesystems:

- FAT12 & FAT16
- FAT32
- NTFS
- Ext2
- Ext3
- ReiserFS
- Linux Swap v. 2
- HPFS

In fact, the program displays only filesystems that can be correctly placed to the selected partition, with taking account the capacity of the selected partition. For example, the program will not allow FAT16 filesystem for partitions that are larger than 2Gb, or FAT32 for partitions that are less than 128Mb. See the section Capacity limitations for different filesystems for more details.

Surface test

Set this checkmark so that the program performs the surface test on the formatted partition. In this case, the program will find bad and unreliable sectors and mark them as unusable in the *filesystem metadata*.

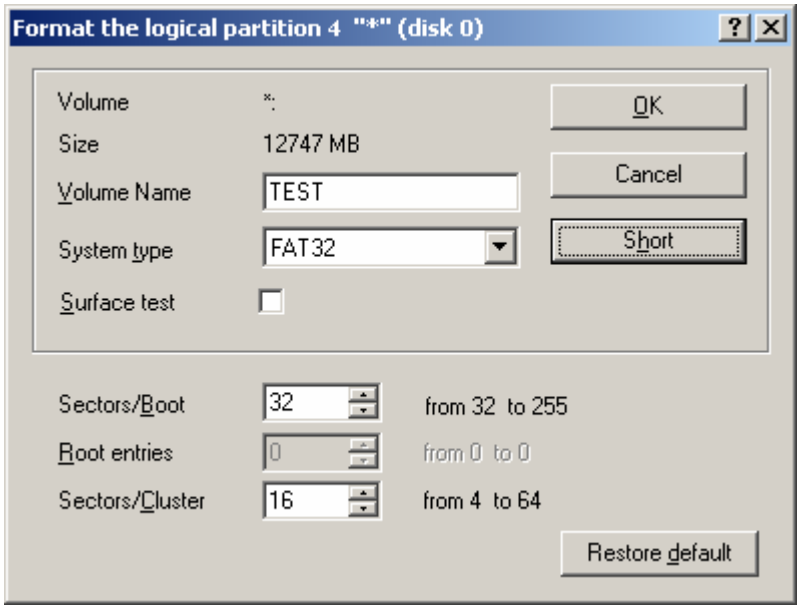
The program supports several levels of the surface test thoroughness (see the description of the *Surface test* option in the chapter Settings overview, section General Page):

Surface Test used during format	Surface Test level in the Settings	Description
Normal	None	No surface test
Normal	Normal	Single-pass read test
Extreme	Extreme	Three-pass read & write test

Advanced (button)

Press this button to get access to advanced format features. The Advanced form of the *Format Partition* dialog allows controlling some important parameters of the installed filesystem:

- Cluster Size
- Amount of entries in the Root Directory
- Size of Boot Area



Sectors/Boot

This parameter is available for FAT16 and FAT32 filesystems only.

Set the amount of sectors to be reserved for the boot area on the partition in this *spinner* control. The program relies on the rules of FAT filesystems when determining the available range and default value for this parameter.

Root entries

This parameter is available for FAT16 filesystem only.

Set the maximum amount of files/directories that can be placed to the Root Directory on the FAT16 partition. The program relies on the rules of FAT16 filesystem when determining the available range and default value for this parameter.

In FAT12 and FAT16 filesystems, the Root Directory is the very specific one. The capacity of the Root Directory cannot be changed until re-formatting the partition.

For example, you have formatted FAT16 partition so that the capacity of the Root is 64. Then, if you populate this partition with files, you are unable placing more than 64 files in the root of the volume. The only way to place more files is to create directories and place files inside. Still, you're again unable to make more than 64 directories in the Root directory.

Sectors/Cluster

This parameter is available for all known filesystems (excepting HPFS and Linux Swap 2).

Define the *Cluster Size* for the formatted partition in this *spinner* control. The program relies on the rules of these filesystems when determining the available range and default value for this parameter.

Hard Disk Manager reports the Cluster Size value as the *Sectors per Cluster* ratio. To get the Cluster Size in Kb, just halve this value.

It is the important filesystem parameter: the *Cluster Size* affects on such significant characteristics as the input-output performance of file operations on the volume, the waste space percentage and so on. In Windows 2000 and XP, some advanced features of the NTFS filesystem are available only in case of *Cluster Size* is 4Kb or less.

Restore default (button)

Press this button to roll back to the default values of the *Cluster Size*, *Amount of Root Entries* and *Size of Boot Area* parameters.

Short (button)

Press the button to return to the Simple form of the *Format Partition* dialog. In the simple form, the Format Partition function applies the default values for the *Cluster Size*, *Amount of Root Entries* and *Size of Boot Area* parameters for the partition being formatted.

4.13.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed

The program treats the Surface Test and the updating of system information as suboperations, and the program displays suboperation statistics.

The formatting itself takes just few seconds: it includes only a creation of clean filesystem metadata and empty Root Directory.

However, the Surface Test suboperation can take large amount of time. The real value of the elapsed time fundamentally depends on the size of the new partition, hardware performance and the software platform being used.

On modern disk models, the *Normal* surface test may take 0.5 - 2 minutes per 1Gb of capacity. For the *Extreme* surface test, the elapsed time is approximately 6 times more.

4.13.5 Comments

4.13.5.1 Capacity limitations for different filesystems

Filesystems have maximum capacity, different for each filesystem. The maximum capacity is derived from the maximum cluster size available in the filesystem and from the maximum amount of clusters. The last value depends on the bit capacity of cluster index in the filesystem.

Filesystem	Max Cluster	Max amount of clusters	Max partition capacity
FAT16	32K	2^{16}	2 Gb
FAT32	32K	2^{28}	8 Tb
NTFS	32K	2^{64}	~$6*10^8$ Tb
Ext2/Ext3	4K	2^{32}	16 Tb
ReiserFS	8K	2^{64}	~$1.5*10^8$ Tb

4.13.5.2 Default Cluster Size values

Capacity	NTFS		FAT32
< 512 Mb	0.5 K	(1 sector/cluster)	4K
< 1 Gb	1 K	(2 sectors/cluster)	4K
< 2 Gb	2 K	(4 sectors/cluster)	4K
< 4 Gb	4 K	(8 sectors/cluster)	4K
> 4 Gb	8 K	(16 sectors/cluster)	4K
> 8 Gb	16 K	(32 sectors/cluster)	8K
> 16 Gb	32 K	(64 sectors/cluster)	16K

Windows 200 and XP provide advanced service for NTFS partitions such as contents compression and encryption. These services are available only on partitions that have cluster size 4Kb or less.

In case of formatting NTFS partitions with using Hard Disk Manager, take care of changing the cluster size from the default value to 8 sectors/cluster in order to use advanced capabilities of NTFS partitions.

The table above exhibits default *Cluster Size* values for partitions of different size. NTFS partitions, which require manual diminishing of cluster size for enabling advanced features of NTFS, are yellow-shaded.

4.13.6 Formatting locked and system partitions

Hard Disk Manager allows to format locked and system partitions. This operation requires rebooting the computer to be completed. The partition formatting unconditionally destroys an existing on-partition filesystem, so that it violates any unfinished read-write activity.

To avoid the damage of data consistency, Hard Disk Manager reboots the system to a single-tasking environment in order to eliminate the interference of other programs.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

As the matter of fact, the formatting of a partition being in use looks illogistic (say, you're keep using a partition and want to destroy it at the same time). It's recommended to re-check your actions to avoid the destruction of usable data.

4.14 Delete Partition

This chapter explains how to delete existing partitions with using Hard Disk Manager.

4.14.1 Overview

Hard Disk Manager allows deleting partitions on hard disks that are partitioned with using the *DOS partitioning scheme* (see Glossary). The program removes references to the partition from the *Partition Table*, so that information from the deleted partition becomes inaccessible. The freed disk space can be used for creation of more partition(s) or added to an existing partition.

Information from the deleted partition can be retrieved back, in case of the block of disk space remains unpartitioned. Hard Disk Manager provides the convenient and powerful function Undelete Partition that allows to find and restore back deleted partitions within blocks of unpartitioned space.

Data from the deleted partition do not disappear from the disk but just become unavailable in the operating system. The specially designed software allows retrieving an entire partition or separate files from it. In particular, confidential information can be analyzed and stolen from deleted partitions.

Hard Disk Manager provides an option to irrevocably wipe out all contents of the partition being deleted. The Wipe Partition function allows to securely remove data from hard disks in cases of hardware clearance sale, disks sanitizing etc.

4.14.1.1 Restrictions

Current version of Hard Disk Manager works with *DOS partitioning scheme* only.

4.14.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a location of a new partition

Select a single existing partition of any kind (*Primary*, *Logical* or *Extended Partition*) in the Tree Layout panel, or in the UDP Layout panel, or in the List of Partitions. The selected partition will be deleted.

In case of the Extended Partition is selected, all *Logical* partitions within the Extended Partition will be removed, too.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Delete…

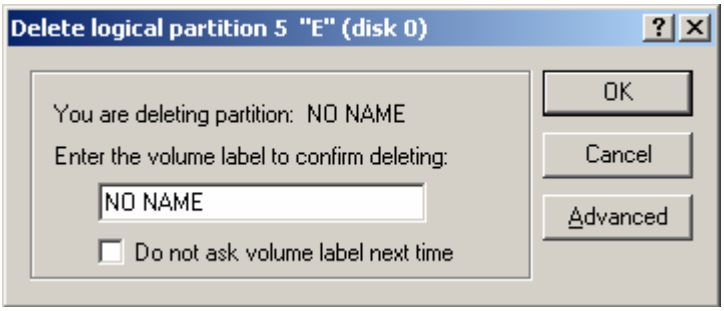
- Call the *popup menu* for the selected partition, then select the menu item:

  Delete…

- Press **Alt+D** keyboard combination
- Press **Delete** button on the Main Toolbar.

Step 3. Confirm the partition deletion

After selection a partition, the *Delete Partition* dialog appears:



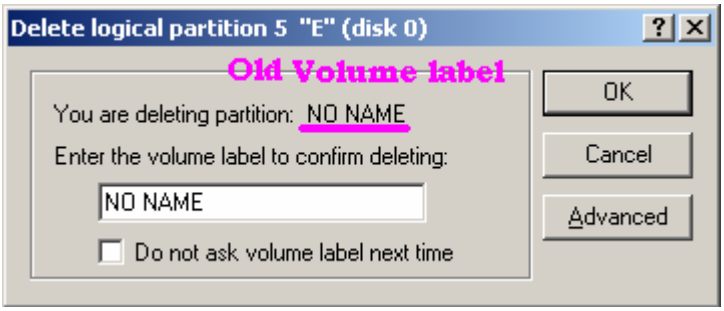
With the *Delete Partition* dialog, Hard Disk Manager asks the confirmation of the partition removal. In addition, the program can wipe the partition during the deletion.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Delete Partition* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.14.3 Description of the parameters



You are deleting partition:

This text contains the current value of the *Volume Label* of the selected partition. In the case the partition has no the *Volume Label* assigned, the "**NO NAME**" text is displayed.

The Volume Label is used for the purpose of additional deletion confirmation from the user.

Enter the volume label to confirm deleting:

To confirm the deletion of the selected partition, enter the Volume Label for it. The actual Volume Label value is displayed just above.

In the case, the partition has no the *Volume Label* assigned, enter the text "**NO NAME**".

Generally, this confirmation is used only to get a user a chance to think over his actions.

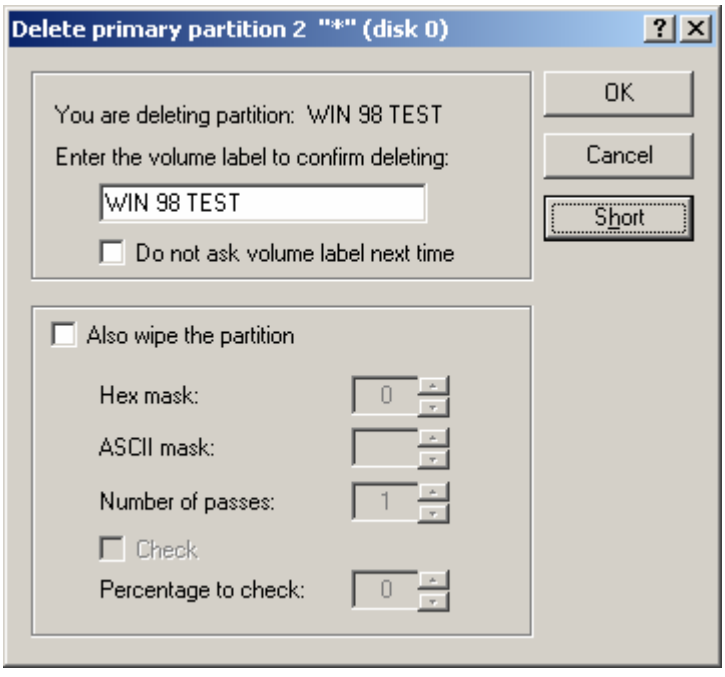
Do not ask volume label next time

Mark this checkmark to inhibit the confirmation of partition deletion. This option duplicates the similar option in the program's settings (see the chapter Settings overview, section General Page).

In fact, Hard Disk Manager now reduces to zero the danger of occasional deletion of a partition. The virtual deletion can be undone; in case of physical removal of a partition, the Undelete Partition function allows to retrieve the partition back.

Advanced (button)

Press this button to get access to advanced properties of the program. The Advanced form of *Delete Partition* dialog appears.



The program provides the ability to wipe the deleted partition.

In case of wiping a partition, all on-partition information becomes irrevocably destroyed: filesystem metadata, programs, personal data and settings.

Also wipe the partition

Set this checkmark so that the program will wipe the contents of the deleted partition (see also the chapter Wipe Partition).

Note, that contents wiping takes some time for the completion. The actual value of the elapsed time fundamentally depends on the hardware performance, operating system used and amount of wiping passes.

Hex mask

Hard Disk Manager wipes the disk space by filling with the single 8-bit character.

The **Hex mask** spinner control allows defining the two-figure hexadecimal presentation of the character being used. The default value is "00" (that is corresponds to the hexadecimal value 0x00). The available range is from "00" to "FF".

This value is synchronized with the **ASCII mask** value.

ASCII mask

Hard Disk Manager wipes the disk space by filling with the single 8-bit character.

The **ASCII mask** spinner control allows defining the symbolic presentation of the character being used. The default value is zero character (character #0).

This value is synchronized with the **Hex mask** value.

Number of passes

Set the amount of wiping passes here. The default value is 1, the available range is 1 to 100.

Check

Set this checkmark ON to force the program to additionally check the just wiped disk space for "clearance" – this feature just allows to ensure the result of wiping.

Percentage to check

The program allows checking for "clearance" a part of the disk space being wiped: set the percentage of the disk space to be checked. The available range is from 0% to 100%, the default value is 0%.

Percentage	Action
100%	All sectors are checked
0%	Nothing is checked
1-99%	The program checks randomly selected sectors over the entire range of disk space.

Short (button)

Press this button to return to the Simple form of the *Delete Partition* dialog. Remember that the supplementary function of wiping of a partition contents will be ignored.

4.14.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time

The program treats the updating of system information, wiping and clearance checking actions as suboperations, and the program displays suboperation statistics.

The deletion of a partition takes just a fraction of a second. Still, the program waits until Windows accommodates changing of the disk layout. This operation may take 5-20 seconds in Windows 2000 and XP.

Instead, the *Wiping* and *Checking* suboperations can take large amount of time. The real value of the elapsed time fundamentally depends of the hardware performance and the software platform being used. On modern disk models, the performance may take 0.5 - 2 minutes per 1Gb of capacity.

4.14.5 Comments

A deletion of partition(s) may lead to mixing drive letters assigned to partitions, in DOS and in Windows 95, 98, ME.

In Windows NT, 2000 and XP, a deletion of a primary partition, which was located in front of the Windows system partition, may lead to inability Windows to boot. The problem can be solved by editing the BOOT.INI system file.

4.14.6 Deleting locked and system partitions

Hard Disk Manager allows to delete locked and system partitions. This operation requires rebooting the computer to be completed. The partition deletion breaks the accessibility of the partition, so that it violates any unfinished read-write activity. In turn, this may lead to the severe problems in the functionality of software being worked.

To avoid mentioned problems, Hard Disk Manager reboots the system to a single-tasking environment in order to eliminate the interference of other programs.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

As the matter of fact, the deletion of a partition being in use looks illogistic (say, you're keep using a partition and want to destroy it at the same time). It's recommended to re-check your actions to avoid the destruction of usable data.

4.15 Mount Partition

This chapter describes how to assign drive letters (mount) and dismount partitions in Windows NT, 2000 and XP, with using Hard Disk Manager.

4.15.1 Overview

Hard Disk Manager allows assigning or changing drive letters to existing formatted partitions. This function is available only in Windows NT, 2000 and XP.

This functionality is not available in DOS and Windows 95, 98 and ME because of these operating systems do not support customizing of drive letters assignment.

Operating systems require some name (e.g. *drive letter*) should be assigned to a partition, which should be used. DOS and Windows 95, 98, ME automatically assign drive letters to all partitions of supported filesystem types during the startup. These operating systems do not allow changing drive letters.

Instead, Windows NT, 2000 and XP freely changing, deleting or adding drive letters to partitions that are supported by these operating systems. Windows provides the Windows Disk Administrator utility for assigning and changing drive letters. Hard Disk Manager can be used instead, for this purpose.

4.15.2 Initiating the operation in the Windows-based version

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select an existing partition

Select any *Primary* or *Logical* partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions.

The operation is not available for the Extended Partition.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

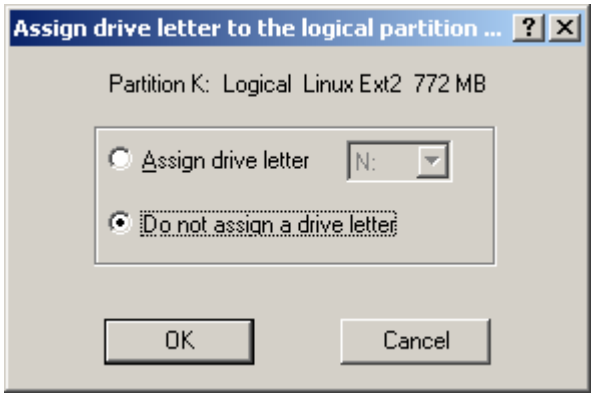
  Partition ➔ Mount…

- Call the *popup menu* for the selected partition, then select the menu item:

  Mount…

Step 3. Assign or change drive letter

After selecting a partition, the *Mount Partition* dialog appears:



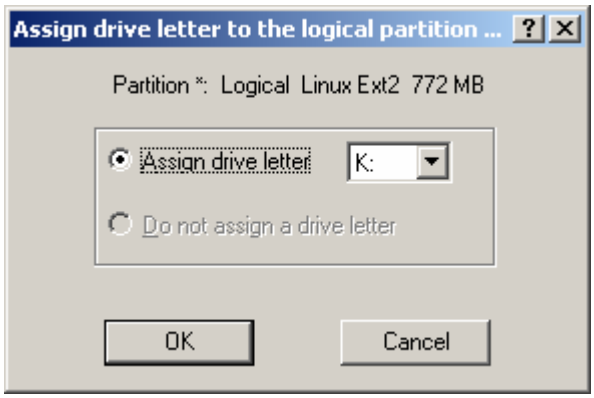
The *Mount Partition* dialog allows to assign, change or delete a drive letter that is associated with partition.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution modes for the *Mount Partition* operation (see the chapter Virtual operations for more details):

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).
- In the *smart mode*, the program:
  - ⇨ immediately executes the operation, in case of no virtual operations are scheduled in the *List of Pending Operations*.
  - ⇨ schedules the operation, if any virtual operations are scheduled in the *List*.

4.15.3 Description of the parameters



Partition

This text contains a brief description of the selected partition. It is used for notification purposes only.

Assign drive letter

Set this radio button to assign a drive letter to a non-mounted partition, or to change a drive letter for already mounted partition.

The pull-down list contains unused drive letters that can be associated with the selected partition.

Do not assign a drive letter

Set this radio button to remove the drive letter associated with a partition from the system. By default, the program suggests this action for mounted partitions. The action of drive letter removal is also named *dismounting*.

4.15.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time

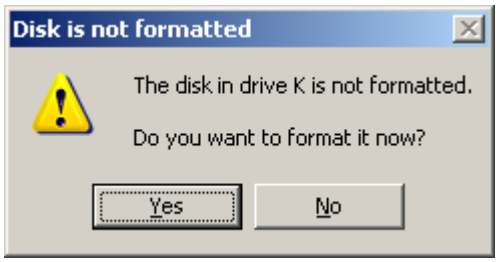
The program treats the updating of system information as a suboperation, and the program displays suboperation statistics.

The mounting/dismounting a partition itself takes several seconds: the program waits until Windows accommodates changing of the disk layout. This operation may take 5-20 seconds in Windows 2000 and XP.

4.15.5 Comments

4.15.5.1 Mounting partitions with unsupported filesystem types

Hard Disk Manager allows mounting partitions of any filesystem type, including ones not supported by an operating system. In this case, a partition has a drive letter assigned; but when attempting exploring partition's contents, a user receives the error message:



4.15.5.2 Manipulating drive letters in Windows 95, 98 and ME

DOS and Windows 95, 98 and ME automatically assign drive letters to partitions during the system startup. These operating systems scan partitions in the predefined order, which cannot be changed, and successively assign drive letters to unhidden FAT16 and FAT32 partitions. In case of IFS drivers for other filesystems are installed, Windows can additionally mount partitions of other types.

The order Windows looks through partitions:

1. The active partition on the first hard disk takes the drive letter C:
2. Windows scans first primary partitions on all other hard disks.
3. Windows scans all logical partitions on the first hard disk, then it adds logical partitions on the second hard disk, and so on.
4. Finally, it adds residue primary partitions on all hard disks.

The legal way of "dismounting" partitions in DOS and Windows 95, 98 and ME is to *Hide* unnecessary partitions. Disadvantages of this method are following:

- Other partitions may change their drive letters.
- A system reboot is required to apply changes.

4.15.6 Dismounting locked partitions

Hard Disk Manager allows dismounting locked partitions. Locked partitions are used by other software; An untimely dismounting of locked partitions may lead to unpredictable results.

To avoid mentioned problems, Hard Disk Manager reboots the system to a single-tasking environment in order to eliminate the interference of other programs. In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.

As the matter of fact, the dismounting of a partition being in use looks illogistic (say, you're keep using a partition and want to cancel its accessibility at the same time). It's recommended to re-check your actions to avoid the corruption of usable data.

4.16 Changing Partition Attributes

This chapter explains how to change some partition attributes ("Hidden" flag, "Active" flag, Partition ID, Volume Label) with using Hard Disk Manager.

4.16.1 Set a Partition Active/Inactive

4.16.1.1 Overview

Hard Disk Manager allows selecting an active partition on a hard disk. *Active* (or *bootable*) partition is the one a system will boot from, at the PC startup (see Glossary for more details).

The *Set active*/*Set inactive* operation is available only for Primary Partitions.

The "Active" flag is kept in entries of the Partition Table. The standard MBR bootstrap code use this flag to define which of primary partitions should be used to boot the system.

With changing an active partition, one can manage an operating system to be activated at next reboot.

4.16.1.2 Initiating the operation

The operation can be activated from the main program's menu or from the partition's popup menu. For inactive partitions, only **Set active** operation is available, For active partitions, only **Set inactive** operation is available.

Step 1. Select a Primary Partition to be activated/deactivated

Select an existing primary partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions.

The *Set Active/Set Inactive* operation is enabled only for Primary Partitions.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Set active…

- Call the *popup menu* for the selected partition, then select the menu item:

  Set active…

4.16.1.3 Running the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution modes for the *Set Active/Inactive* operation (see the chapter Virtual operations for more details):

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).
- In the *smart mode*, the program:
  - ⇨ immediately executes the operation, in case of no virtual operations are scheduled in the *List of Pending Operations*.
  - ⇨ schedules the operation, if any virtual operations are scheduled in the *List*.

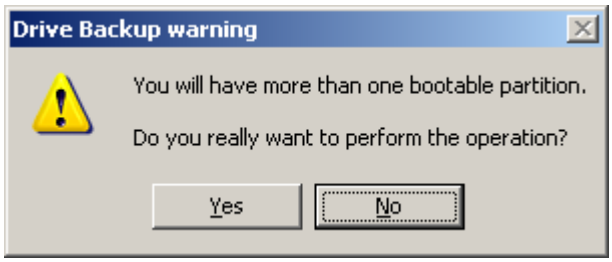
4.16.1.4 Comments

About setting active/inactive locked and system partitions

The *Set Active/Inactive* operation does not interfere with file input-output operations. No reboot is required to change the "Active" flag for locked partitions.

Multiple active partitions

Potentially, Hard Disk Manager allows setting all primary partitions inactive, or setting multiple primary partitions active. In the last case, the program displays the warning message:



The thing is both situations may lead to problems at next system reboot:

- In case of there are no active partitions on a hard disk, the standard bootstrap is unable to continue the startup process. The error message will be displayed:

  ```
  DISK BOOT FAILURE, INSERT SYSTEM DISK AND PRESS ENTER
  ```

- In case of there are multiple active partitions on a hard disk, the standard bootstrap is unable to continue the startup process. The error message will be displayed:

  ```
  Invalid partition table
  ```

8Gb boundary limitation

Another problem comes from the standard bootstrap code limitation: the standard MBR code is unable continuing the startup process from primary partitions that begin beyond the 1023rd Cylinder. On most modern disk subsystems, the 1023rd Cylinder corresponds to approximately 8Gb of disk space. This limitation is named sometimes the "8Gb boundary limitation".

In case of the active partition starts beyond the 8Gb from the beginning of the disk, the standard MBR code displays the error message:

```
Invalid partition table
```

4.16.2 Hide/Unhide Partition

4.16.2.1 Overview

Hard Disk Manager allows *hiding* and *un-hiding* partitions. Operating systems do not mount "hidden" partitions, preventing getting access to their contents (see Glossary for more details).

This function is available for Primary and Logical Partitions only.

The function can be usable in following cases:

- Managing partitions availability in DOS, Windows 95, 98, ME and Windows NT, XP.
- Managing drive letters assigning in DOS and Windows 95, 98 and ME.
- Manual un-hiding of just copied and restored partitions.

4.16.2.2 Initiating the operation

The operation can be activated from the main program's menu or from the partition's popup menu. For hidden partitions, only **Unhide** operation is available, For un-hidden partitions, only **Hide** operation is available.

Step 1. Select a partition to hide/unhide

Select an existing partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Hide…

- Call the *popup menu* for the selected partition, then select the menu item:

  Hide…

4.16.2.3 Running the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution modes for the *Hide/Unhide* operation (see the chapter Virtual operations for more details):

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).
- In the *smart mode*, the program:
  - ⇨ immediately executes the operation, in case of no virtual operations are scheduled in the *List of Pending Operations*.
  - ⇨ schedules the operation, if any virtual operations are scheduled in the *List*.

4.16.2.4 Comments

About hiding locked partitions

The *Hide/Unhide* operation does not interfere with file input-output operations. No reboot is required to change the "Hidden" flag for locked partitions.

Hiding bootable partitions

Potentially, Hard Disk Manager allows hiding bootable partitions. Unfortunately, most operating systems cannot boot from hidden partitions.

Hiding partitions in Windows 2000

Unfortunately, Windows 2000 and XP ignore "Hidden" flag. In these operating systems, it is possible mounting and accessing hidden NTFS, FAT32 and FAT16 partitions. Windows 2000 and XP allow mounting partitions of any type through the intermediation of the internal programming interface (so-called WinAPI), within the current Windows session.

However, there is the method of preventing a partition from the automatic drive letter assigning at Windows startup: the *Partition ID* should be changed to the of 0x12 or 0xDE value (see the chapter Change Partition ID). A propo, this method is recommended by Microsoft in the document "Windows XP OEM Preinstallation Kit, Design Notes".

About hiding the entire Extended Partition

This operation is disabled in Hard Disk Manager.

The thing is that Windows 2000 and XP cannot process hidden Extended Partitions correctly. The system may hang at the stage of accommodating disk layout changes, or it may generate errors at the system startup.

4.16.3 Change Partition ID

The chapter explains how to change Partition IDs with using Hard Disk Manager.

4.16.3.1 Overview

Partition ID is an identifier of a filesystem that is placed in the partition. Partition ID is saved in the Partition Table; it is purposed for fast detecting partitions of supported types.

With manual changing of Partition ID value, it is possible to manipulate the accessibility of partitions.

4.16.3.2 Initiating the operation

Step 1. Select a partition

Select an existing partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

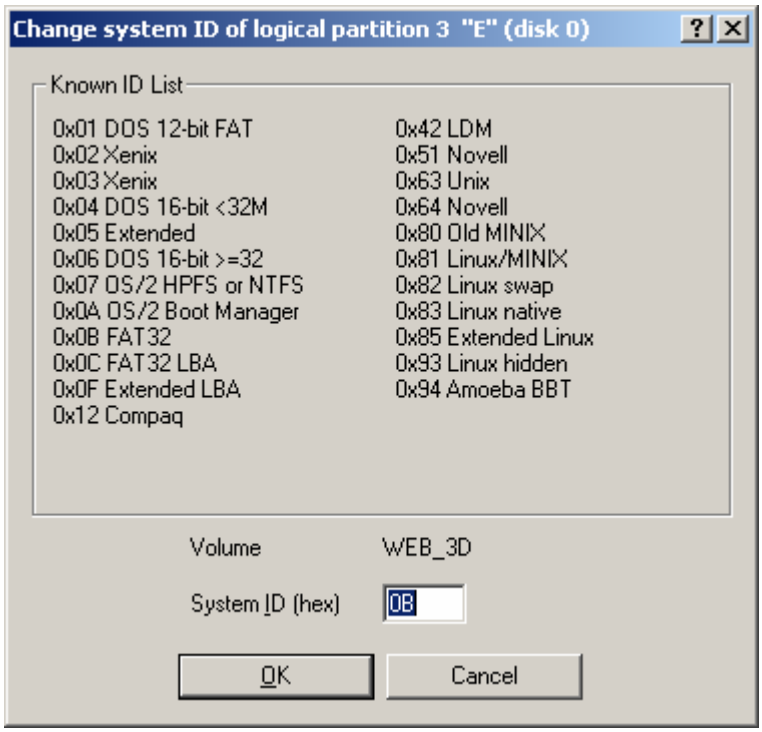
  Partition ➔ Modify ➔ Change partition ID…

- Call the *popup menu* for the selected partition, then select the menu item:

  Modify ➔ Change partition ID…

Step 3. Assign new Partition ID value

After selecting a partition, the *Change partition ID* dialog appears:



With this dialog, you are able to observe and modify the Partition ID value for the selected partition.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Change Partition ID* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

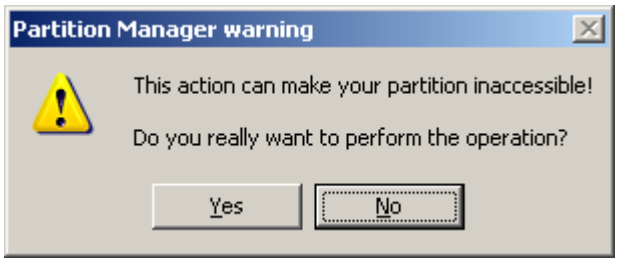
- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.16.3.3 Description of parameters

System ID (hex)

The textual field contains a hexadecimal presentation of the Partition ID. Generally, the Partition ID should be presented as 1-2 digits hexadecimal number; only hexadecimal digits {0..9, A..F} are allowed in this value.

In case of entering of an ID value that is not supported by a currently used operating system, the following warning message is displayed:



Volume

This parameter just displays the Volume Label of the selected partition. This information is only for notification purpose.

Known ID List

The section just provides information about some frequently used Partition IDs. This information is only for notification purpose.

4.16.3.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time

The program treats the updating of system information as a suboperation, and the program displays suboperation statistics.

The opeartion itself takes several seconds: the program waits until Windows accommodates changing of the disk layout. This operation may take 5-20 seconds in Windows 2000 and XP.

4.16.3.5 Comments

How operating systems use Partition ID

DOS and Windows 95, 98, ME, NT and XP rely on the Partition ID value when searching for partitions with supported filesystem types. These operating systems do not allow using partitions with unknown IDs.

Linux actually ignores the Partition ID value. Windows 2000 ignores this value (with only exception of some special values).

The Partition ID value can be used for the following purposes:

- Hiding partitions in Windows 2000, by changing Partition ID to 0x12 or 0xDE values.
- In simple cases, converting Dynamic Disks back to the Basic ones. This option is available only for hard disks that were initially partitioned as the Basic Disks, and then converted from Basic to Dynamic ones.

Changing Partition ID for locked partitions

This action does not require a reboot, because the Changing Partition ID operation does not violates file input-output activity.

4.16.4 Set Label of a Partition

The chapter explains how to change the *Partition Label* parameter with using Hard Disk Manager.

4.16.4.1 Overview

The *Partition Label* is a small textual field (up to 11 characters) that is located in the partition's boot sector. This value is detectable by any partitioning tool; it is used for notification purposes only.

4.16.4.2 Initiating the operation

Step 1. Select a partition

Select an existing partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Modify ➔ Set Label…

- Call the *popup menu* for the selected partition, then select the menu item:

  Modify ➔ Set Label…

Step 3. Assign the new Partition Label

After selecting a partition, the *Change partition label* dialog appears:

With this dialog, you are able to observe and modify the Partition Label value for the selected partition.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution modes for the *Set Label* operation (see the chapter Virtual operations for more details):

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).
- In the *smart mode*, the program:
  - ⇨ immediately executes the operation, in case of no virtual operations are scheduled in the *List of Pending Operations*.
  - ⇨ schedules the operation, if any virtual operations are scheduled in the *List*.

4.16.4.3 Description of parameters

Current label

This parameter just displays the current Partition Label value of the selected partition. This information is only for notification purpose.

New label

Enter the new value of the Partition Label. The length of the Label is limited to 11 characters.

4.16.4.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time

The program treats the updating of system information as a suboperation, and the program displays suboperation statistics.

The opeartion itself takes several seconds: the program waits until Windows accommodates changing of the disk layout. This operation may take 5-20 seconds in Windows 2000 and XP.

4.16.4.5 Changing Label of locked and system partitions

Hard Disk Manager can change Partition Label of locked and system partitions. This operation requires rebooting the computer to be completed.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

4.17 Undelete Partition

This chapter describes how to return back occasionally deleted partitions with using Hard Disk Manager.

4.17.1 Overview

Hard Disk Manager provides an ability of finding and recovering deleted partitions. This functionality is usually named "*undelete*". The undelete feature nullifies the danger of an occasional partitions deletion.

To delete a partition, disk management software just removes references to a partition in the Partition Table (see the chapter Delete Partition). So that a previously deleted partition can be recovered, in case of valid restoration of the record in the Partition Table. A restored partition will be usable, in case of other partitions were not created, moved or expanded over disk space that is occupied by this partition. For this reason, the program enables the *Undelete Partition* function only for blocks of free space.

To find and restore a partition, Hard Disk Manager scans sectors in order to detect servicing structures of a filesystem that was installed on a deleted partition. Of course, the program allows finding and correctly restoring only Primary and Logical partitions of known filesystem types (see the section Limitations).

Hard Disk Manager supports the *manual* and the *semi-automatic* un-deletion modes:

- In the manual mode, a user is allowed freely defining location and size of a partition that should be undeleted. Potentially, this option provides an ability of restoring partitions of any type.
- In the semi-automatic mode, the program searches traces of deleted partitions within a block of free space. The program makes a list of all places, which look like a deleted partition. A user can inspect listed variants and select one to undelete.

Hard Disk Manager identifies deleted partitions by some starting fragments of a filesystem. There can be situations when the program finds "phantom" partitions, i.e. places that only look like a deleted partition but they are not ones in reality. In such situations, a user can undelete a partition and then check a filesystem or browse its contents (again, with using Hard Disk Manager) in order to understand is it a real partition or not.

Phantom partitions usually do not pass the filesystem integrity check; when browsing, their "contents" seem strange or corrupted. A user can simply delete a phantom partition and make a new attempt of undeletion.

4.17.1.1 Limitations of Undelete function

The *Undelete Partition* function does not support virtual execution. Either mode Hard Disk Manager works, the Undelete is always executed in the *Immediate Execution* mode.

The special feature of the *Undelete Partition* function is that the program *disables the function* in case of any virtual operations are placed in the *List of Pending Operations*.

To check if any virtual operations are in the *List of Pending Operations*, inspect the state of the Virtual Operations Toolbar or the state of the **Operations** submenu – all contents of the toolbar and the submenu are disabled (with the only exception of the "**Reload** " item), in case of no virtual operations is accumulated in the List.

Other limitations:

- The current version of Hard Disk Manager allows restoring one partition per *Undelete Partition* operation. To undelete several deleted partitions, multiple undelete sessions are required.
- The current version of Hard Disk Manager supports *DOS partitioning scheme* only. In Windows 2000 and XP, these disks are named Basic Disks. Deleted partitions on Dynamic Disks cannot be restored!

- The program allows restoration of partitions of following types:
  - ⇨ FAT16 & FAT32
  - ⇨ NTFS
  - ⇨ Ext2 & Ext3
  - ⇨ ReiserFS
- The program cannot un-deleting partitions, which have corrupted boot sector.
- Hard Disk Manager is unable retrieving partitions that were *wiped-&-deleted* with using Hard Disk Manager, Disk Wiper or similar software. Such software is specially designed for complete and irreversible destruction of data.

4.17.2 Initiating the operation

Step 1. Select a block of free space

The Undelete Partition operation is enabled only for blocks of free space.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Undelete…

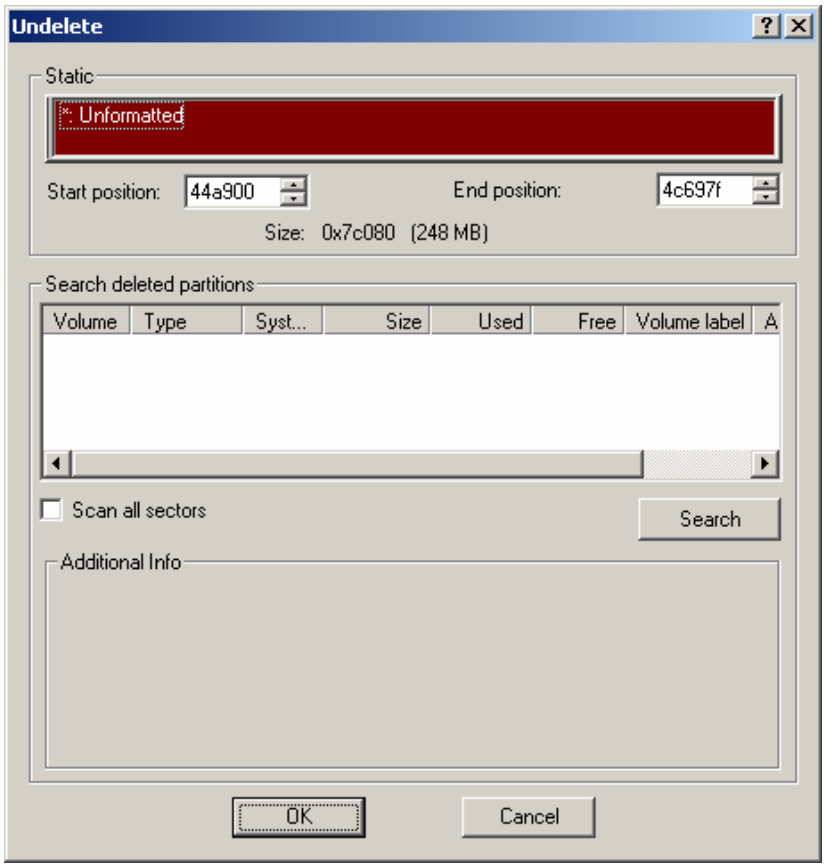
- Call the *popup menu* for the selected block of free space, then select the menu item:

  Undelete…

In case of this operation is disabled, check that there no pending operation in the *List of Pending Operations*; you can either clear the list (**Undo all** operation) or execute pending operations (**Apply** operation).

Step 3. Work with the Undelete Partition dialog

The *Undelete Partition* dialog should appear:



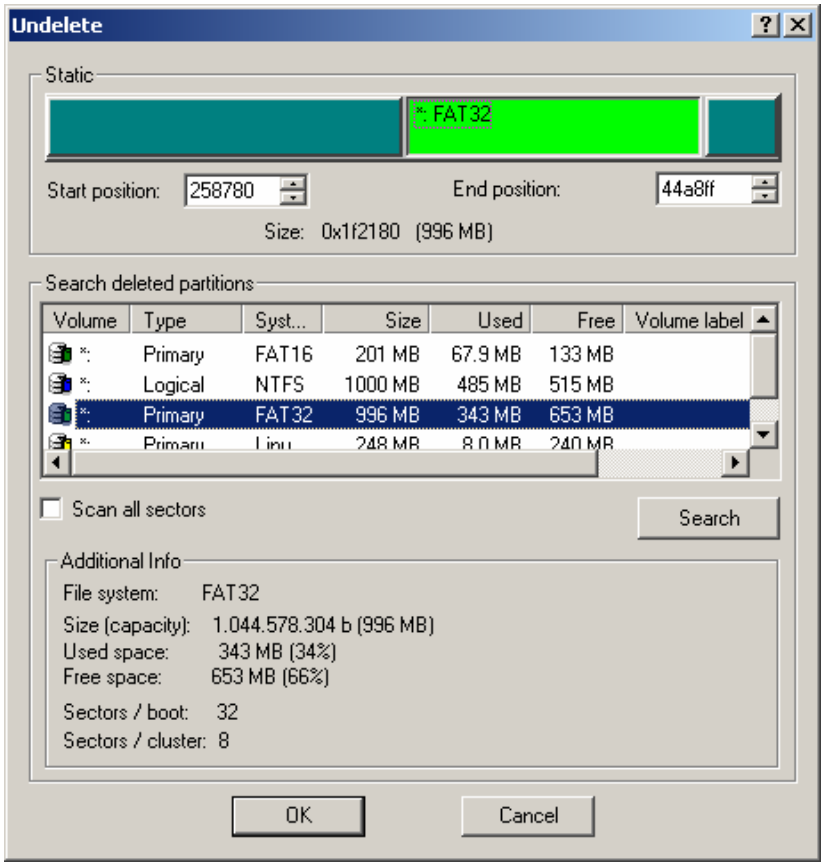
The same dialog is used for both manual and semi-automatic un-deletion.

The section Description of the dialog explains purposes and functionality of dialog elements.

See the section Semi-automatic Undeletion of Partitions to learn about the procedure and abilities of semi-automatic undeletion.

See the section Manual Undeletion of Partitions to learn about the procedure and abilities of manual partition undeletion.

4.17.3 Description of the dialog



Static (group of elements)

The top group of elements named "Static" is purposed for the "final" assignment of parameters of a partition, which should be undeleted after a user presses **OK** button bottom the dialog window.

(UDP control)

The *UDP control*, which is located on the top of this group of elements, displays a resulting position, a size and an estimated filesystem type of a partition being undeleted. This UDP provides an ability of changing resulting location and size of an undeleted partition, with using the drag-&-drop technique (see the section UDP control activity for more details).

Start position

This spinner control displays (and allows changing) the starting position of a partition, in sectors from the beginning of the disk, in the hexadecimal presentation.

Note about specific behavior of this spinner control:

- With changing the displayed value by clicking on *up* and *down* arrows, the spinner increases or decreases the position value by the [amount of sectors in the Cylinder for the current disk geometry].
- With direct entering a new value in the spinner control, one can define any acceptable value of the starting position.

See the detailed explanation this specific feature in the section Manual Undeletion of Partitions.

End position

This spinner control displays (and allows changing) the ending position of a partition, in sectors from the beginning of the disk, in the hexadecimal presentation.

This spinner control provides the specific behavior, same with the behavior of the **Start Position** spinner control.

Size

This text just displays the partition size in sectors and megabytes.

Search deleted partitions (group of elements)

This group of elements provides a functionality that is used in the semi-automatic undeletion.

(List of found partitions)

This control contains the list of currently found deleted partitions. The program retrieves and correctly displays the following information about found partitions:

- Type (Primary or Logical)
- Filesystem type
- Size, Used Space and Free Space
- Volume Label

Scan all sectors

Set this checkmark to force the program an exhaustive searching of deleted partitions. This feature allows finding deleted partitions that do not meet rules of partition alignment , which are effective in the DOS partitioning scheme.

This option affects significantly slows the search performance. In addition, it can lead to detecting more phantom partitions.

Search

Press this button to search deleted partitions within the selected block of free space.

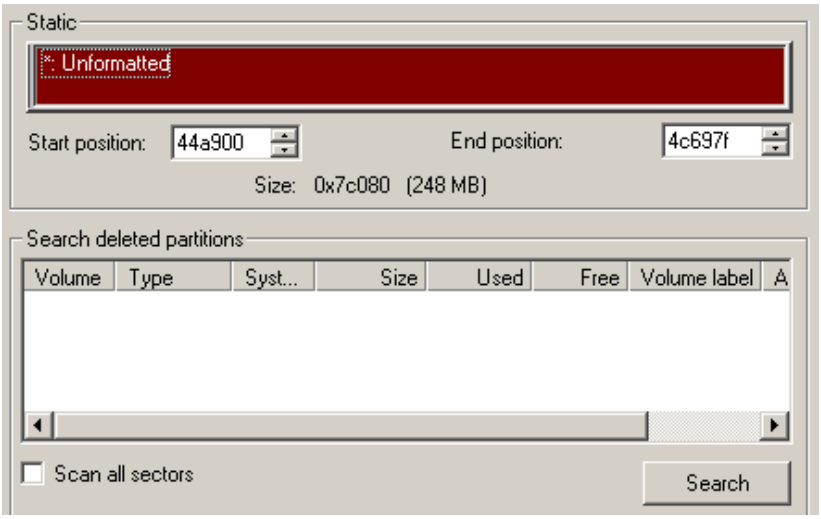
When a user presses the **Search** button, the program clears the list of found partitions. Then, it populates the list again, with found partitions.

Additional Info

This control displays some details about a partition, which is currently selected in the List of found partitions. This information can help in distinguishing desired partition.

4.17.4 Semi-automatic Undeletion of Partitions

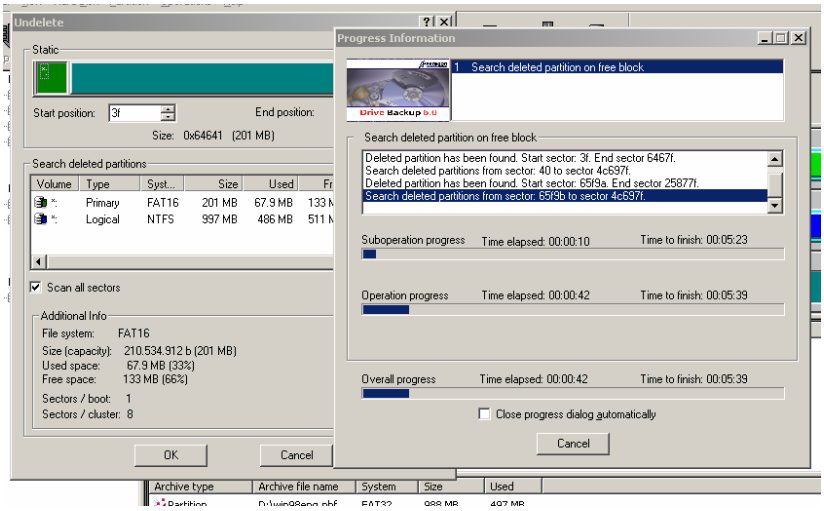
Initially, the *Undelete Partition* dialog suggests to "restore" a partition, which occupies a whole block of free space.



Initially the program displays its type as "Unformatted" because the program haven't analyzed disk yet.

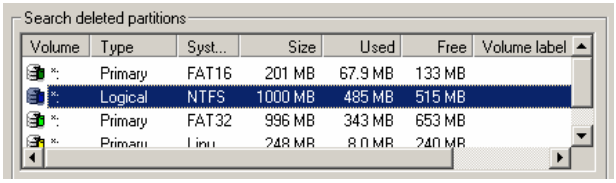
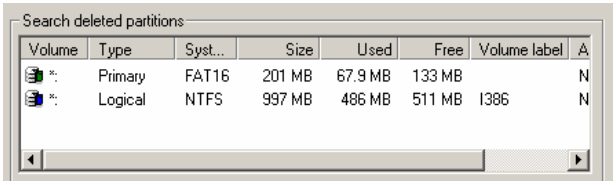
Step 1. Searching deleted partitions

Press the **Search** button; the program begins scanning the disk and filling the list of found partitions:



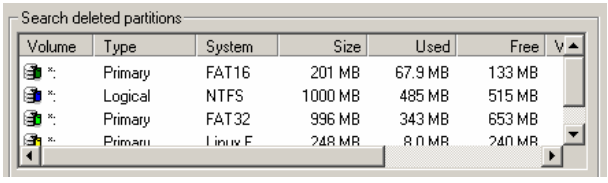
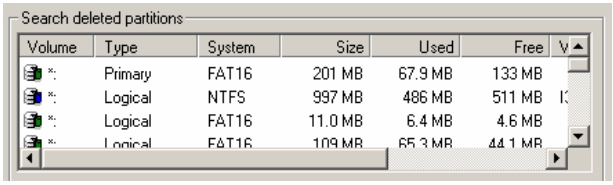
The program will open the Progress dialog for the Undelete operation, which reports the overall progress of the searching process. Simultaneously, the program populates list of found partitions.

A user can break scanning the rest of the disk, if the program have already finds and includes the required partition in the list.

Complete scanning **The program has found 4 partitions**	
Interrupted scanning **The program has found 2 partitions before interruption**	

Step 2. (optional) Scanning all sectors

There can be situations that Hard Disk Manager fails detecting desired partitions. In this case, a user can set the option "**Scan all sectors**" and repeat searching.

Normal Scan **4 partitions found, 1 is phantom one, 1 real partition was not found**	
Scan all sectors **14 variants found, 10 phantom ones, all real partitions are in the list.**	

The difference between these modes is the following:

In the normal mode, Hard Disk Manager expects that deleted partitions were correctly allocated in accordance with partition alignment rules, and, that the *disk geometry* was not changed since these partitions were created. Taking into account these assumptions, Hard Disk Manager reads just a few sectors per each Cylinder.

In the exhaustive searching mode, Hard Disk Manager examines all sectors within the selected range of the disk space. In this case, it can detect not only "normal" partitions but also ones that are not correctly aligned.

Incorrectly aligned partitions may appear in case of creating different partitions with different BIOS settings, or in different operating systems. For example, Linux, Windows and DOS apply different *disk geometry* values (see Glossary and Partition alignment rules of the DOS partitioning scheme); Windows 98 and Windows XP can apply different disk geometry values for very large hard disks (greater than 32Gb capacity).

Step 3. Inspecting found partitions

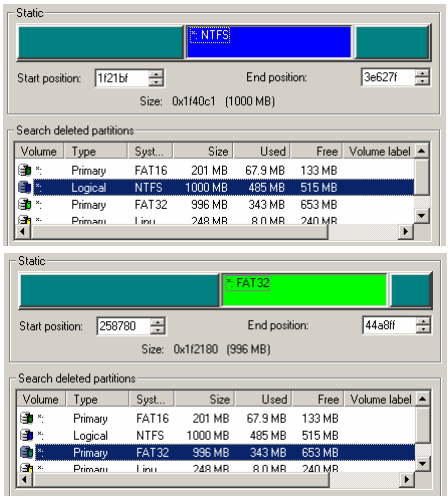
When a user selects an item in the list of found partitions, the UDP control displays its position and size within the selected range of the disk space, and spinner controls display its location.

A user can inspect type, size and location of every found partition.

On the picture: two of found partitions intersect each other. One of these partitions is definitely a phantom one.

In such situations, the program is unable distinguish a valid partition. A user must decide by self which of partitions should be restored.

Information about partition's type, size, amount of data or its relative position within the block of free space may help in making a right choice.



Formally, the program still allows a user changing "**Start position**" and "**End position**" parameters even after selecting an item in the list of found partitions. In most cases, it just leads to an incorrect restoration of a partition.

Step 4. Appling the selection

Finally, a user should press **OK** button bottom the dialog window in order to undelete the selected partition. The program will modify the Partition Table and then ask the system to accommodate changes. This operation takes several seconds.

Note, the program allows retrieving only one partition per *Undelete* operation.

4.17.5 Manual Undeletion of Partitions

The program allows retrieving partitions with manual entering parameters of an undeleted partition.

4.17.5.1 Actions

To manually undelete a partition, a user should just enter desired values to the "**Start position**" and "**End position**" spinner controls, and press **OK** button in order to complete the operation.

Generally, these steps are explained in sections Semi-automatic Undeletion of Partitions and Description of the dialog.

In spite of meager interface support, the *Undelete Partition* function combined with the Change Partition ID function allows recovering partitions of any type, including unknown filesystems, and ones that are incorrectly aligned. The feature can be useful for people who use operating systems and filesystems Hard Disk Manager does not support, who frequently toggles a hard disk between several computers and so on.

Operating systems take into account rules of partitions alignment (see Comments); unfortunately, they do not follow common standards in *disk geometry* parameters, which are used in the mentioned rules.

For example, Linux always use so-called NORMAL disk geometry, while Windows prefer LBA geometry. A partition created in Linux is correctly aligned for the NORMAL disk geometry. In Windows, this partition becomes "incorrectly aligned". In most cases, Windows will see this partition and can work with it. However, after deleting this partition, Windows-based utilities that rely on disk geometry are unable detecting and restoring it. Hard Disk Manager is free of this limitation.

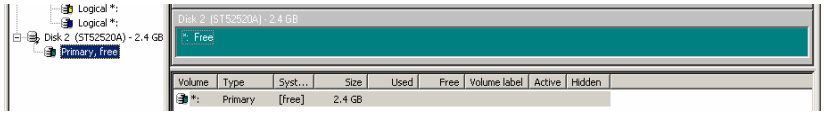
4.17.5.2 Automatic restoration of size and filesystem type

In case of manual entering partition's location, the program does not perform detection of the filesystem type, so that it displays the partition as "Unformatted".

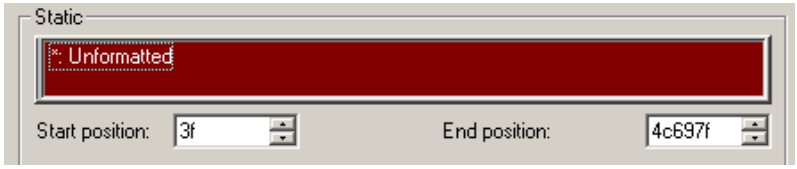
When committing changes in disk layout, in case of a partition being undeleted coincides with existed partition, the program can detect the actual filesystem type and size, and correctly retrieve these parameters.

Pictures below demonstrate the feature of automatic parameters restoration:

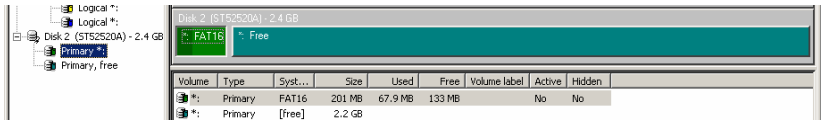
Initial disk layout:



Disk presentation in the Undelete Partition dialog:



The actually undeleted partition has correct size and filesystem type:



4.17.6 Undeleting the Extended Partition

An undeletion of the Extended Partition is the special case.

The Extended Partition is the specific one: it is used as the container for Logical Partitions. If the Extended Partition is deleted, all nested Logical Partitions become deleted, too.

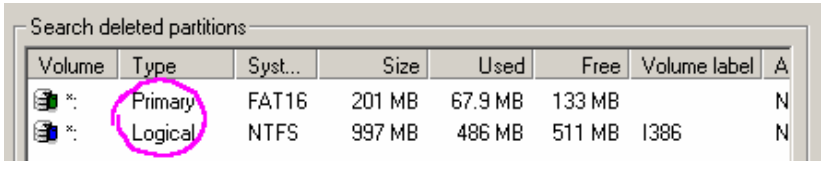
Unfortunately, current version of Hard Disk Manager does not support a direct undeletion of the Extended Partition, primarily because of intricate logic and multiple possible variants that should be examined.

Still, Hard Disk Manager can be used for recovering the entire Extended Partition with all Logical Partition included.

Like ordinary partitions, the Extended Partition can be undeleted in the manual and in the semi-automatic modes. The difference from ordinary partitions is that in both cases some additional actions are required.

4.17.6.1 Semi-automatic undeletion of the Extended Partition

When searching deleted partitions, Hard Disk Manager estimates their partition type (Primary/Logical) by analyzing their actual location and parameters in the boot sector.



Still, at the time of undeletion, the program ignores this information:

- If a deleted partition is located within the Extended Partition, it will be restored unconditionally as a Logical Partition.
- If a deleted partition is located outside the existing Extended Partition (or in case of no Extended Partition exists), it will be restored unconditionally as a Primary Partition.

The disadvantage of this logic becomes apparent in the situation of the Extended Partition was deleted: the program finds all Logical partitions, but it allows restoring them only as Primary ones.

To correctly restore Logical Partitions, use the following procedure:

1. Select the block of free space, which covers the former location of the deleted Extended Partition.

2. Create the new Extended Partition over the entire block of free space (see the chapter Create Partition for details).
3. After this operation, an empty Extended Partition should appear. It should contain a block of free space within. Complete all pending operations (by pressing **Apply** button on the Toolbar) in order to enable the *Undelete Partition* operation.
4. Select the block of free space within the Extended Partition and perform the Undelete Partition operation.

Remember, you can retrieve only one partition per Undelete session. To recover multiple Logical Partitions, multiple calls of the Undelete operation are required.

4.17.6.2 Manual undeletion of the Extended Partition

The procedure of manual undeletion of the Extended Partition may lead to the complete restoration of all Logical Partitions in only two steps. On the other hand, it succeeds only in the case a user guesses right the location of the Extended Partition.

Generally speaking, the manual procedure is effective in case of the existing block of free space has appeared in place of the deleted Extended Partition.

To restore Logical Partitions, use the following procedure:

1. Select the block of free space and call the *Undelete Partition* operation.
2. In the *Undelete Partition* dialog, press the OK button in order to retrieve a partition. In case of you know the exact location of the deleted Extended Partition, enter this value to the "**Start position**" spinner control.
3. After these actions, the program will "recover" an *Unformatted* partition over the selected free block.
4. Select this partition and call the *Change Partition ID* operation.
5. In the *Change Partition ID* dialog, enter the "F" or "5" Partition ID value. These values correspond to the Extended Partition
   - ⇨ 0xF – the Extended Partition requires EBIOS support; it exceeds 8Gb in size, or it begins over 8Gb boundary from the beginning of disk space.
   - ⇨ 0x5 – the Extended Partition does not require EBIOS, it entirely fits in first 8Gb of disk space.

In case of the starting position of the deleted Extended Partition was guessed right, all Logical Partitions would be restored, too.

A user can retrieve information about partitions from the log-file that Hard Disk Manager keeps during working sessions. Early records in the log-file may contain data about deleted partitions.

4.17.7 Comments

4.17.7.1 Alignment of undeleted partitions

When *undeleting* partitions, Hard Disk Manager only updates records in the Partition Table, so that an undeleted partition becomes available. The program does not perform extra actions to adjust the starting position and capacity of a partition to the actual *hard disk geometry*. The partition alignment rules are discussed in the section Create Partition ➔ Comments ➔ Partition alignment rules of the DOS partitioning scheme, the hard disk geometry is discussed in the Glossary. This feature allows undeleting partitions having allocation mismatched to the *partition alignment rules*.

This feature substantially extends the class of retrievable partitions. Such a functional flexibility is of value for users:

- who uses multiple operating systems on the single computer.
- who frequently switches a hard disk over multiple computers.

The thing is that *partition alignment rules* rely on the hard disk geometry. On modern systems, the *hard disk geometry* is not the native feature of a device but only a firmware dependable (or a software programmable) representation of hard disk parameters. The same hard disk can have different geometry on various computers and even in different operating systems on the same computer. So that a partition may appear misaligned in the actual system, if it was created (or modified by Hard Disk Manager) under other conditions.

Hard Disk Manager allows detecting and retrieving deleted partitions even in case of mismatching the actual hard disk geometry with one used at the partition creation. Most of modern operating systems and disk managing tools successfully work with misaligned partitions.

4.17.7.2 Known problems with Undeleting misaligned partitions

When searching deleted partitions, Hard Disk Manager uses information from a boot sector of a found partition in order to distinguish its type (Primary or Logical). Formally, Logical Partitions are located within the Extended Partition, and Primary Partitions are outside. However, in case of a user somehow changes the capacity of the Extended Partition, this rule can be violated.

Primary and Logical partitions meet different alignment requirements (see the section Create Partition ➔ Comments ➔ Partition alignment rules of the DOS partitioning scheme), so that an undeleting of a logical partition outside the Extended Partition, as well as an undeleting of a primary partition inside the Extended Partition makes a retrieved partition misaligned. The following sections recover known problems with misaligned partitions.

4.17.7.3 Undeleting a Logical Partition outside the Extended Partition

This operation is safe, and a Logical Partition can be retrieved outside the Extended Partition without problems. The resulting partition becomes a Primary one. Modern operating systems will recognize and successfully access such partitions. Old versions of MS-DOS can fail recognizing partitions of this kind.

4.17.7.4 Undeleting a Primary Partition inside the Extended Partition

Under some conditions, this operation can lead to damaging of an undeleted partition. Conditions are the following:

- Searching deleted partitions within the Extended Partition,
- A deleted *Primary* partition is found,
- It is located right behind an existing Logical Partition,
- It is aligned to the beginning of an appropriate Cylinder (that's correct for primary partitions, but it is wrong for logical ones).

In this case, the program is unable undelete the partition correctly. After the completing the operation, the undeleted partition becomes "*unformatted*".

To get around the problem, use the following procedure:

1. Move boundaries of the Extended Partition in order to exclude the location of the deleted primary partition from the Extended Partition.
2. Run the Undelete Partition operation again
3. Search deleted partitions. The program will find this partition once again.
4. Restore this partition as a Primary one.

The partition will be restored correctly.

4.18 Supplementary functions

This chapter describes supplementary functionality available in Hard Disk Manager.

4.18.1 Retest surface

4.18.1.1 Overview

Hard Disk Manager allows performing additional surface test on existing partitions and blocks of free space.

This option allows detecting unreliable sectors on a hard disk. Unfortunately, current version of the program does not support retrieving data that become located in bad sectors.

4.18.1.2 Initiating the operation

The operation is available for partitions of any type and for blocks of free space. It can be activated from the main program's menu or from the partition's popup menu.

Step 1. Select a Partition or a Block of Free space

Select a partition or a block of free space in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Retest surface…

- Call the *popup menu* for the selected partition, then select the menu item:

  Retest surface…

4.18.1.3 Running the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Retest surface* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed

The *Retest surface* operation takes a long while; the actual value of elapsed time depends on the size of a partition (or a free block) being tested, hardware performance and the Surface test settings (see the chapter Settings overview for more details).

4.18.1.4 Comments

The primary purpose of this function is detecting of bad and unreliable sectors on existing partitions. In case of detecting bad sectors in a usable partition, exit Hard Disk Manager session and use a standard disk-checking tool from a used operating system (**CHKDSK**, **SCANDISK**, **e2fsck** and so on).

4.18.1.5 Retest surface on locked partitions

The program allows testing surface of locked and system partitions. The operation requires rebooting the computer to be completed.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

4.18.2 Check Filesystem Integrity

4.18.2.1 Overview

Hard Disk Manager allows checking of filesystem integrity on existing FAT16, FAT32 and NTFS partitions. This function can be used for detecting filesystem errors before applying other operations on a partition.

Most useful operations require a targeted partition must have a valid filesystem to be processed; otherwise, the program cancels this operation and all consecutive virtual operations in the *List of Pending Operations*.

The advantage of Hard Disk Manager is that it can check both mounted and non-mounted partitions of a many types. Unfortunately, current version of the program cannot fix filesystem errors, just detects them.

4.18.2.2 Initiating the operation

The operation is available for primary and logical partitions, which are formatted to the following filesystem types: FAT16, FAT32, NTFS. The program allows checking both mounted and non-mounted partitions.

The function can be activated from the main program's menu or from the partition's popup menu.

Step 1. Select a Partition to be checked.

Select a formatted partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

  Partition ➔ Check file system integrity

- Call the *popup menu* for the selected partition, then select the menu item:

  Check file system integrity

4.18.2.3 Running the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Check filesystem integrity* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed

The *Check filesystem integrity* operation takes a long while; the actual value of elapsed time depends primarily on amount of data being stored on the targeted partition.

4.18.2.4 Comments

The primary purpose of this function is a preliminary detection of erroneous partitions, as they cannot be handled with the program. In case of detecting filesystem errors, exit Hard Disk Manager session and use a standard disk-checking tool from an appropriate operating system (**CHKDSK**, **SCANDISK** and so on).

4.18.2.5 Check filesystem integrity on locked partitions

The program allows checking filesystem on locked and system partitions. The operation requires rebooting the computer to be completed.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

4.18.3 Check archive integrity

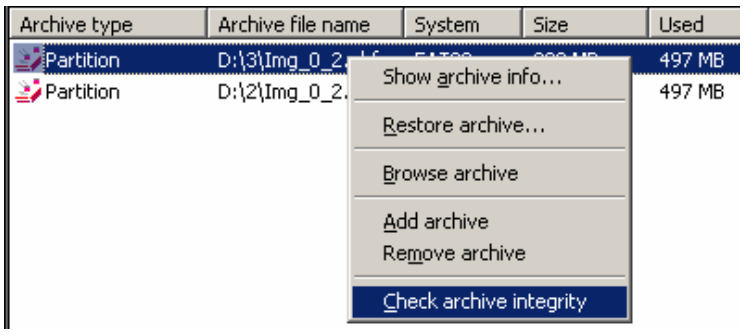
4.18.3.1 Overview

Hard Disk Manager provides an ability performing the archive integrity check for backup images. The function allows distinguishing between valid and corrupted images prior using them.

4.18.3.2 Initiating the operation

The *Check archive integrity* function is available only from application's interface, it does not support virtual execution. Either execution mode the program is switched to, the integrity check is always executed in the *Immediate Execution* mode.

The operation is available only for backup images listed in the *List of Backup Images* via the context menu. The program automatically appends the *List of Backup Images* at cases of creating new backup images and opening existing ones.



To perform the integrity check, select an image from the *List of Backup Images* and call the context menu. Select the item **Check archive integrity**.

4.18.3.3 Running the operation

Hard Disk Manager supports only the *Immediate* execution of the *Check Archive Integrity* operation. The program starts the operation immediately after submitting parameters.

During the operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed

The program treats the verifying of a single volume of a multivolumic archive as a suboperation. In this case, the program displays suboperation statistics.

The operation takes a long while. Real performance depends on:

- The integrity checking algorithm used during the image creation (see the section Settings overview➔Backup➔Do not control archieve integrity).
- File input-output system performance.

4.18.4 Send log-files

4.18.4.1 Overview

Hard Disk Manager simplifies the procedure of sending a support requests to the Paragon Support Team. Generally, support engineers require some technical details about a user's computer, disk layout and performed operations. Most of this information is kept in LOG-files the program keeps.

After activating this function, the program starts a default mail client, generates a template request e-mail with compressed LOG files being attached. A user should only include a generic description of a problem.

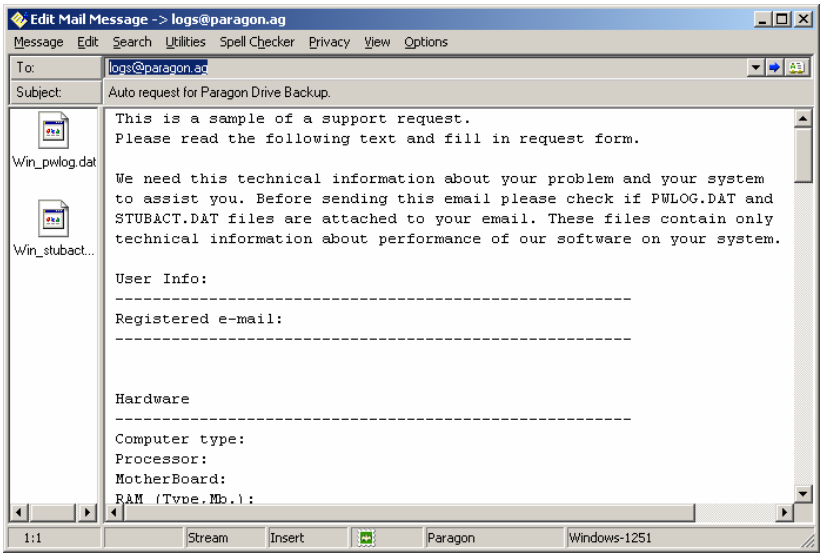
4.18.4.2 Performing the operation

Select from the main menu:

Help ➔Troubleshooting ➔Send log files

The program will make following actions:

1. compress LOG files with using built-in compressing module.
2. start the default mail client registered in the system.
3. The mail client starts with a support query template, having compressed LOG-files attached:



4. Finally, a user should fill in a support request form, describe a problem in details and send email to the Support Team.

4.18.4.3 Comments

LOG files

Hard Disk Manager 6.0 keeps following LOG-files:

STUBACT.LOG	Contains the extended information about parameters and performance of every executed operations and changes being made in disk layout.
PWLOG.TXT	Contains brief information about operations and the extended information about state of all hard disks.
BioNTlog.TXT or Bio95log.TXT	An OS-dependent supplementary LOG-file from BIOxx.DLL. It may contain valuable information on Windows NT/2000/XP-managed systems.

LOG files are in readable textual format; they contain information about Hard Disk Manager performance and just most common information about hard disks layout. These files do not include confidential information about system settings, user documents or other things.

The request template is included in the file **AUTOREQUEST_ENG.TXT**, which is located in the subfolder "**Resource**".

Error sending mail

In Windows, in case of the program fails starting the default mail client, the following system error message appears:

To solve the problem:

- Check settings of a default mail client and default Internet connection.
- Run Internet Connection Wizard and create a new mail account.
- In the Internet Explorer settings, choose a default mail client.

An alternative routine is the following:

1. run mail client with keeping Hard Disk Manager working.
2. repeat sending LOG files (with keeping mail client working)

4.18.5 View Partition/Hard Disk Properties

4.18.5.1 Overview

Hard Disk Manager allows obtaining general information about hard disks and partitions. The program displays not only a standard information such as capacity, used space or filesystem type, but also some advanced information such as *hard disk geometry*, *Cluster Size*, exact partition location, and so on. This information can be used for revealing mistaken situations, finding reasons of errors and other purposes.

4.18.5.2 Running the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select an interesting object (a disk or a partition)

Select a hard disk or a partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. In any case, it will be highlighted in all three panels.

Step 2. Select the operation to perform

Variants:

- call the *popup menu* for the selected object and select the item:

  Properties

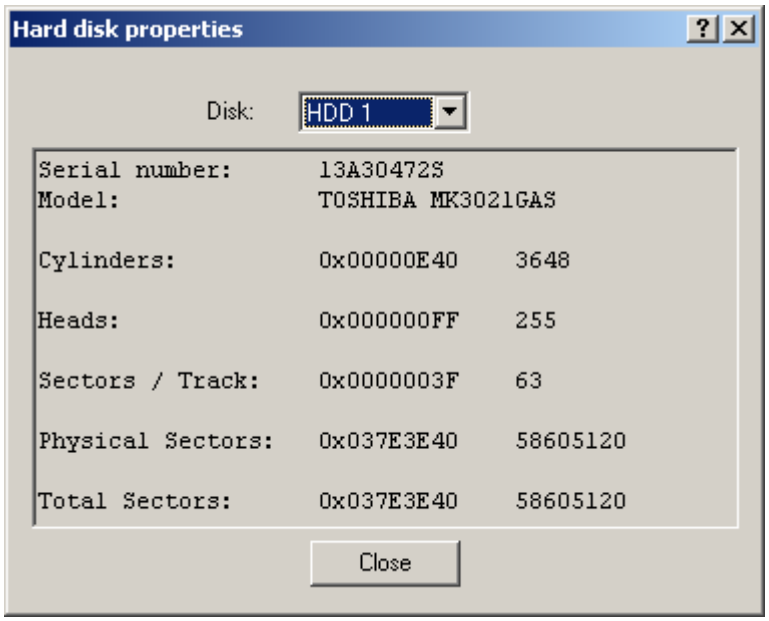
- Press the **Properties** button on the Main Toolbar.
- Select in the main menu:

  Hard disk ➔ Hard disk properties… (for hard disks)

  Partition ➔ Properties (for partitions)

- Press **Alt+Enter** for displaying Partition properties.
- Press C**trl+I** for displaying Hard Disk properties.
- Press **Copy** button on the Main Toolbar.

4.18.5.3 Hard Disk Properties



Disk

This pull-down list allows choosing a hard disk for observing its properties.

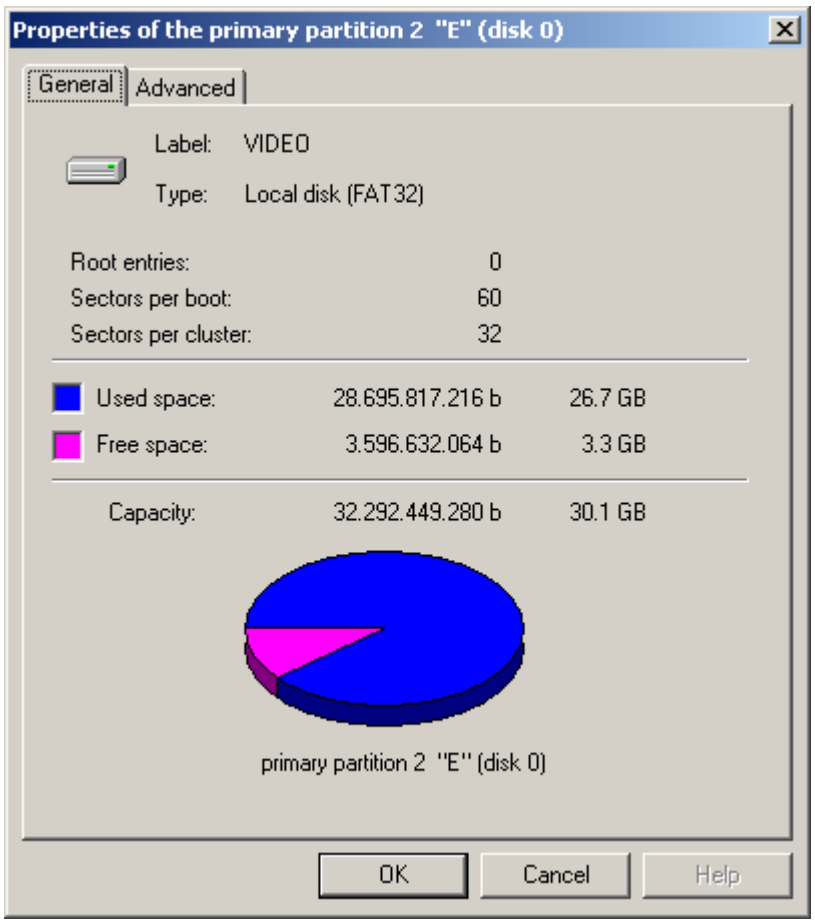
The program displays the following information about a hard disk:

Serial number	This OEM information can be unavailable in DOS, Linux and in some cases in Windows 95/98/ME environments.
Model	
Cylinders	These parameters constitute the actual *Hard Disk Geometry*. The program displays both the decimal and the hexadecimal representation of these values.
Heads	
Sectors / Track	
Physical Sectors	The maximum disk capacity value as reported by the hardware.
Total Sectors	The maximum disk capacity value as reported by the operating system. This value is actually derived from the *Hard Disk Geometry*: <*Total Sectors*> = <*Cylinders*> * <*Heads*> * <*Sectors/Track*>

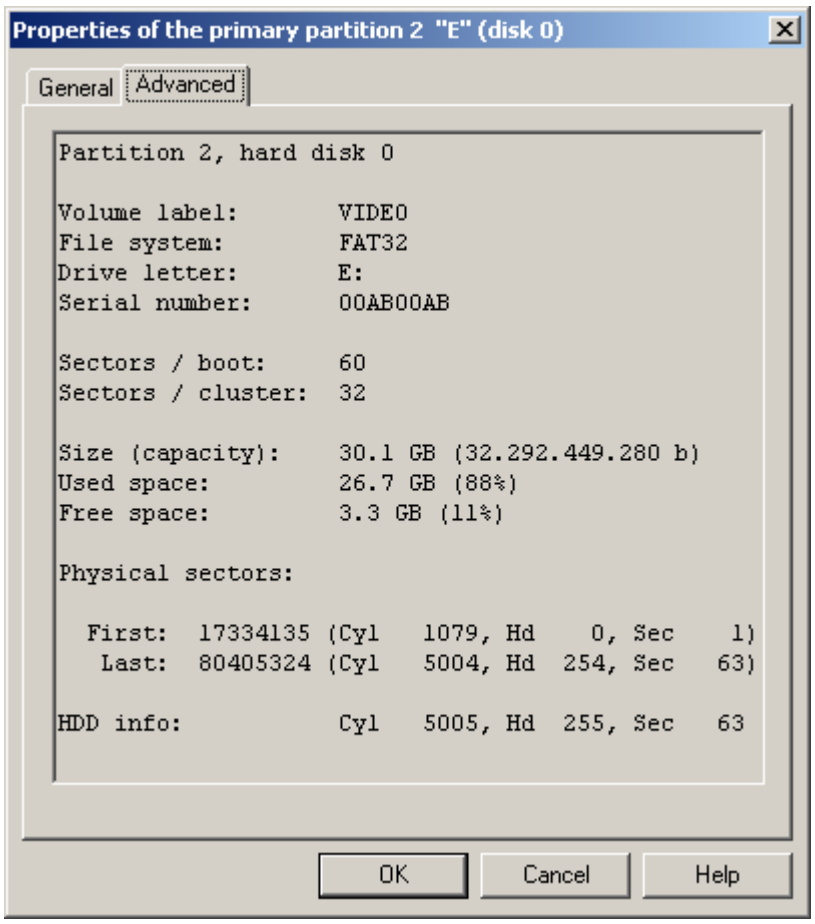
4.18.5.4 Partition Properties

The *Partition Properties* window contains two tabs.

The first one includes general information about a partition that is usually available in Windows for mounted partitions.



This tab displays filesystem type, volume label, capacity, amount of used and free space, Cluster Size value, the size of the Bootable Area and the capacity of the Root Directory. The Root Directory capacity is valuable only for the FAT16 filesystem. The program displays zero Root entries for other filesystems.



The second tab contains advanced information about a partition:

Partition #, disk #	Includes the partition's index on a disk, and the disk index as reported by the operating system.
Volume label	Partition's Volume Label value, which is saved in the boot sector, available for FAT16, FAT32, NTFS and HPFS filesystems
File system	Filesystem type (only for known filesystems)
Drive letter	Drive letter that is assigned to a partition in the operating system. In DOS, the program can fail revealing drive letters for partitions being mounted by IFS drivers (such as NTFSDOS)
Serial number	Partition's Serial Number taken from the boot sector
Sectors / boot	The *Bootable Area* size
Sectors / cluster	The *Cluster Size* value
Size (capacity)	The absolute partition's capacity: <Capacity> = (<*Last Sector*> – <*First Sector*>) * <*Sector Size*> In fact, this value is a little bit greater than the filesystem capacity.
Used space	Includes files, filesystem metadata and reserved space: <*Used Space*> = <*Capacity*> – <*Free Space*>
Free space	Amount of free space taken from the filesystem.
First physical sector	The address of the first partition's sector, expressed in both the C/H/S and the linear addressing formats
Last physical sector	The address of the first partition's sector, expressed in both the C/H/S and the linear addressing formats
HDD info	The actual *Hard Disk Geometry* of a disk holding the partition.

4.18.6 Show Archive Info

This chapter describes how to inspect contents of existing backup archives by using Hard Disk Manager.

4.18.6.1 Overview

The function *Show Archive Info* allows preliminary inspecting backup archives for better recognizing their contents, finding images of required partitions or hard disks, browsing archived contents and verifying image integrity.

Hard Disk Manager keeps a separate list of archives that were created or opened by the program. The Archives List panel displays archives placed in this list. By default, the program includes every successfully created archive in the List. In addition, the program includes each backup archive that is opened for the first time, by the following functions:

- ⇨ *Show Archive Info*
- ⇨ Restore Partition
- ⇨ Restore Hard Disk
- ⇨ Selective Partition Restore
- ⇨ Check archive integrity
- ⇨ Browse Archive

The list of archives can be edited manually, the *popup menu* for the Archive List panel includes **Add archive** and **Delete archive** items, which allow customizing the archive.

4.18.6.2 Inspecting archives listed in the Archives List panel

To inspect a listed archive, use the following routine:

Step 1. Select an interesting archive

Select an archive in the Archives List panel.

Step 2. Select the operation

Call the *popup menu* for the selected archive and then choose the item:

Show archive info…

After these actions, the *Archive Info* dialog should appear (see next sections for more details and its functionality explanation).

4.18.6.3 Inspecting arbitrary archives

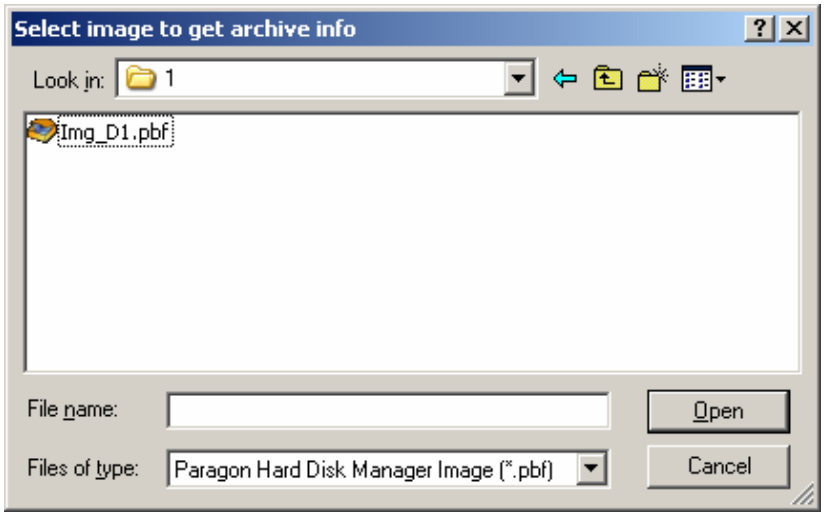
To inspect an archive that is not included in the Archives List yet, use the following procedure:

Step 1. Select the operation

Select the menu item in the main program's menu:

General ➔ Show archive Info

Step 2. Select an archive file



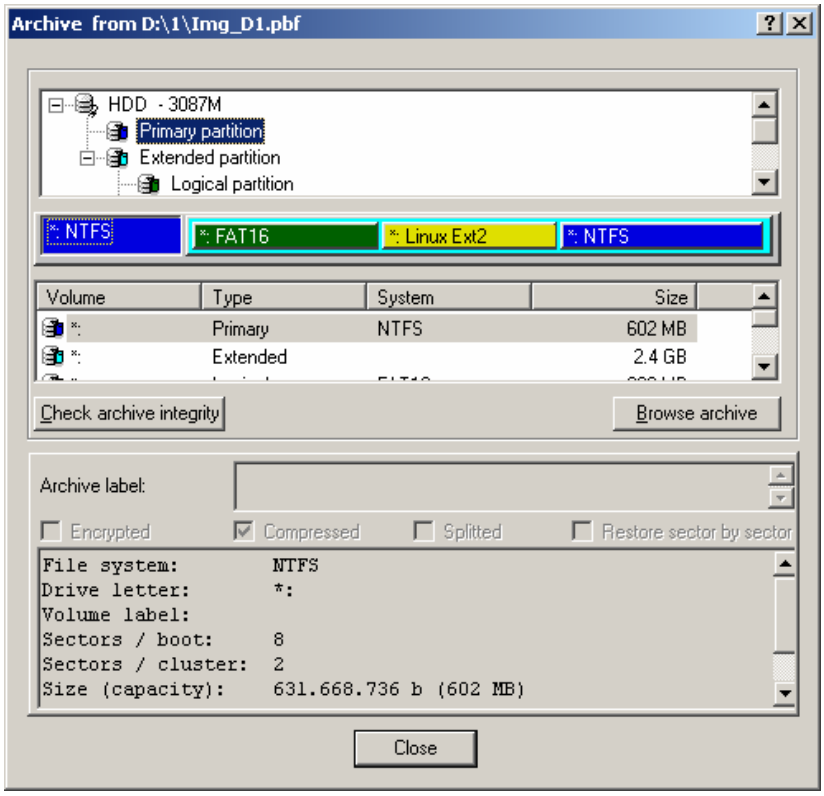
The program displays the advanced *Open File* dialog, which allows selecting files on:

- Mounted local volumes
- Mounted local removable media
- Mapped network drives
- Unmounted FAT16, FAT32, NTFS, Ext2 and Ext3 partitions

The functionality of the advanced *Open/Save File* dialog is explained in the section Backup Partition ➔ Selecting unmounted partitions as the target location for saving backup images.

After these actions, the *Archive Info* dialog should appear (see the next section for more details and its functionality explanation).

4.18.6.4 Description of the Archive Info dialog



Let's remind that Hard Disk Manager supports multiple formats of backup archives:

- ⇨ Archives of a single partition
- ⇨ Archives of multiple partitions
- ⇨ Archives of hard disks
- ⇨ Archives of 1st track
- ⇨ Archives of MBR

The *Archive Info* dialog reveals contents of archives of any type in similar fashion. Archives of a hard disk are the most complex ones. The picture placed above displays layout of a hard disk archive.

The **top section** of the dialog contains the *Tree Layout*, *UDP* controls and the *List of Partitions* panels. They show the disk layout and basic properties of partitions.

When an object is selected in either of these panels, extended information about the selected object is displayed in the **bottom section** of the dialog.

In addition, the dialog provides the ability of performing some operation on the selected archive:

Check archive integrity

Press this button to perform the Check archive integrity operation on the selected archive.

Browse archive

Press this button to perform the Browse archive operation on the selected archive.

4.18.6.5 Comments

The **bottom section** of the dialog displays the following information about selected objects:

Partition

File system	File system that is placed on the selected partition
Drive letter	--- not available for partitions from backup images ---
Volume label	The partition's Volume Label. It can be useful for distinguishing between partitions in case of selective restoration of partitions.
Sectors/boot	The Bootable Area size.
Sectors/Cluster	*Cluster Size* value expressed in Sectors. Halve this value to get the *Cluster Size* value in Kbytes.
Size (capacity)	The partition's Capacity (in bytes and Mbytes)
Used space	Amount of usable data on the selected partition
Free space	Amount of free space on the selected partition

Hard disk, 1st Track, MBR

Cylinders	Reveals the *Hard Disk Geometry*
Heads	
Sectors/Track	
Total Sectors	The hard disk capacity expressed in Sectors

4.18.7 Browse Partition

4.18.7.1 Overview

The windows-based version of Hard Disk Manager is integrated with the *Paragon Partition Explorer* utility, which provides the ability of browsing mounted and unmounted partitions of FAT16, FAT32, NTFS, Ext2 and Ext3 filesystem type.

The Partition Explorer utility is described in the chapter Paragon Partition Explorer. Briefly, Partition Explorer allows browsing, exporting and importing files and folders from a partition, renaming and deleting files and folders, creating new folders and open documents by associated applications.

4.18.7.2 Starting the operation

Step 3. Select a partition to browse

Select an existing *Primary* or *Logical* partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. The operation is not enabled for the Extended Partition and for blocks of free space.

Step 4. Select the operation to perform

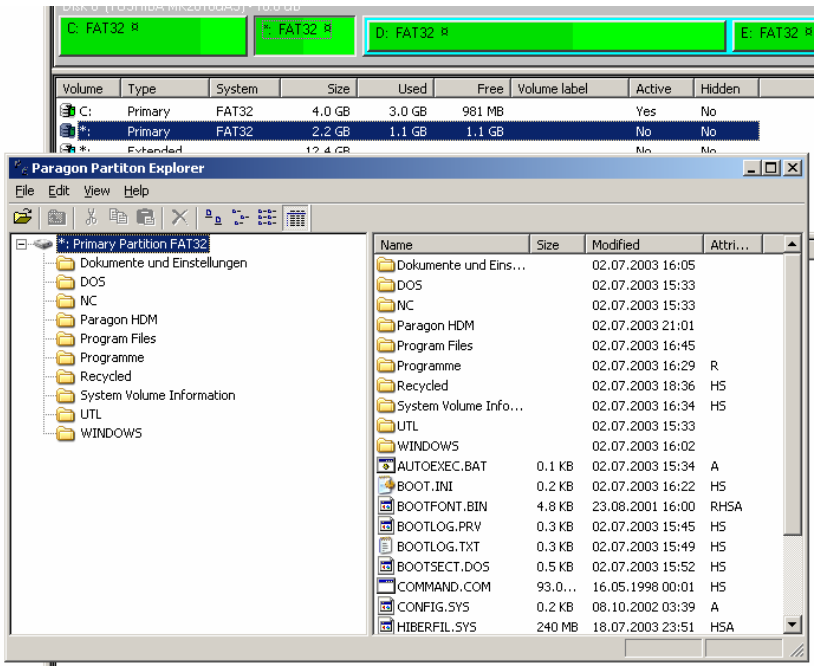
- Select in the main menu:

  Partition ➔ Browse partition

- Call the *popup menu* for the selected partition, then select the menu item:

  Browse partition

After these actions, Hard Disk Manager will start the Partition Explorer utility to browse the selected partition:



See the chapter Paragon Partition Explorer to learn more about usage of the utility.

4.18.8 Browse Archive

4.18.8.1 Overview

The windows-based version of Hard Disk Manager is integrated with the *Paragon Image Explorer* utility, that provides the ability of browsing contents of partitions of FAT16, FAT32, NTFS, Ext2 and Ext3 filesystem type, which are saved in a backup archive.

The Image Explorer utility is described in the chapter Paragon Image Explorer. Briefly, Image Explorer allows browsing and exporting files and folders from an archive and opening documents by associated applications.

4.18.8.2 Starting the operation

Step 1. Select an archive to browse

Select an archive presented in the Archives List panel.

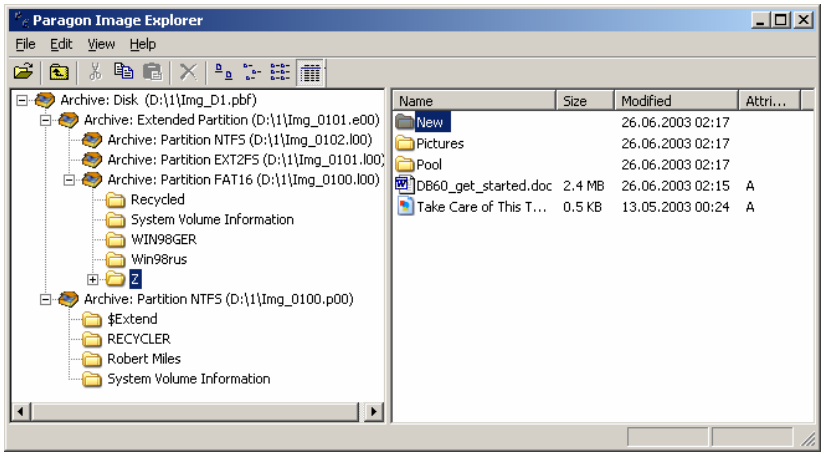
In case of an interesting archive is not listed yet, add it to the Archives List by calling the popup menu for the Archives List panel and by selecting the item **Add archive…**

Step 2. Select the operation to perform

Call the *popup menu* for the selected archive, then select the menu item:

Browse archive

After these actions, Hard Disk Manager will start the integrated *Image Explorer* utility for browsing the selected archive.



See the chapter Paragon Image Explorer to learn more about usage of the utility.

4.18.9 View Sectors

4.18.9.1 Overview

Hard Disk Manager includes a simple disk editing tool, which allows directly accessing and editing disk sectors, saving and restoring sectors from files, navigating through filesystem metadata and so on.

The built-in *Disk Editor* is not an end-user tool. The current version of Disk Editor can be used primarily for troubleshooting purposes.

4.18.9.2 Starting the Disk Editor

The *View Sectors* function can be applied to partitions and hard disks only. Hard Disk Manager limits Disk Editor to navigate only within sectors of a selected object: if a partition was selected, the program can explore only partition's sectors.

Step 1. Select an object to view its sectors

Select a partition or a hard disk, which needs to be explored, in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. In any case, the object will be highlighted in all three panels. The function is not available for *blocks of free space*.

Step 2. Select the operation to perform

If a hard disk is selected:

- Select the item in the main menu:

  Hard disk ➔ View sectors

- Call the *popup menu* for the selected hard disk, then select the item:

  View sectors

If a partition is selected:

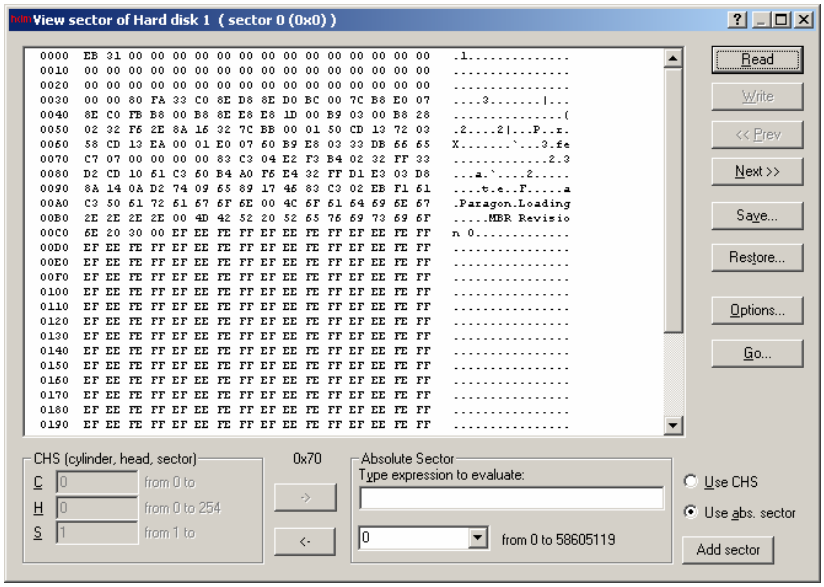
- Select the item in the main menu:

  Partition ➔ Modify ➔ View sectors

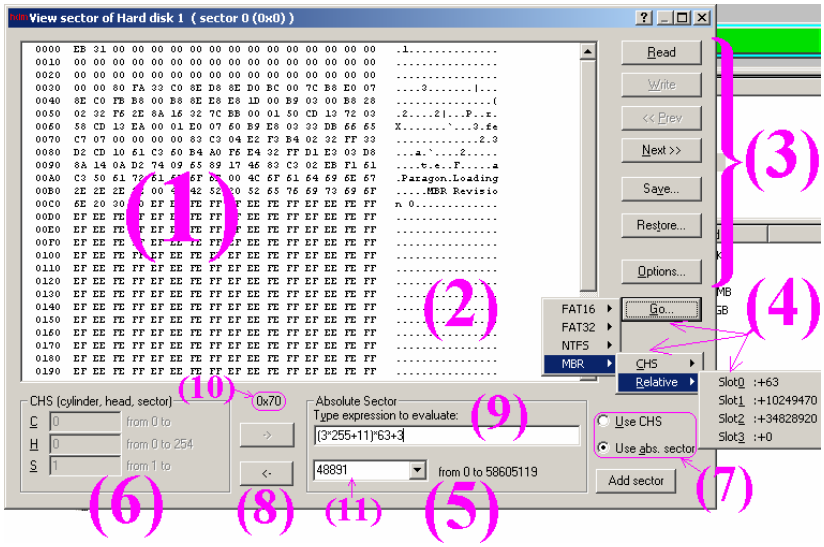
- Call the *popup menu* for the selected hard disk, then select the item:

  Modify ➔ View sectors

After these actions, the *View Sectors* dialog appears:



4.18.9.3 Functionality description



The *View Sectors* window includes the following components:

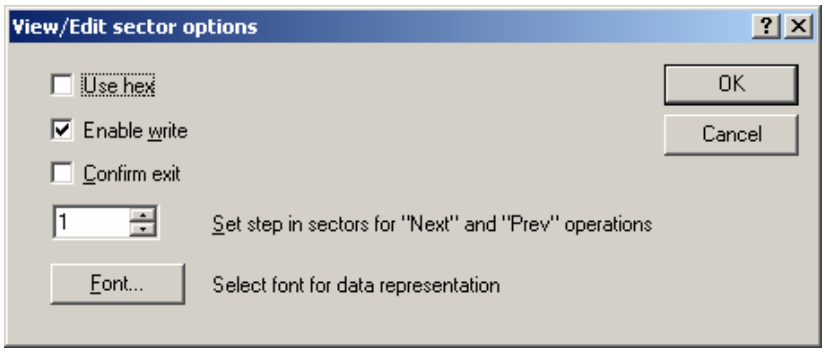
1. The hexadecimal representation of the sector's contents. Contents are divided into lines, 16 bytes per line. Each byte is represented by the two hexadecimal digits (0..9, A..F). The left-most column contains the offset values of lines within the sector (expressed in hexadecimal numbers, 0x000 to 0x200).
   The "hex numbers" text can be edited and written back to the disk.
2. The textual representation of the sector's contents. Contents are divided into lines, which are 16 characters long. This text cannot be edited.
3. Controlling buttons:

Read	Re-read sector contents. All changes made in the editor are cancelled.
Write	Write changes to the disk.
<<Prev	Show to the "previous" sector. The real step is defined in the Options.
Next>>	Show to the "next" sector.
Save…	Save a group of sectors (beginning from the current one), in a file.
Restore…	Overwrite a group of sectors (beginning from the current one), by contents of a file.
Options…	Define options of the Disk Editor.
Go…	Fast navigation between servicing structures (see below).

4. Fast navigation controlling menus.
   The fast navigation menus are available only for sectors, which are "look like" servicing sectors of the *DOS partitioning scheme* or boot sectors of the FAT16, FAT32 or NTFS filesystems.
   The current sector is interpreted as a servicing one, and the program interprets its data as references to most important elements of a filesystems or a partitioning scheme.

5. A group of controls that allow navigating through sectors in the mode of the *absolute addressing of sectors*. In this mode, the index of a sector is independent from the actual *Hard Disk Geometry*.
6. A group of controls that allow navigating through sectors in the mode of *C/H/S addressing of sectors.* In this mode, the address of a sector fundamentally depends on the actual *Hard Disk Geometry*.
7. The group of the *choice* controls that alter the current addressing mode.
8. A group of buttons that allow translating a sector's address between the two addressing modes.
   They do not alter the actual addressing mode but only allow inspecting the sector's address in the other addressing model.
9. The textual box that allows setting the sector's address by a formula.
   Enter a valid arithmetic expression in this text box and press the ENTER key to jump to the appropriate sector.
10. The textual mark that displays the offset of the currently selected byte in the hexadecimal text (section (1)). When the textual cursor is moved through the hex numbers, this mark displays the actual position within the sector.
11. This control allows manually entering the number of a sector to be displayed next. The pull-down list contains the list of favorite addresses. Press the **Add sector** button in order to add the number of the current sector to the list of favorite addresses.

4.18.9.4 Disk Editor Options



Use hex

Set this option ON in order to force the program displayed sector address value in the hexadecimal format.

Enable write

Set this checkbox in order to be able editing sector contents and restoring sectors from files.

Confirm exit

If this option is activated, the program produces a confirmation message when exiting the Disk Editor tool. This feature just prevents from occasional closing of the *View Sectors* window.

Set step in sectors for "Next" and "Prev" operations

This control allows regulating the amount of sectors that will be skipped when jumping to a next or previous sector.

Font…

Press this button in order to select the font, which will be used for displaying sector contents in the View Sectors dialog.

4.18.9.5 Fast navigation menu

The *Fast navigation menu* allows quickly positioning the Disk Editor to some locations on a disk or a partition, which contain system information.

The Fast navigation menu is activated by the button **Go…**

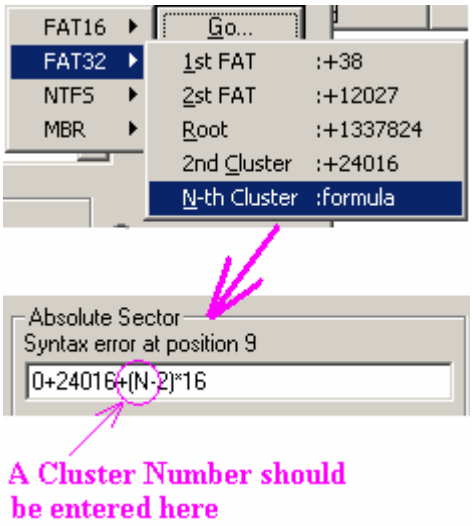
Structure

FAT16		Interpret the sector as the Boot Sector of a FAT16 filesystem
	1st FAT	Find the first copy of FAT (the 1st sector occupied by FAT)
	2nd FAT	Find the second copy of FAT
	Root	Find the Root Directory
	2nd Cluster	Find the 2nd cluster of the FAT16 partition
	Nth Cluster	Find the Nth cluster.

		See the section Navigating the N[th] Cluster for more details.
FAT32		Interpret the sector as the Boot Sector of a FAT32 filesystem
	1[st] FAT	Find the first copy of FAT (the 1[st] sector occupied by FAT)
	2[nd] FAT	Find the second copy of FAT
	Root	Find the Root Directory
	2[nd] Cluster	Find the 2[nd] cluster of the FAT16 partition
	N[th] Cluster	Find the N[th] cluster. See the section Navigating the N[th] Cluster for more details.
NTFS		Interpret the sector as the Boot Sector of a NTFS filesystem
	MFT	Find the 1[st] sector occupied by the MFT (Master File Table)
	MFT mirror	Find the 1[st] sector occupied by the MFT-mirror (a servicing object, that is the backup copy of the MFT's 1[st] cluster)
	N[th] Cluster	Find the N[th] cluster. See the section Navigating the N[th] Cluster for more details.
MBR		Interpret the sector as a Partition Table sector (MBR or EPT)
	CHS	Use the *C/H/S addresses* of partitions
		Slot 0	Find a partition that is referenced by the Slot#0 of MBR/EPT
		Slot 1	Find a partition that is referenced by the Slot#1
		Slot 2	Find a partition that is referenced by the Slot#2
		Slot 3	Find a partition that is referenced by the Slot#3
	Relative	Use *linear addresses* of partitions
		Slot 0	Find a partition that is referenced by the Slot#0 of MBR/EPT
		Slot 1	Find a partition that is referenced by the Slot#1
		Slot 2	Find a partition that is referenced by the Slot#2
		Slot 3	Find a partition that is referenced by the Slot#3

4.18.9.6 Navigating the N[th] Cluster

When the item **"N[th] Cluster"** of the *Fast navigation menu* is selected, the program places an arithmetic expression in the *Formula Box* (9):



The expression includes a character "N", which should be replaced manually by a number of an interesting cluster.

4.18.10 Generate scripts

4.18.10.1 Overview

Hard Disk Manager provides the ability of the batch processing of disk management operations. Hard Disk Manager includes separate command line utilities for working in the unattended mode, which are generally named *Paragon Script Interpreter* (PSI):

- The utility **SCRIPTS.EXE** is included to the Windows-based version
- The utility **PSI.EXE** is included to the DOS-based version
- The utility **PSI.** is included to the Linux-based version

These utilities take job tasks from so-called *script files*. A *script file* is just a textual file written in the *Paragon Scripting Language*, it contains the description of operations to be executed and program's settings that should be applied to these operations.

The interactive versions of Hard Disk Manager provide the ability of generating a script file from accumulated virtual operations. The resulting script file can be used "as is", or it can be used as a template for building custom script files.

With using this feature, one can automate the disk/partition backup routines or cloning procedures of almost any complexity. The *Paragon Scripting Language* supports all operations that are available in interface-managed versions. In addition, it supports the conditional execution, subroutines, repeatable iterations, disk/partition properties analysis, errors management and even the controllable console-like input-output features.

4.18.10.2 Running the operation

Let's remind that Hard Disk Manager generates scripts only from virtual operations that are accumulated in the *List of Pending Operations*.

Step 1. Switch on the Virtual Execution mode for all operations

Open the program's Settings and force the virtual execution for all operations (see the chapter Settings overview for more details):

(menu) General ➔ Settings ➔
(tab) Operations ➔ Virtual operations are enabled

Step 2. Execute required operations virtually

The next step is to emulate all operations that should be executed by the Script Interpreter in the automated mode. A user should execute these operations virtually in the Windows-based interactive application.

(!) Do not apply accumulated pending operations!

Step 3. Select the operation to perform

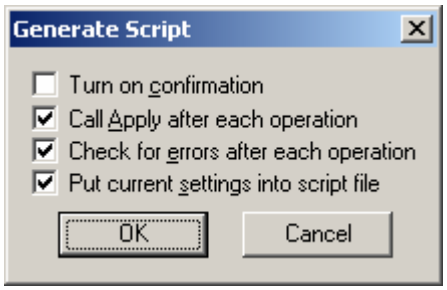
Select the following menu item:

General ➔ Generate Script

This menu item remains disabled if there are no operations accumulated in the *List of Pending Operations*.

Step 4. Define parameters of the operation

The following dialog should appear:

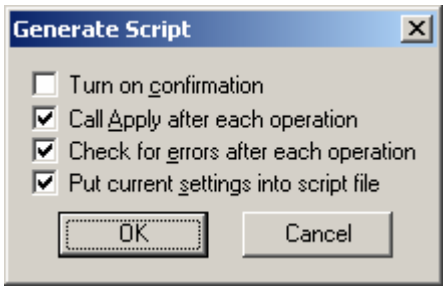


A user is able customizing some supplementary features of the generated script file, such as including of error management, enabling or disabling interactivity with a user and so on. See the next section for more details.

Step 5. Select a filename for a newly created script

Finally, the program suggests choosing a filename for a new *script file*. The default file extension that is reserved for scripting files is .**PSL**. However, a script can be saved under any filename.

4.18.10.3 Description of parameters



This dialog allows controlling a script's code that will be generated.

Turn on confirmation

When this option is active, the program inserts the command:

```
confirmation on
```

This command enables to pause the script execution in order to acquire input information from a user. For example, the program will pause when creating a multivolumic archive (with the volume filenames generating option switched off), when the program needs erasing a non-empty rewritable CD and so on.

When the confirmation mode is switched off, the program does not pause the script execution. Instead, it behaves as if a user always selects a default answer to each question. In this mode, the program really does not need a user intervention. However, in some cases the program will simply abort operations.

Call Apply after each operation

When this option is active, the program inserts the `apply all` command after each disk management operation. Otherwise, the `apply all` command will be included only once, after the last operation.

The thing is that the script processing module works like the interactive application, it accumulates operations in its own List of Pending Operations. The script command `apply all` is similar to the **Apply** action in the interactive application, it forces executing all pending operations.

There is the difference between the applying of a large list of pending operations and the stepwise execution of multiple operations, e.g. in case of processing of *locked partitions*. See the Paragon Scripting Language Manual for more details.

Check for errors after each operation

When this option is active, the program inserts a code, which checks the status of a last executed operation and halts the script processing on errors of any kind.

This feature is useful, if a set of operations should be applied to the same partition. If some operation fails for any reason, the error checking code cancels sequent data modifications and prevents them from further corruption.

In some situations, this feature can be a drawback, if operations are independent from each other. For example, a backup operator wrote a script that creates an incremental backup of first two partitions of all hard disks, and he intends using this script for unattended processing of office computers. On computers having only single partitions on the 1st disk, this script wouldn't save contents of other hard disks. In case of switching off the error checking (at least between independent operations), the program will save all available partitions.

Put current settings into script file

When this option is active, the program inserts the actual program's settings to the `settings ... endsettings` section.

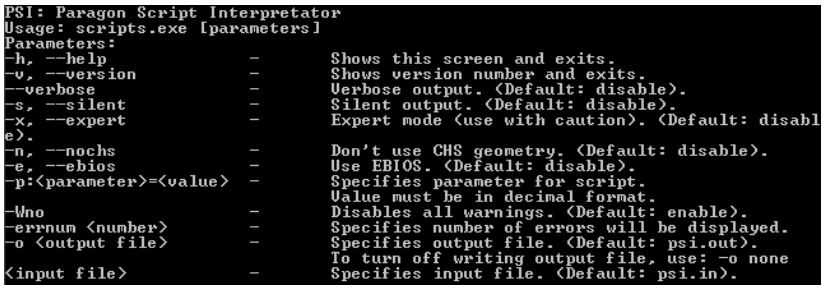
Otherwise, the program omits the `settings ... endsettings` section.

4.18.10.4 Comments

All versions of the *Script Interpreter* have similar usage rules. PSI supports several command line keys. To obtain the usage prompt, one should run the utility from the command line with `-h` parameter:

in Windows:	`scripts.exe -h`
in DOS:	`psi.exe -h`
in Linux:	`psi -h`

The program will display the prompt screen:



The Windows-based utility SCRIPTS.EXE

The **SCRIPTS.EXE** utility is located in the same directory with the Windows-based application of Hard Disk Manager. By default, it is the folder:

```
C:\Program Files\Paragon Software\Hard Disk Manager\WinHDMTL
```

The DOS-based utility PSI.EXE

The **PSI.EXE** utility is located in the `scripts` subfolder of the directory containing the DOS-based version of Hard Disk Manager. By default, it is the folder:

```
C:\Program Files\Paragon Software\Hard Disk Manager\DosHDMTL\scripts
```

The program is packed in the **SRPTPACK.EXE** self-extracting archive. Before using the utility, one should extract all contents of the **SRPTPACK**.

The Linux-based utility (PSI.)

The **PSI.** utility is located on the Linux-based bootable CD of Hard Disk Manager, in the folder:

```
/usr/local/bin
```

and in the original location on the bootable CD:

```
/mnt/cdrom/usr/local/bin
```

However, the on-CD Linux is configured so as a user can run the utility from any location.

How to execute scripts

All versions of the *Script Interpreter* are used in the same fashion:

1. Run an interactive version of Hard Disk Manager and prepare a script file as it described in the above sections.
2. Run the *Script Interpreter* from the command line in the following fashion:

In Windows: *<path>***scripts.exe** *<path><script-file>*

In DOS: *<path>***psi.exe** *<path><script-file>*

In Linux: **psi** *<path><script-file>*

The *Script Interpreter* supports the *silent* and the *verbose* working modes. In the verbose mode, the program outputs the detailed report executed operations on the console. In the silent mode, no information is displayed.

However, in any of these modes the *Script Interpreter* produces the report file that is named **PSI.OUT**. In addition, the utility also keeps two log files, the **STUBACT.LOG** and the **PWLOG.TXT**.

4.18.11 Show CD/DVD Burners info

4.18.11.1 Overview

Hard Disk Manager allows obtaining the list of available CD and DVD writing drives that can be used in the function Burn image of a partition on CD/DVD-R(W). This function is available only in the Windows-based and the Linux-based versions of the program.

The DOS-based version of Hard Disk Manager does not support CD and DVD burners. It does not provide an ability of burning archives on recordable CD/DVD discs.

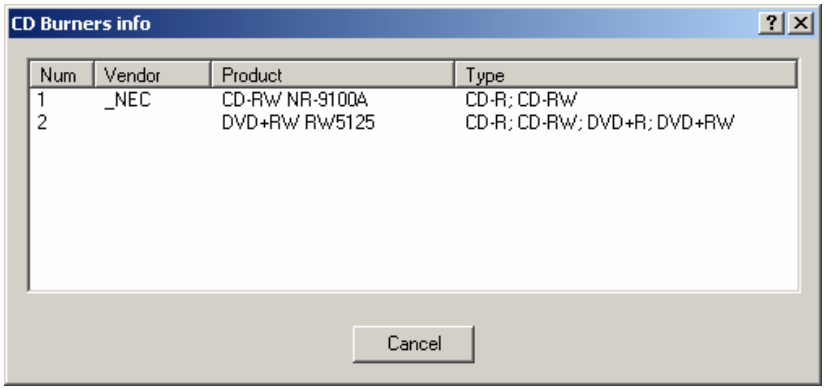
In Windows environment, Hard Disk Manager can use all CD/DVD burners supported by the Win32 ASPI service: these devices should be listed in the Device Manager, but they need not having drive letters assigned.

4.18.11.2 Running the operation

To obtain the list of available CD and DVD burning devices, select the menu item:

General ➔ CD/DVD Burners…

The program will display the **CD Burners Info** window:



The program reveals the following information about burning devices:

Vendor	OEM code (Original Equipment Manufacturer code)
Product	The model name. This value is provided by the device drivers, so that it can differ from the device model in case of using generic and non-original drivers.
Type	Lists all supported media types. (in fact, this the only important information about a CD/DVD burning device).

4.18.12 Change Root Size

This chapter explains how to change the capacity of the Root Directory of formatted FAT16 partitions with keeping data intact, by using Hard Disk Manager.

4.18.12.1 Overview

The maximum capacity of the *Root Directory* is the important parameter of "elder" FAT12 and FAT16 filesystems. On FAT12 and FAT16 partitions, the maximum capacity of the Root Directory is defined at the moment of formatting a partition (see the Glossary).

A user may have a need of changing the capacity of the Root Directory:

- The Root Directory should be enlarged in order to enable placing more files and directories in the drive's root.
- The Root Directory should be reduced in order to increase the common file space.

Common disk management tools and operating systems usually do not provide an ability of managing the capacity of the Root Directory.

Hard Disk Manager allows arbitrary assigning the Root Directory capacity at the formatting a FAT16 partition (see the chapter Format Partition). In addition, the program allows changing the capacity of the Root Directory without destroying existing data.

4.18.12.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a FAT16 partition for changing the Root Directory size

Select a primary or logical partition formatted to FAT16, in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. In any case, it will be highlighted in all three panels.

Step 2. Select the operation to perform

Variants:

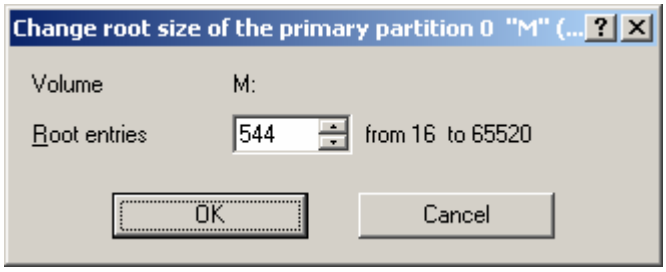
- Select in the main menu:

  Partition ➔ Modify ➔ Change Root size…

- call the *popup menu* for the selected partition and select the item:

  Modify ➔ Change Root size…

Step 3. Define parameters of the operation



You are able assigning the maximum capacity of the Root Directory.

Initially, the program suggests using the actual value of this parameter that is meaning making no changes in the partition's layout.

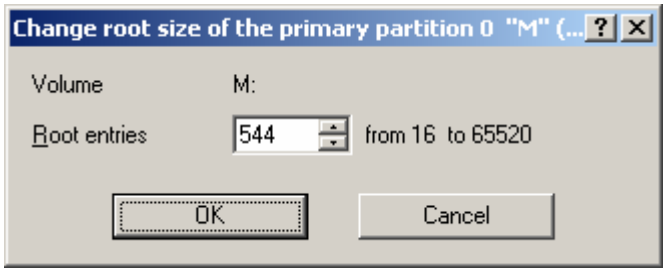
After choosing desired operation parameters, one should press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Change Root Size* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.18.12.3 Description of parameters



The dialog displays the selected value of the Root Directory capacity and the available range for this value.

Root entries

This *spinner* control contains the selected value of the *Root Directory Size* filesystem's parameter. The text on the right side displays the available range of values.

The program accommodates the Root Directory capacity to the FAT16 filesystem standards, so that the program rounds an entered value to a nearest valid one.

Volume

This text contains a drive letter that is assigned to the selected partition. It is used for notification purposes only. In case of no drive letter is assigned, the text "***:**" is displayed.

4.18.12.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed
- Amount of moved and remaining data
- Information about actions being performed.

The actual time required for the completing the *Change Root Size* operation depends on the actual allocation of data on the modified partition.

4.18.12.5 Comments

About the maximum amount of files in the Root

The maximum amount of files and directories, which can be placed in the Root Directory on a FAT16 partition, can be actually equal or less than the Root Directory Capacity value.

The thing is that the Root Directory Capacity is the total amount of *directory entries* in the Root Directory. *Directory entry* is 32 bytes long record purposed for saving the reference to a file.

In FAT family filesystems, a *short filename* of a file takes one directory entry, while a long filename actually occupies multiple directory entries. If some files or directories in the root directory have long filenames, the maximum amount of files is less than the Root Directory capacity.

4.18.12.6 Changing the Root Size on locked partitions

The difference between *locked* and *unlocked* partitions is that locked ones are used by other programs for file input-output activity (see Glossary).

Hard Disk Manager requires the exclusive controlling of a partition when changing the capacity of the Root Directory. This operation is incompatible with any other data activity, because the program needs temporarily distort filesystem data in order to successfully complete the operation.

Hard Disk Manager requires rebooting the system to a single-tasking environment in order to complete the *Change Root Size* operation, in case of a locked partition is selected.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

The detailed explanation of stepwise actions is given in following sections:

- Working with locked/system partitions and hard disks
- Backup system and locked partitions

4.18.13 Change Boot Size

This chapter explains how to change the size of the Boot area of formatted FAT16 and FAT32 partitions with keeping data intact, by using Hard Disk Manager.

4.18.13.1 Overview

In almost all known filesystems, first sectors of a partition are used for saving a *bootable code* of an operating system. This region of a partition is generally named the *Bootable Area*, or the *Boot Sector* (actually, it takes multiple sectors).

The Bootable Area holds only the starting part of a bootable code, which is responsible for detecting, initializing and starting the *OS kernel*.

In FAT family filesystems, the size of the Bootable Area can vary within the predefined limits. Potentially, this feature can be used for changing the alignment of the filesystem contents.

Common disk management tools and operating systems usually do not provide an ability of managing the size of the Bootable Area.

Hard Disk Manager allows arbitrary assigning the Bootable Area size at the formatting FAT partitions (see the chapter Format Partition). In addition, the program allows changing this value on formatted FAT partitions without destroying existing data.

4.18.13.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a FAT partition for changing the Boot Area size

Select a primary or logical FAT16 or FAT32 partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. In any case, it will be highlighted in all three panels.

Step 2. Select the operation to perform

Variants:

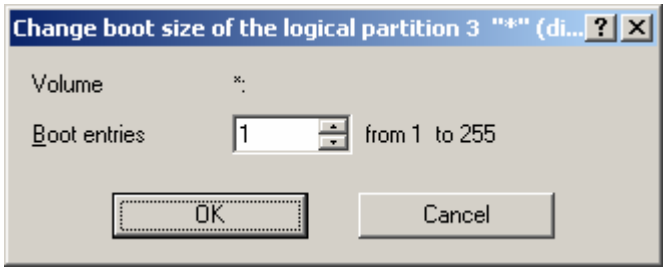
- Select in the main menu:

  Partition ➔ Modify ➔ Change Boot size…

- call the *popup menu* for the selected partition and select the item:

  Modify ➔ Change Boot size…

Step 3. Define parameters of the operation



You are able assigning the amount of sectors reserved for the Boot Sector.

Initially, the program suggests using the actual value of this parameter, which is meaning making no changes in the partition's layout.

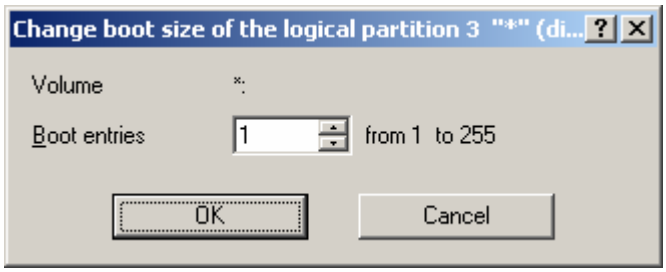
After choosing desired operation parameters, one should press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Change Boot Size* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.18.13.3 Description of parameters



The dialog displays the selected value of the Bootable Area size and the available range for this value.

Root entries

This *spinner* control contains the selected value of the *Boot Size* filesystem's parameter. The text on the right side displays the available range of values.

Volume

This text contains a drive letter that is assigned to the selected partition. It is used for notification purposes only. In case of no drive letter is assigned, the text "***:**" is displayed.

4.18.13.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed
- Amount of moved and remaining data
- Information about actions being performed.

The *Change Boot Size* operation takes noticeable time, because it actually requires moving all other contents of a partition for several sectors forward or backward. The actual time required for the completing the operation depends on the amount of data that should be processed.

4.18.13.5 Changing the Boot Size on locked partitions

The difference between *locked* and *unlocked* partitions is that locked ones are used by other programs for file input-output activity (see Glossary).

Hard Disk Manager requires the exclusive controlling of a partition when changing the Bootable Area size. This operation is incompatible with any other data activity, because the program needs temporarily distort filesystem data in order to successfully complete the operation.

Hard Disk Manager requires rebooting the system to a single-tasking environment in order to complete the *Change Boot Size* operation, in case of a locked partition is selected.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

The detailed explanation of stepwise actions is given in following sections:

- Working with locked/system partitions and hard disks
- Backup system and locked partitions

4.18.14 Change Serial Number of a Partition

This chapter explains how to change the partition's Serial Number by using Hard Disk Manager.

4.18.14.1 Overview

FAT16, FAT32, HPFS and NTFS filesystems include the *Serial Number* parameter (see the Glossary). The partition's Serial Number is saved in the bootsector, its value is generated at the moment of formatting a partition.

Common disk management tools and operating systems generate partition's Serial Number automatically and do not provide an ability of customizing its value.

Hard Disk Manager allows arbitrary changing the partition's Serial Number on formatted FAT16, FAT32, HPFS and NTFS partitions without re-formatting.

4.18.14.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a partition for changing the Serial Number

Select a primary or logical FAT16, FAT32, HPFS or NTFS partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. In any case, it will be highlighted in all three panels.

Step 2. Select the operation to perform

Variants:

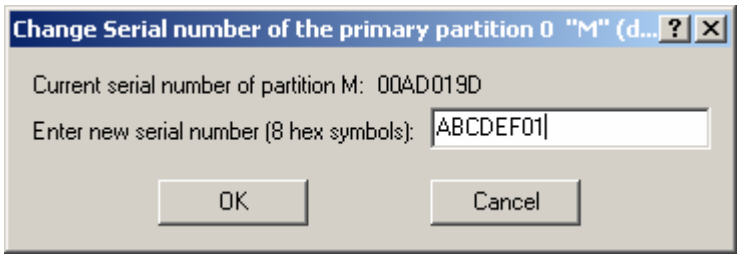
- Select in the main menu:

  Partition ➔ Modify ➔ Change Serial Number…

- call the *popup menu* for the selected partition and select the item:

  Modify ➔ Change Serial Number…

Step 3. Define parameters of the operation



You are able assigning the new Serial Number value.

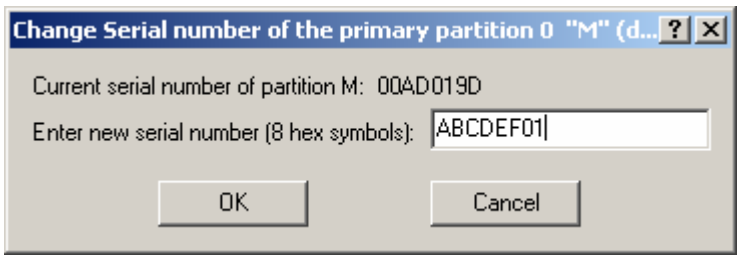
After choosing desired operation parameters, one should press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Change Serial Number* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.18.14.3 Description of parameters



The dialog displays the current value of the Serial Number and allows customizing it.

Enter new Serial Number (8 hex symbols)

A user should enter the new Serial Number value in this text box.

The Serial Number should contain 8 hexadecimal figures (0..9 or A..F). The **OK** button is disabled until a user enters all 8 symbols.

Current Serial Number of partition

This text contains a drive letter and the current Serial Number of the selected partition. It is used for notification purposes only. In case of no drive letter is assigned, the text "***:**" is displayed.

4.18.14.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed
- Amount of moved and remaining data
- Information about actions being performed.

The *Change Serial Number* operation takes just a fraction of a second for the completion.

4.19 Wipe Partition

This chapter explains how to wipe contents of separate partitions and disk space on local drives.

4.19.1 Overview

Wipe Partition function allows irreversibly destroying all contents of a partition by overwriting all of its sectors with unusable data.

This function can be usable, if a user intends:

- destroying on-partition data without an ability of restoration any of their parts;
- reselling or renting a workable hard disk;
- surely exclude any traces of old data on a newly formatted partition;
- destroying non-standard protection/registration/deactivation hidden marks made by some software;

Usually, operating systems and disk management software just remove references to a partition from the Partition Table when deleting a partition. Specially designed software allows retrieving deleted partitions back, with complete restoring data accessibility. For example, Hard Disk Manager supports Undelete Partition function.

When implementing *Wipe Partition* function, the program overwrites all sectors of a partition with non-interpretable information, so that it's impossible recovering previous contents.

Hard Disk Manager allows safely wiping following objects:

- single partitions of any type
- blocks of free space
- entire Extended Partition
- entire Hard Disk (see the chapter Wipe Hard Disk)
- an unused space on partitions of known filesystem type (see the chapter Clear Free Space).

The program allows multi-pass wiping and post-operation checking of a wiped space for clearance. Hard Disk Manager supports only a simplified edition of *Wipe* function: the program chokes up partition space with a single character.

4.19.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select an object you want to wipe

Select an object you want to wipe (see below), in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. In any case, it will be highlighted in all three panels.

The function is available for:

- ⇨ partitions of any type
- ⇨ blocks of free space
- ⇨ the Extended Partition

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

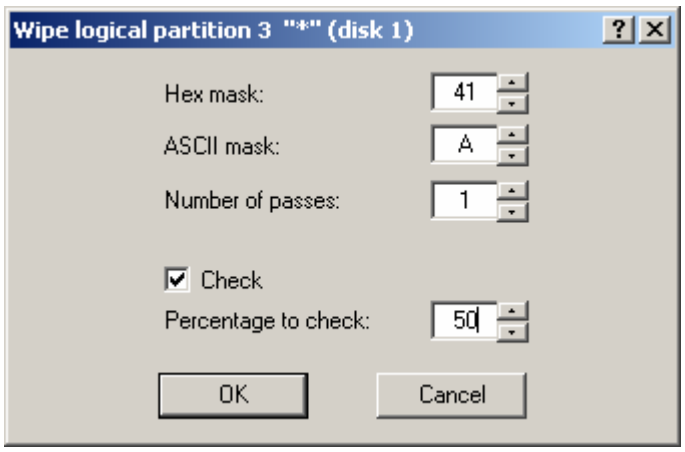
Partition ➔ Wipe partition…

- call the *popup menu* for the selected partition and select the item:

  Wipe partition…

- Press **Alt+W** keyboard combination.

Step 3. Define parameters of the operation



You are able assigning following parameters:

- Amount of wiping passes
- A character that will be used for filling an object
- Enable post-operation clearance check.

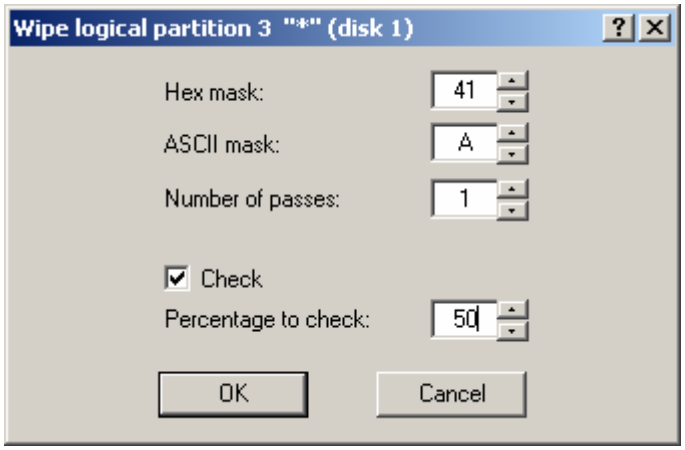
Initially, the program suggests some consistent values for all parameters. In most cases, you just need to press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Wipe Partition* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.19.3 Description of the Wipe Partition parameters



Hex mask

Hard Disk Manager wipes disk space by filling with the single 8-bit character.

The **Hex mask** spinner control allows defining the two-figure hexadecimal presentation of the character being used. The default value is "00" (that is corresponds to the hexadecimal value 0x00). The available range is from "00" to "FF".

This value is synchronized with the **ASCII mask** value.

ASCII mask

Hard Disk Manager wipes disk space by filling with the single 8-bit character.

The **ASCII mask** spinner control allows defining the symbolic presentation of the character being used. The default value is zero character (character #0).

This value is synchronized with the **Hex mask** value.

Number of passes

Set the amount of wiping passes here. The default value is 1, the available range is 1 to 100.

Check

Set this checkmark ON in order to perform post-operation check of wiped disk space for "clearance" – this feature just allows ensuring in wiping.

Percentage to check

The program allows checking for "clearance" all or just a part of wiped disk space.

Set percentage of disk space to be checked in this spinner control. The available range is from 0% to 100%, the default value is 0%.

Percentage	Action
100%	All sectors are checked
0%	Nothing is checked
1-99%	The program checks random sectors over the selected range of disk space.

4.19.4 Running the Wipe Petition operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed

The program treats each wiping pass and the clearance-checking pass as suboperations. In case of multiple wiping passes is selected, the program displays suboperation statistics.

The operation is lengthy. Real performance fundamentally depends on hardware and an operating system being used.

4.19.5 Comments

4.19.5.1 How Hard Disk Manager wipes Extended Partition

When the Extended Partition is selected for wiping, the program destroys:

- All contents of all logical partitions.
- Information about location and type of Logical Partitions.
- The Extended Partition Table, which keeps information about Logical Partitions.

As a result, the Extended Partition becomes empty.

4.19.5.2 If a wiping session was interrupted

In case of *Wipe Partition* operation is interrupted by a user, some data may remain non-destroyed. However, usually these data are of least chance to retrieve, because the program destroys system information in first place.

- If a single partition was selected, the program destroys information about structure of directories and information about allocation of files and directories. An operating system will treat this partition as "unformatted", and there is no chance to "undelete" files and directories. However, contents of some files can be found with special tools, which allow scanning sectors of a hard disk.
- If the Extended Partition was selected, the program destroys information about layout of Logical Partitions. First Logical Partition(s) can be destroyed partially or fully, too. However, Logical Partitions, which are located at the end of the Extended Partition, remain unharmed. These partitions can be restored back with using the Undelete Partition function.

4.19.5.3 When a wiping of free space blocks can be useful

Blocks of free space may appear after deleting, resizing or moving some partitions, and potentially may contain important or confidential data.

The mentioned functionality allows safely destroying this information.

4.19.6 Wiping locked and system partitions

The difference between *locked* and *unlocked* partitions is that locked ones are used by other programs for file input-output activity (see Glossary). Destruction contents of locked partitions is potentially dangerous action, it can lead an operating system to malfunction.

Hard Disk Manager requires rebooting the system to a single-tasking environment in order to complete the Wipe Partition operation, in case of a locked partition is selected.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

The detailed explanation of stepwise actions is given in following sections:

- Working with locked/system partitions and hard disks
- Backup system and locked partitions

4.20 Wipe Hard Disk

This chapter explains how to wipe contents of entire hard disks.

4.20.1 Overview

Wipe Hard Disk function allows irreversibly destroying all contents of all partitions on a hard disk by overwriting all sectors with unusable data.

This function can be usable, if a user intends:

- destroying on-disk data without an ability of restoration any of their parts;
- reselling or renting a workable hard disk;
- surely exclude any traces of old data, which may violate correct working of newly installed software;
- destroying non-standard protection/registration/deactivation hidden marks made by some software;

Usually, operating systems and disk management software limit their actions with just removal of references to partitions from the Partition Table. Specially designed software allows retrieving deleted partitions back, with complete restoring data accessibility. For example, Hard Disk Manager supports Undelete Partition function.

When implementing *Wipe Hard Disk* function, the program overwrites all sectors of a disk with non-interpretable information, so that it's impossible recovering previous contents.

Hard Disk Manager allows safely wiping following objects:

- single partitions of any type (see the chapter Wipe Partition)
- blocks of free space (see the chapter Wipe Partition)
- entire Extended Partition (see the chapter Wipe Partition)
- entire Hard Disk
- an unused space on partitions of known filesystem type (see the chapter Clear Free Space).

The program allows multi-pass wiping and post-operation checking of a wiped space for clearance. Hard Disk Manager supports only a simplified edition of *Wipe* function: the program chokes up partition space with a single character.

4.20.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a hard disk you want to wipe

Select a hard disk you want to wipe in the Tree Layout panel or in the UDP Layout panel. In any case, the hard disk will be highlighted in both panels.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

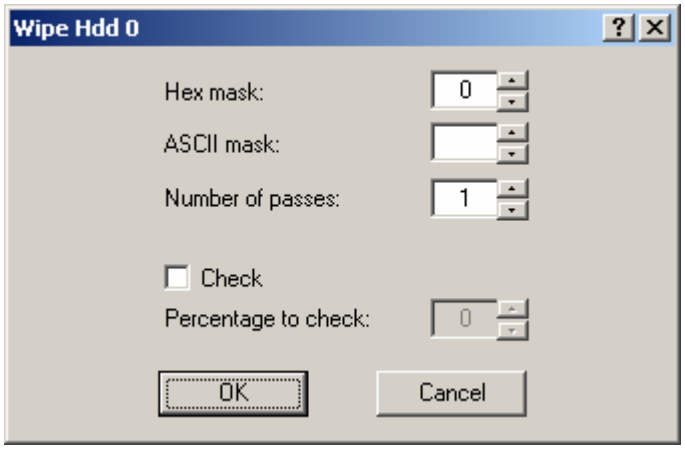
  Hard Disk ➔ Wipe hard disk…

- call the *popup menu* for the selected hard disk and select the item:

  Wipe hard disk…

- Press **Ctrl+W** keyboard combination.

Step 3. Define parameters of the operation



You are able assigning following parameters:

- Amount of wiping passes
- A character that will be used for filling sectors
- Enable post-operation clearance check.

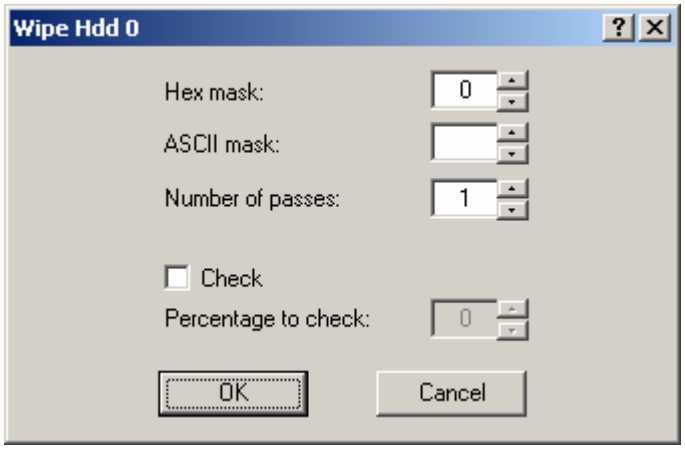
Initially, the program suggests some consistent values for all parameters. In most cases, you just need to press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Wipe Hard Disk* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.20.3 Description of the Wipe Hard Disk parameters



Generally, parameters of this operation are same with ones of the *Wipe Partition* operation. See the section Description of the Wipe Partition parameters for more details.

Hex mask

Hard Disk Manager wipes disk space by filling with the single 8-bit character.

The **Hex mask** spinner control allows defining the two-figure hexadecimal presentation of the character being used. The default value is "00" (that is corresponds to the hexadecimal value 0x00). The available range is from "00" to "FF".

This value is synchronized with the **ASCII mask** value.

ASCII mask

Hard Disk Manager wipes disk space by filling with the single 8-bit character.

The **ASCII mask** spinner control allows defining the symbolic presentation of the character being used. The default value is zero character (character #0).

This value is synchronized with the **Hex mask** value.

Number of passes

Set the amount of wiping passes here. The default value is 1, the available range is 1 to 100.

Check

Set this checkmark ON in order to perform post-operation check of wiped disk space for "clearance" – this feature just allows ensuring in wiping.

Percentage to check

The program allows checking for "clearance" all or just a part of wiped disk space.

Set percentage of disk space to be checked in this spinner control. The available range is from 0% to 100%, the default value is 0%.

Percentage	Action
100%	All sectors are checked
0%	Nothing is checked
1-99%	The program checks random sectors over the selected range of disk space.

4.20.4 Running the Wipe Hard Disk operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time

- Averaged read & write speed

The program treats each wiping pass and the clearance-checking pass as suboperations. In case of multiple wiping passes is selected, the program displays suboperation statistics.

The operation is lengthy. Real performance fundamentally depends on hardware and an operating system being used.

4.20.5 Comments

4.20.5.1 If a wiping session was interrupted

In case of the *Wipe* operation was interrupted, some data remain non-destroyed and can be restored back. The program surely destroys Partition Table, so that a hard disk looks empty.

However, partitions located at the end of disk space remain unharmed. These partitions can be restored back with using the Undelete Partition function.

4.20.6 Wiping locked hard disks

The difference between *locked* and *unlocked* hard disks is that locked ones contain partitions, which are used by other programs for file input-output activity (see Glossary). Destruction contents of locked hard disks is potentially dangerous action, it can lead an operating system to malfunction.

Hard Disk Manager requires rebooting the system to a single-tasking environment in order to complete the Wipe Hard Disk operation, in case of a locked hard disk is selected.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

The detailed explanation of stepwise actions is given in following sections:

- Working with locked/system partitions and hard disks
- Backup system and locked partitions

4.21 Clear Free Space

This chapter explains how to clear unused space of formatted partitions on local drives with using Hard Disk Manager.

4.21.1 Overview

Clear Free Space function allows wiping unused space of a formatted partition by overwriting contents of unused clusters.

This function can be usable, if a user intends:

- surely exclude any traces of deleted files and directories, without destroying usable data placed on the selected partition;
- destroying non-standard protection/registration/deactivation hidden marks made by some software.

Usually, an operating system just removes references to deleted files and makes no actions to really erase their contents. Specially designed software allows retrieving information from deleted files and directories, fully or partially. Restored information can be analyzed, thieved and used against user's confidentiality.

When implementing *Clear Free Space* function, the program overwrites all unused clusters with non-interpretable information, so that it's impossible recovering previous contents.

The operation *Clear Free Space* is available only for *Primary* and *Logical* Partitions of known filesystem types. The program uses knowledge about filesystem structure to distinguish between used and unused clusters.

The operation is not allowed for the Extended Partition and blocks of free space. To wipe the Extended Partition or a block of free space, use the function Wipe Partition.

The program allows multi-pass wiping and post-operation checking of a wiped space for clearance. Hard Disk Manager supports only a simplified edition of wiping: the program chokes up disk space with a single character.

4.21.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a partition you want to clear free space

Select a primary or logical partition formatted to FAT16, FAT32, NTFS, Ext2, Ext3 or ReiserFS filesystem, in the Tree Layout panel, or in the UDP Layout panel, or in the List of Partitions. In any case, it will be highlighted in all three panels.

Step 2. Select the operation to perform

Variants:

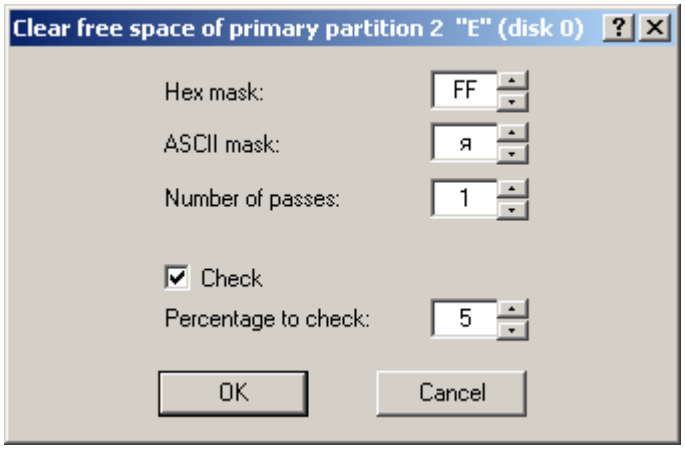
- Select in the main menu:

  Partition ➔ Clear free space…

- call the *popup menu* for the selected partition and select the item:

  Clear free space…

Step 3. Define parameters of the operation



You are able assigning following parameters:

- Amount of wiping passes
- A character that will be used for filling unused clusters
- Enable post-operation clearance check.

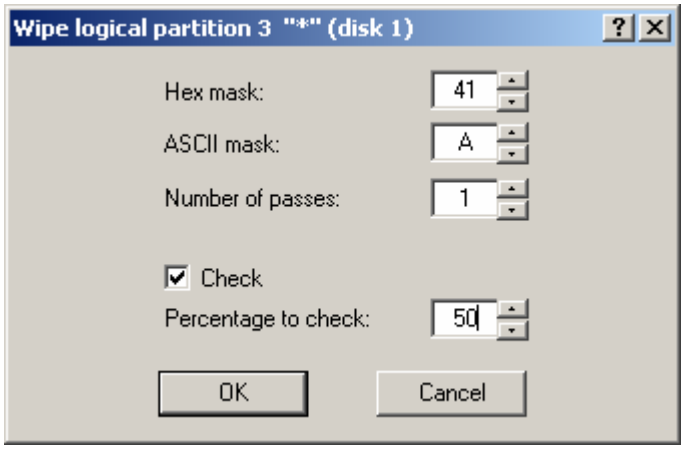
Initially, the program suggests some consistent values for all parameters. In most cases, you just need to press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Clear Free Space* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.21.3 Description of the Clear Free Space parameters



Hex mask

Hard Disk Manager wipes unused clusters by filling with the single 8-bit character.

The **Hex mask** spinner control allows defining the two-figure hexadecimal presentation of the character being used. The default value is "00" (that is corresponds to the hexadecimal value 0x00). The available range is from "00" to "FF".

This value is synchronized with the **ASCII mask** value.

ASCII mask

Hard Disk Manager wipes unused clusters by filling with the single 8-bit character.

The **ASCII mask** spinner control allows defining the symbolic presentation of the character being used. The default value is zero character (character #0).

This value is synchronized with the **Hex mask** value.

Number of passes

Set the amount of wiping passes here. The default value is 1, the available range is 1 to 100.

Check

Set this checkmark ON in order to perform post-operation check of wiped disk space for "clearance" – this feature just allows ensuring in wiping.

Percentage to check

The program allows checking for "clearance" all or just a part of wiped disk space.

Set percentage of disk space to be checked in this spinner control. The available range is from 0% to 100%, the default value is 0%.

Percentage	Action
100%	All free space is checked
0%	Nothing is checked
1-99%	The program checks random sectors within free space.

4.21.4 Running the Clear Free Space operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed

The program treats each wiping pass and the clearance-checking pass as suboperations. In case of multiple clearance passes is selected, the program displays suboperation statistics.

The operation is lengthy. Real performance of this operation fundamentally depends on data allocation, hardware and an operating system being used.

4.21.5 Comments

4.21.5.1 Hard Disk Manager does not destroy files in the Recycle Bin

Modern operating systems use two-level scheme of files deletion known as *Recycle Bin*: an operating system moves "just deleted" files to a special system folder, with keeping information about initial location. Files are kept in the Recycle Bin, until they will be *restored* by a user or finally deleted either by a user or by an operating system.

Recycle Bin folders have predefined filenames depending on the filesystem type:

Filesystem	Recycle Bin directory
FAT16/FA32	RECYCLED
NTFS	RECYCLER
Ext2/Ext3, ReiserFS	lost+found

Files, which are kept in the Recycle Bin, aren't truly deleted ones, they legally hold disk space on a partition. Hard Disk Manager does not wipe these files. To clear contents of files in the Recycle Bin, one should remove them from the Recycle Bin, and then perform the operation *Clear Free Space*.

4.21.6 Clearing free space on locked partitions

The difference between *locked* and *unlocked* partitions is that locked ones are used by other programs for file input-output activity (see Glossary).

The program cannot clear free space on locked partitions, because this operation may interfere disk input-output activity of an operating system or other applications.

Hard Disk Manager requires rebooting the system to a single-tasking environment in order to complete the Clear Free Space operation, in case of a locked partition is selected.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

The detailed explanation of stepwise actions is given in following sections:

- Working with locked/system partitions and hard disks
- Backup system and locked partitions

4.22 Resize & Move Partition

This chapter explains how to change size and location of existing partitions with keeping data intact, by using Hard Disk Manager.

Hereafter an operation of changing partition size is named *resizing* (resize).

4.22.1 Overview

The *Resize & Move Partition* function allows arbitrarily changing size and position of partitions on the disk. Combining resizing and moving of several partitions, it's possible solving tasks of :

- ⇨ Redistribution of disk space between existing partitions
- ⇨ Releasing disk space for creating new partitions
- ⇨ Adding extra space to existing system partition(s)

Resizing partitions

Common disk management tools including ones supplied with operating systems, provide only the destructive method of changing partition size: the old partition should be deleted, a new one having other capacity should be created and formatted. These actions irreversibly destroy all data previously located on a partition.

Instead, Hard Disk Manager can *resize* the partition, i.e. it allows changing partition capacity without destroying data. When performing this operation, the program analyzes the current allocation of files and evaluates the resulting allocation of files on the changed partition. Next, the program moves data to the new location and finally, corrects the system information about the partition properties.

Moving partitions

A partition can be re-located (i.e. *moved*) as a whole. Again, common disk management tools provide only the destructive method of changing partition's position: the old partition should be deleted, a new one having other location should be created and formatted. These actions irreversibly destroy all data previously located on a partition.

Hard Disk Manager can *move* the partition, without destroying data. When performing this operation, the program just moves usable sectors of a partition to the new location and updates system information about the partition properties.

Combined Resize & Move operation

This version of Hard Disk Manager integrates both *Resize* and *Move* operation into the single operation *Resize & Move Partition*, which is more convenient for users. The program uses the combined algorithm that allows freely changing both capacity and location of a partition.

The combined *Resize & Move* operation submits to both resizing and moving terms, which are described in the following text.

4.22.1.1 Supported filesystem types

Hard Disk Manager supports *Resizing* only for partitions of known filesystem types: FAT16, FAT32, NTFS, Ext2, Ext3, ReiserFS.

The program uses knowledge about internal structure of filesystems for correct changing its size. *Unknown* and *unformatted* partitions are not acceptable!

Instead, *Moving* is supported for partitions of any type. The program always uses the *sector-to-sector copying* algorithm for unknown and unformatted partitions (see the section Functionality Overview ➔ Fast copying algorithm). Known filesystems can be processed in both *sector-to-sector* and *fast* copying modes.

In the *fast copying* mode, the program copies only usable sectors. This feature significantly diminishes time required for the operation completion. In this case, the program accepts only valid partitions, which have no filesystem errors.

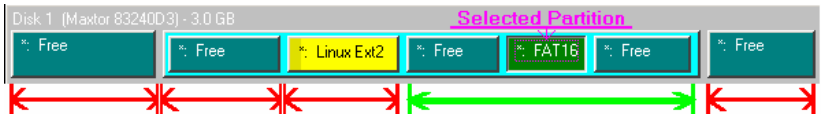
In the *sector-to-sector copying* mode, the program ignores filesystem information and simply copies all sectors. This method is slower than the fast copying; however, it allows moving any partitions.

4.22.1.2 Limitations on the new partition capacity and location

Hard Disk Manager allows resizing and moving a partition only within blocks of unpartitioned space that are contiguous with the selected partition. The program does not allow the modified partition neither overlapping nor "jumping over" other partitions.

In particular, the program does not allow making in one operation the expansion of the selected partition at the expense of free space on other partitions. This procedure consists of several steps, and is described in the section How to expand a partition at the expense of free space from another partition.

The picture placed below summarizes these constraints:



- the selected partition is FAT16 (green-colored).
- disk space, which is available for moving a partition, and which can be added to the partition during the resizing, is marked with green arrows.
- disk space, where the selected cannot be moved, and which cannot be added to the partition, is marked with red arrows.

In addition, this theme is discussed in the section How Hard Disk Manager evaluates space available for relocating and resizing of a partition.

4.22.2 Initiating the operation

The *Resize & Move Partition* operation can be initiated in several ways that differ in their user friendliness and the flexibility. According to your aims, you can choose either the fast method based on the *drag-&-drop* technique or the accurate method based on entering parameters in the *Resize & Move Partition* dialog.

4.22.2.1 Using the drag-&-drop technique for Resizing a Partition

The *drag-&-drop* technique is available only in case of enabling virtual execution of operations, in the *Smart* and *Virtual Execution* modes (see the chapter Virtual operations).

The *drag-&-drop* technique for the *Resize Partition* operation is supported only in the UDP Layout panel:

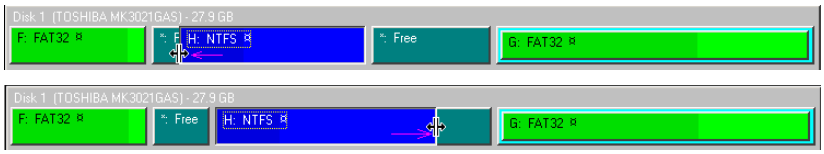
- select a partition you want to resize in the UDP control
- position the mouse cursor over its right or left edge
- press and hold the primary (=left) mouse button
- drag an edge of the partition to an adjacent block of free space
- and drop it to the desired position.

Remember, the program allows expanding a partition only to an adjacent block of free space (see Limitations of the Resize & Move operation).

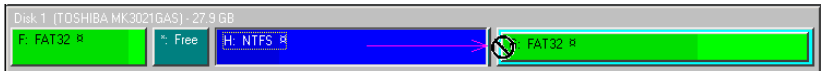
During the dragging a partition's edge, the program indicates the availability of the operation for the current position:

Initial layout of partitions:

The program is able expanding the partition:



The program cannot expand the partition:



The mouse cursor is dragged over another partition. Nevertheless, partitions cannot overlap.

See the section Comments for more details about *Resize & Move* rules.

4.22.2.2 Using the drag-&-drop technique for Moving a Partition

The *drag-&-drop* technique is available only in case of enabling virtual execution of operations, in the *Smart* and *Virtual Execution* modes (see the chapter Virtual operations).

The *drag-&-drop* technique for the *Move Partition* operation is supported in all layout panels (the Tree Layout panel, the UDP Layout panel and the List of Partitions):

- select a partition you want to move
- position the mouse cursor within a colored bar representing the partition
- press and hold the primary (=left) mouse button
- drag a partition to an adjacent block of free space
- and drop it to the desired position.

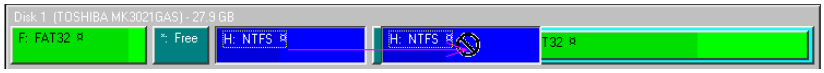
Remember, the program allows expanding a partition only to an adjacent block of free space (see Limitations of the Resize & Move operation).

During the dragging a partition, the program indicates the availability of the operation for the current position:

Initial layout of partitions:

The program is able moving the partition to this place:

The program cannot move the partition to this place:



The moved partition would overlap with another partition, but this is illegal case.

See the section Comments for more details about *Resize & Move* rules.

4.22.2.3 Using the Resize & Move Partition dialog

The *Resize & Move Partition* dialog provides the full functionality of the Resize&Move function: one can set accurately any admittable position for the partition and define precisely its final size.

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a partition you want to Move and/or Resize

Select a partition you want to move or resize, in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions. In any case, the partition will be highlighted in all three panels. The function is not available for blocks of free space.

Step 2. Select the operation to perform

Variants:

- Select in the main menu:

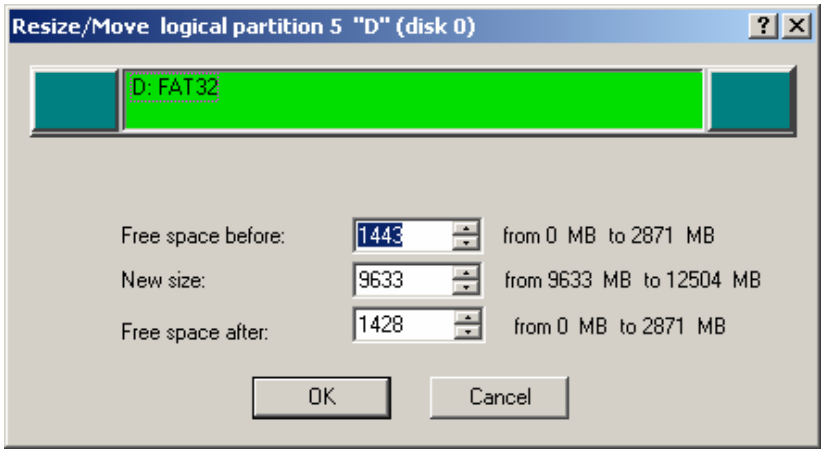
  Partition ➔ Resize/Move…

- Call the *popup menu* for the selected partition in any of layout panels (click right mouse button) then select the menu item:

  Resize/Move…

- Press **Alt+Z** keyboard combination

Step 3. Define parameters of the operation



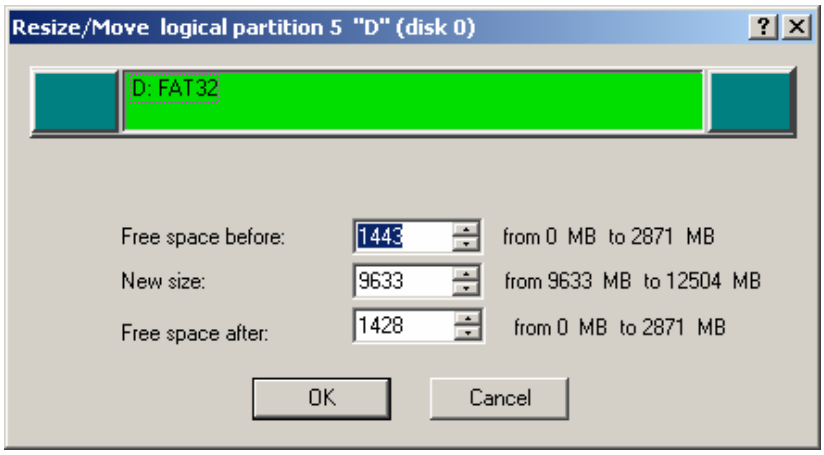
In the *Resize & Move Partition* dialog, one can define new location and size of the partition. The section Description of parameters contains the detailed description of the dialog's functionality.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Resize & Move Partition* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.22.3 Description of parameters



The *Resize/Move partition* dialog provides the ability to fully control the resulting position and size of the selected partition. The program determines the range of disk space available for placing the restored partition (see the section How Hard Disk Manager evaluates space available for relocating and resizing of a partition). Briefly, Hard Disk Manager allows re-locating and re-sizing the partition within the range of disk space that includes the partition and the enveloping blocks of free space.

One can move the partition and change its size either by using the *UDP control* that is located in the window or by using three *spinner* controls placed on the bottom:

Free Space Before

The **Free space before** spinner control defines the position (in Mb) of the partition relative to the beginning of the available range of disk space.

New Size

The **New size** spinner control defines the size (in Mb) of the partition.

Free Space After

The **Free space after** spinner control defines the amount of trailing free space (in Mb) at the end of the available range of disk space.

The *restrictions* the program takes into account are those:

- The partition cannot be enlarged beyond the range of available disk space.
- The partition cannot be reduced less than the amount of used space.

The UDP control and spin controls are synchronized, the changing of any of these elements affects on all other ones.

How the spin controls behave:

Free space before	Moves the beginning of the partition (left edge), preferably with keeping the partition size.
New Size	Changes the size of the partition, preferably with keeping the starting position (left edge).
Free space before	Moves the end of the partition (right edge). On increasing the value, it (preferably) keeps the partition size. On decreasing the value, it (preferably) keeps the starting position (left edge) so that the partition expands.

There are rules that take effect in the partition resizing. See the section Comments for more details.

4.22.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed
- Amount of moved and remaining data
- Information about actions being performed.

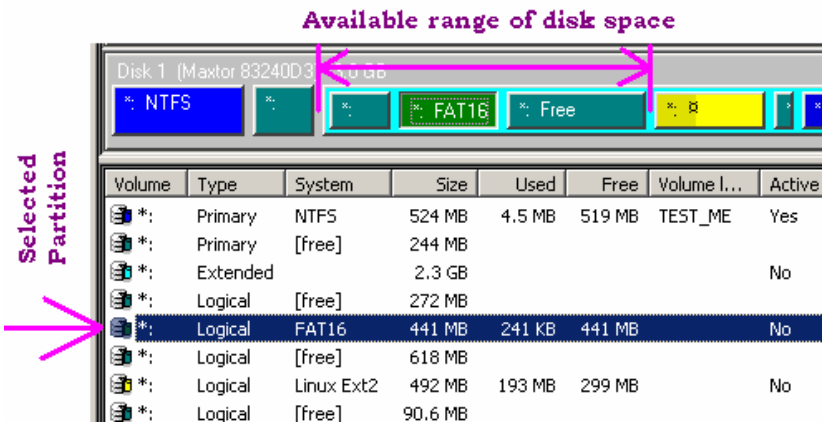
Resize&Move is lengthy operation, and usually requires much memory. Its real performance fundamentally depends on amount of files, branching and nesting of directories, hardware and an operating system being used.

4.22.5 Comments

4.22.5.1 How Hard Disk Manager evaluates space available for relocating and resizing of a partition

Hard Disk Manager uses following rules for determining the available range of disk space and resizing and moving partitions:

1. The program includes to the available range:
   - ⇨ the selected partition
   - ⇨ the left adjacent block of free space (if exists)
   - ⇨ and the right adjacent block of free space (if exists).



2. The moved (resized) partition must completely have room within the available range of disk space. The program does not allow expand a partition over other partitions.
3. In case of miniaturizing a partition, the resulting partition size cannot be less than amount of used space stored on this partition.

4.22.5.2 How to avoid bad sectors during the resizing/moving

In case of moving data to bad sectors, the resulting partition may become corrupted. To avoid loosing data due to bad sectors, select the *surface test* to the value other than **None**.

(menu) General ➔ Settings… ➔ (page) General ➔ Surface test

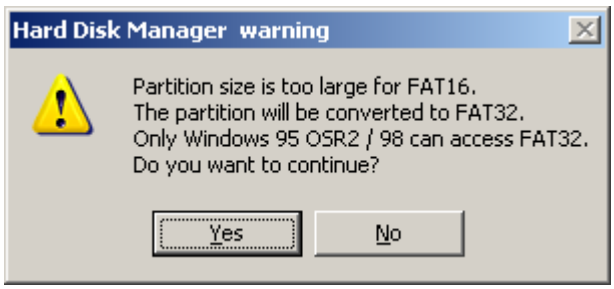
In this case, the program performs the surface test of disk space added to the partition, prior the data moving. In case of detecting bad or unreliable sectors, the program will mark them *bad* and will not place data to those sectors.

The additional surface test can noticeably increase the overall time required to complete the operation, and the algorithm of bypassing bad sectors slows down the performance. Use this feature only in case of you suspect that there are bad sectors in the new location of the partition.

4.22.5.3 Conversion of FAT16 to FAT32 during the resizing

FAT16 partitions are limited to the value of 2Gb. However, Hard Disk Manager allows expanding FAT16 partition over the 2Gb boundary by converting FAT16 filesystem to FAT32.

In this case, the program displays the following warning message:



The program allows inhibiting this warning (see the chapter Settings overview, section General Page):

(menu) General ➔ Settings… ➔ (page) General ➔ (checkmark) Convert FAT16 to FAT32 automatically

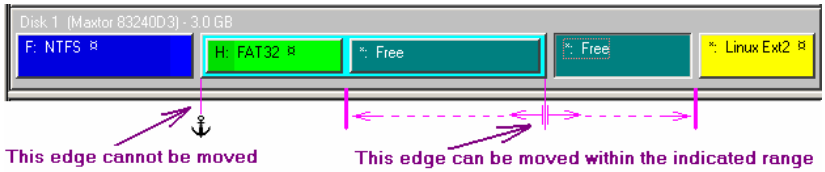
In this case, the program automatically converts FAT16 partitions to FAT32 without warnings.

4.22.5.4 Resizing & moving the Extended Partition

Hard Disk Manager provides the limited functionality in changing size and location of the Extended Partition:

- The Extended Partition cannot be *moved* as a whole.
- When *resizing* the Extended Partition, only its starting and ending edges can change positions. Logical and primary partitions stay intact.

In fact, edges can move within adjacent blocks of free space. The picture below illustrates these rules:



4.22.5.5 How to expand a partition at the expense of free space from another partition.

According to the Hard Disk Manager strategy, partition-related operations should modify only one partition at once.

In fact, the procedure of redistributing space between partitions affects two or more partitions, so that it should be executed in multiple steps. Each of these steps is resizing or moving of some partition.

Resize & Move is noticeably lengthy operation. For this reason, it is very convenient activating the Virtual Execution of operations (see the section Settings Overview ➔ Virtual operations Page). In the Virtual Execution mode, one can quickly "construct" new disk layout in the program's GUI, and then leave the computer while the program completes all pending operations (see the chapter Virtual operations to learn about the delayed execution of virtual operations).

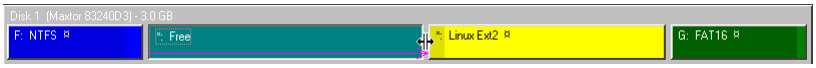
Example 1: Transferring space between neighboring partitions

Initial state: there are three partitions on the disk, the NTFS partition has insufficient free space, and Ext2 partition is almost free:

Our aim is to "transfer" some free space from Ext2 to NTFS partition.

Step 1. Diminish Ext2 partition

At first, the partition having plenty of free space, should be diminished. In our example, the starting edge of the Ext2 partition must be moved away from the NTFS partition:



The released disk space is adjacent to the NTFS partition, so that it can be added to that partition.

Note: although the program supports visual drag-&-drop technique for the *Resize & Move Partition* operation, the using of *Resize&Move dialogs* can be more convenient, because they allow accurate defining amount of released space and new location of a partition.

Step 2. Expand NTFS partition

Next, the NTFS partition should be expanded to the adjacent block of free space, by moving its ending edge towards the Ext2 partition:

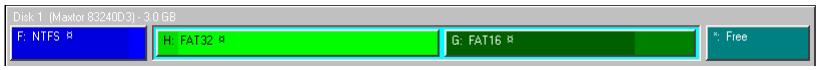
Step 3. Applying pending operations

Finally, one should complete the physical execution of pending operations by pressing the **Apply** button on the *Virtual Operations Toolbar* (see the chapter Interface Layout ➔ Virtual Operations Toolbar).

Example 2: Gathering free space from several Logical Partitions and adding it to a Primary Partition.

This example is just a little bit more complicated. In addition to the **Example 1**, it demonstrates technique of "transferring" disk space through boundaries of the Extended Partition.

Initially, the 1st Primary Partition (NTFS, drive F:) has insufficient amount of free space, while 1st Logical Partition (FAT32, drive H:) and 2nd Logical Partition (FAT16, drive G:) have plenty of free space. In addition, there is a block of free space behind the Extended Partition:



Our aim is to pick up all unpartitioned free space and most of unused space (e.g. 50-80%) from all logical partitions, and add it to the Primary Partition.

Potentially, this task can be completed in several ways. This is the optimal one: an outer block of free space must be included into and then "walked through" the Extended Partition (by moving logical partitions). During moving logical partitions, we can additionally shrink them in order to gather unused disk space into a single free block. On the opposite side of the Extended Partition, the free block will be extracted outside.

Step 1. Include the last block of free space to the Extended Partition

Expand the ending edge of the Extended Partition to the free block in order to include free space to the Extended Partition:

Now the block of free space can be re-positioned within the Extended Partition.

Step 2. Diminish and move the last logical partition

The last logical partition should be reduced and then moved to the end of the Extended Partition:

Now the free block becomes larger, and it is able taking unused space from the logical partition situated ahead.

The using of *Resize/Move* dialog provides more optimal sequence of operations:

- With using the Resize/Move dialog, the step 1 can be realized within one Resize&Move operation. Within the dialog, only the parameter **Free Space Before** needs to be changed.
- With using *drag-&-drop technique*, two isolated operations are required – *Resize Partition* and then *Move Partition*, separately.

Step 3. Diminish and move next logical partition …

Similarly to the previous step, each logical partition should be moved towards the end of the Extended Partition:

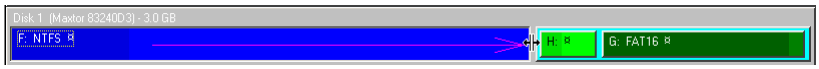
Step 4. Exclude the block of free space from the Extended Partition

Move the starting edge of the Extended Partition towards logical partitions in order to extract the free block outside the Extended Partition:

Now the free block is located just behind the primary partition and can be joined with it.

Step 5. Expand the Primary Partition

Now the Primary Partition is able taking up free space:



Step 6. Applying pending operations

Finally, one should complete the physical execution of pending operations by pressing the **Apply** button on the *Virtual Operations Toolbar* (see the chapter Interface Layout ➔ Virtual Operations Toolbar).

4.22.5.6 About interruption of the Resize/Move operation

Both *resizing* and *moving* actions temporarily distort the original filesystem information in order to adapt data to the new partition layout. A breaking of the *Resize & Move* operation usually makes a processed partition damaged and unusable.

However, it is possible avoiding data loss in case of executing Resize&Move operation in the *fail-safe mode*, which can be activated in program's Settings (see the section Settings overview ➔ General Page ➔ Data loss protection). In this mode, the program is able continuing operations, which were interrupted by hardware malfunction, power break or operating system failure.

A common procedure of resuming operations in the fail-safe mode is described in the section Executing operations in the fail-safe mode.

In case of an operation was interrupted by a user, the program will not provide an ability of resuming the operation, even if it works in the fail-safe mode. Remember, an interruption of the *Resize&Move* operation usually leads to the irrevocable corruption of a processed partition.

4.22.5.7 Memory requirements

The *Resize&Move* operation require large amount of memory for the execution, primarily for building up the map of data that should be moved to another location.

Here are the evaluation formulas:

Filesystem	Memory requirements
FAT16/FAT32, Ext2/Ext3	12 bytes per used cluster (maximum between the original and resulting filesystem states)
NTFS	0.5 bytes per used cluster (maximum between the original and resulting filesystem states)

4.22.6 Resizing & moving locked and system partitions

The difference between *locked* and *unlocked* partitions is that locked ones are used by other programs for file input-output activity (see Glossary).

Hard Disk Manager requires the exclusive controlling of a partition being moved or resized. The *Resize & Move Partition* operation is incompatible with any other data activity, because the program needs temporarily distort filesystem data in order to successfully complete the operation.

Hard Disk Manager requires rebooting the system to a single-tasking environment in order to complete the *Resize & Move Partition* operation, in case of a locked partition is selected.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

The detailed explanation of stepwise actions is given in following sections:

- Working with locked/system partitions and hard disks
- Backup system and locked partitions

4.23 Convert Filesystem

This chapter explains how to change the filesystem type of existing partitions with keeping data intact, by using Hard Disk Manager.

Hereafter, the operation of non-destructive changing filesystem type is named *filesystem conversion*.

4.23.1 Overview

It is known that various operating systems have different stacks of supported filesystems, and filesystems vary significantly in provided features, such as filename standards, security standards and many other properties. In addition, filesystems differ in performance of input-output operations and in maintenance costs.

A user may have a need of changing the filesystem type on a partition in following cases:

- Getting advanced filesystem features (security, compression, encryption etc).
- Getting better performance of file managing operations.
- Getting better fault tolerance
- Allowing an access to an on-partition data from multiple operating systems.

Common disk management tools and operating systems usually provide only a destructive method of changing filesystem type by re-formatting a partition to another filesystem. Unfortunately, this action destroys all data located on a partition (the only exclusion is the **CONVERT** utility provided with Windows NT, 2000 and XP; its performance is discussed in the section The CONVERT utility from Windows NT, 2000 and XP).

Hard Disk Manager provides the ability of the filesystem *conversion*, i.e. the program can change the filesystem type without destroying data.

4.23.1.1 Supported filesystem types

Hard Disk Manager allows following filesystem conversions:

Original filesystem	Resulting filesystem	Comments
FAT16	NTFS	Allows changing the *Cluster Size* to a smaller value
	FAT32	
FAT32	NTFS	
	FAT16	A partition should meet FAT16 filesystem requirements (size<2Gb, amount of files/clusters<65530)
NTFS	FAT16	
	FAT32	Files should be less than 4Gb in size. Keeps the original *Cluster Size* value
Ext2	Ext3	Adds the *journal* metadata

When performing the *Convert Filesystem* operation, the program first checks the current filesystem consistency and verifies that on-partition data satisfies the requirements of the resulting filesystem. If the selected partition passed these tests, the program re-organizes *filesystem metadata* and user files (see the chapter Glossary).

4.23.1.2 Primary Conversion constraints

The primary constraint of the filesystem conversion is that data placed on the selected partition should meet requirements of the chosen resulting filesystem. These requirements are briefly listed in the table:

Conversion	Conditions
FAT16 ➔ NTFS **FAT32 ➔ NTFS** **FAT16 ➔ FAT32**	Available for all partitions having consistent filesystem.
NTFS ➔ FAT32	Available for following consistent partitions: • capacity < 8000 Gb • amount of files and folders < 256 million • each file is less than 4Gb • each folder contains less than 65530 files and subfolders.
NTFS ➔ FAT16 **FAT32 ➔ FAT16**	Available for following consistent partitions: • capacity < 2 Gb • amount of files and folders < 65530 • amount of clusters < 65530

Ext2 ➔ Ext3	Available for all partitions having consistent filesystem.

4.23.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a partition you want to change the filesystem type

Select a primary or logical partition formatted to FAT16, FAT32 or NTFS filesystem, in the Tree Layout panel, or in the UDP Layout panel, or in the List of Partitions. In any case, it will be highlighted in all three panels.

Step 2. Select the operation to perform

Variants:

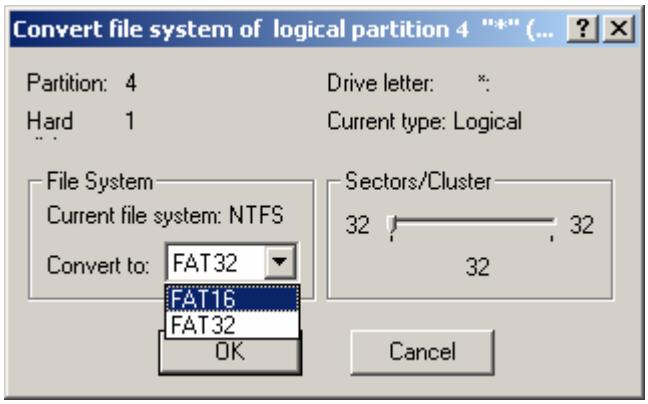
- Select in the main menu:

  Partition ➔ Modify ➔ Convert…

- call the *popup menu* for the selected partition and select the item:

  Modify ➔ Convert…

Step 3. Define parameters of the operation



You are able assigning following parameters:

- Selecting a resulting filesystem type (see the table in the section Overview).
- Choosing new Cluster Size (within a limited range of values).

Initially, the program suggests converting FAT to NTFS, or NTFS to FAT32, without changing the Cluster Size value.

After choosing desired operation parameters, one should press **OK** button to confirm the operation.

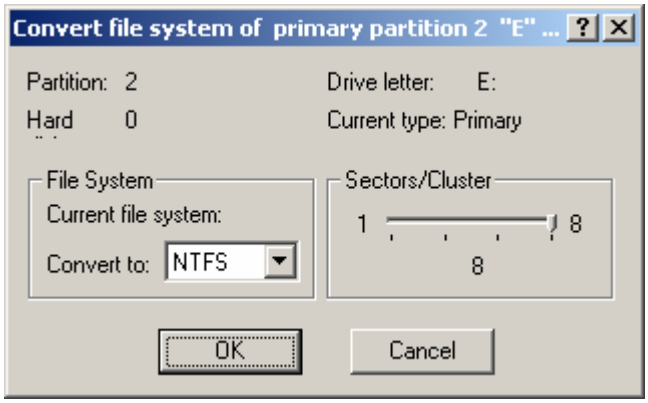
Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Convert Filesystem* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.23.3 Description of parameters

4.23.3.1 "Convert file system" dialog



The dialog allows observing current properties of the selected partition and choosing a new filesystem type. When converting FAT16/FAT32 to another filesystem, the program also allows changing the *Cluster Size* value.

Convert to:

The pull-down list contains filesystems the selected partition can be converted. The program takes into account current parameters of the selected partition and *filesystem limitations*. Unavailable conversion variants are not displayed. See section Comments and Format Partition ➔ Comments for more details.

Sectors/Cluster

This *tracker* control allows additionally changing the *Cluster Size* property of the converted partition. This feature has some limitations:

- This function extension is available only for cases of **FAT16➔NTFS** and **FAT32➔NTFS** conversion.
- The Cluster Size can only be reduced in this function extension.

The operation Change Cluster Size allows fully control the Cluster Size parameter and changes it to any valid value.

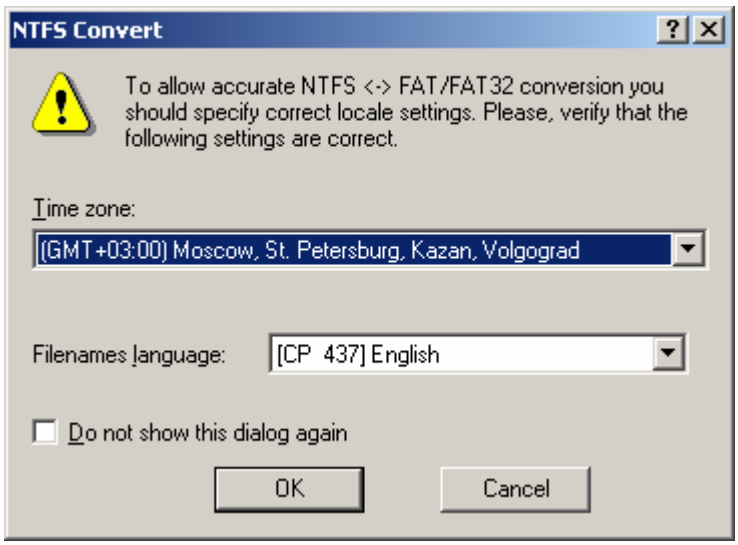
In addition, following fields are used for displaying partition properties:

Partition	Displays the number of the partition on the hard disk
Hard disk	Displays the hard disk number that contains the selected partition
Drive letter	Displays drive letter that is assigned to the selected partition. Non-mounted partitions contain "**:**" in this field
Current type	Displays a partition type (Primary or Logical)
Current file system	Displays the current filesystem placed on the selected partition

4.23.3.2 "Confirm locale settings" dialog

This dialog can appear before the real execution of the *Convert Filesystem* operation. It provides the ability of customizing locale (regional settings), which should be used by the program during the filesystem conversion (see the section Comments for more details).

Default locale settings are defined in the program's Settings (see the section Settings overview ➔ Locale Info Page). This dialog allows using custom settings in a particular case.



Time zone

This pull-down list allows selecting the *time zone* parameter.

This value is used for valid correction of *file timestamps* in case of conversion NTFS➔FATxx or FATxx➔NTFS (see the section Comments for more details).

Filename language

This pull-down list allows selecting the *code page* parameter.

This value is used for the valid translation filenames from *Unicode* to *ANSI* (or *OEM*) coding and vice versa, in case of conversion NTFS➔FATxx or FATxx➔NTFS (see the section Comments for more details).

Do not show this dialog again

Set this checkmark in order to inhibit appearance of the "Confirm locale settings" dialog. In this case, the program will use default locale settings (see the section Settings overview ➔ Locale Info Page).

This option duplicates functionally the program's setting "Convert FAT16 to FAT32 automatically" (see the section Settings overview ➔ General page ➔ Convert FAT16 to FAT32 automatically).

4.23.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed
- Amount of moved and remaining data
- Information about actions being performed.

Convert Filesystem requires much memory, in some situations it can take noticeable time for completion. Its real performance fundamentally depends on amount of files, branching and nesting of directories, hardware and an operating system being used.

4.23.5 Comments

4.23.5.1 Using locale settings in filesystem conversion

Regional settings (or locale settings) are important in converting NTFS to FAT and FAT to NTFS. The thing is that FAT family filesystems (FAT12, FAT16, FAT32) and NTFS use different standards in saving file names and file timestamps.

File timestamps

FAT and NTFS keep *Creation time*, *Last Modification time* and *Last Accessed time* timestamps for every file and directory.

In the FAT filesystems, timestamps, which are saved in servicing records referencing to a file, just duplicate the appropriate values of local time.

In the NTFS filesystem, timestamps, which are saved in servicing records referencing to a file, are kept in GMT (Greenwich Mean Time). An operating system calculates values, which should be saved in the file timestamps, by using *time zone* value from Regional Settings.

Hard Disk Manager allows using not only the current time zone from Regional Settings, but arbitrary selecting this value.

In case of the time zone value is incorrectly assigned, all on-partition files and directories on the converted partition become "younger" or "older" to the fixed time value, which can vary within ±24 hours.

Filename Language

FAT and NTFS keep two filenames for every file and directory. The first one is named *long filename* (LFN), and the second one is named *short filename* (or *DOS alias*).

The LFN can be up to 254 characters long, can include characters in multiple languages. In both FAT and NTFS, long filenames are kept in the *Unicode* character encoding standard.

The short filename must meet the so-called *8.3 filename standard*, which comes from old MS-DOS systems. Old software do not support long filenames, it use short filenames to access files.

Unfortunately, FAT keeps short filenames in *ANSI encoding standard*, while NTFS use *Unicode* for this purpose.

Hard Disk Manager uses information about filename language for the correct conversion of non-Unicode filenames into Unicode filenames or vise versa.

The incorrect assigning of filename language may lead to problems in accessing to files with non-English filenames on partitions converted from NTFS to FAT16/FAT32.

4.23.5.2 The CONVERT utility from Windows NT, 2000 and XP

Windows NT, 2000 and XP provide a command-line CONVERT utility that allows converting FAT16 and FAT32 partitions to NTFS without destroying on-partition data.

Unfortunately, this utility has some disadvantages:

- It does not allow a backward conversion from NTFS to FAT.
- It does not perform a filesystem integrity check at the conversion, so that it can corrupt a partition having minor filesystem errors.
- It unconditionally changes the *Cluster Size* value to the minimum value of 1 Sector per Cluster, which noticeably slows down the performance of the disk subsystem.

4.23.5.3 Memory requirements

The *Convert Filesystem* operation requires large amount of memory for the execution, primarily for building up the map of data that should be moved to another location.

Here are the evaluation formulas:

Filesystem	Memory requirements
FAT16 <=> FAT32	12 bytes per used cluster (maximum between the original and resulting filesystem states)
NTFS <=> FAT	4.5 bytes per used cluster (maximum between the original and resulting filesystem states)

4.23.5.4 About interruption of the Convert Filesystem operation

The *Convert Filesystem* operation temporarily distorts the original filesystem information in order to adapt data to the new partition layout. A breaking of the *Convert Filesystem* operation usually makes a processed partition damaged and unusable.

However, it is possible avoiding data loss in case of executing Convert Filesystem operation in the *fail-safe mode*, which can be activated in program's Settings (see the section Settings overview ➔ General Page ➔ Data loss protection). In this mode, the program is able continuing operations, which were interrupted by hardware malfunction, power break or operating system failure.

A common procedure of resuming operations in the fail-safe mode is described in the section Executing operations in the fail-safe mode.

In case of an operation was interrupted by a user, the program will not provide an ability of resuming the operation, even if it works in the fail-safe mode. Remember, an interruption of the *Convert Filesystem* operation usually leads to the irrevocable corruption of a processed partition.

4.23.6 Converting Filesystem of locked partitions

The difference between *locked* and *unlocked* partitions is that locked ones are used by other programs for file input-output activity (see Glossary).

Hard Disk Manager requires the exclusive controlling of a partition when changing filesystem type. The *Convert Filesystem* operation is incompatible with any other data activity, because the program needs temporarily distort filesystem data in order to successfully complete the operation.

Hard Disk Manager requires rebooting the system to a single-tasking environment in order to complete the *Convert Filesystem* operation, in case of a locked partition is selected.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

The detailed explanation of stepwise actions is given in following sections:

- Working with locked/system partitions and hard disks
- Backup system and locked partitions

4.24 Change Cluster Size

This chapter explains how to change the *Cluster Size* parameter without re-formatting a partition, with using Hard Disk Manager.

4.24.1 Overview

Cluster Size is a one of important parameters of a filesystem (see the chapter Glossary for more details).

The Cluster Size parameter affects implicitly on the performance of file input-output activity, because it defines the size of filesystem metadata. In addition, it is responsible for effectiveness of disk space usage: the so-called *waste space* factor depends on the Cluster Size value.

Briefly, the Cluster Size affects on the disk subsystem properties in the following fashion:

- The diminishing of the Cluster Size reduces the *waste space* factor, but slows down the file activity performance.
- The increasing of the Cluster Size accelerates the file activity performance and reduces maintenance requirements, but reduces the effectiveness of disk space usage.

A user may have a need of changing the Cluster Size value on a workable partition for following reasons:

- To improve the file activity performance (in this case, the Cluster Size should be increased).
- To improve the effectiveness of disk space usage (in this case, the Cluster Size should be decreased).
- To get an ability of using advanced features of the NTFS filesystem such as compression and encryption. The thing is that these features are available only for NTFS partitions having Cluster Size of 4KB or less.

Common disk management tools and operating systems provide only a destructive method of changing the Cluster Size value by re-formatting a partition with another filesystem parameters. Unfortunately, this action destroys all data located on a partition.

Instead, Hard Disk Manager provides the ability of non-destructive changing the Cluster Size to any legal value.

4.24.1.1 Supported filesystem types

The *Change Cluster Size* functionality is provided for following filesystems:

Filesystem	Cluster size (in Sectors per Cluster)	Comments
FAT16	1* to 64 (0.5K to 32K)	* The minimum value essentially depend on the actual capacity of the partition
FAT32	1 to 64 (0.5K to 32K)	
NTFS	1 to 128** (0.5K to 64K)	** Windows cannot boot from NTFS partition having 64K Cluster Size.

When performing the *Change Cluster Size* operation, the program first checks the current filesystem consistency. If the selected partition passes the consistency test, the program reorganizes on-partition data.

4.24.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a partition you want to change the cluster size parameter

Select a primary or logical partition formatted to FAT16, FAT32, NTFS, Ext2, Ext3 or ReiserFS filesystem, in the Tree Layout panel, or in the UDP Layout panel, or in the List of Partitions. In any case, it will be highlighted in all three panels.

Step 2. Select the operation to perform

Variants:

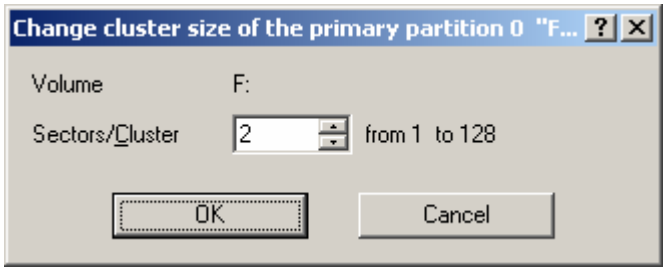
- Select in the main menu:

  Partition ➔ Modify ➔ Change cluster size…

- call the *popup menu* for the selected partition and select the item:

  Modify ➔ Change cluster size…

Step 3. Define parameters of the operation



You are able assigning new value for the *Cluster Size* property.

Hard Disk Manager takes into account current parameters of the selected partition and filesystem limitations in order to determine the available range of the Cluster Size value.

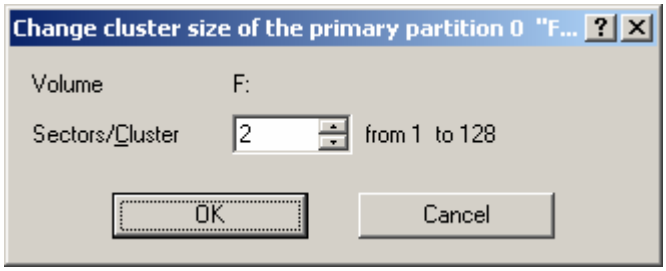
Initially, the program displays the actual value of the *Cluster Size* property. After choosing desired values, one can press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Change Cluster Size* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.24.3 Description of parameters



The dialog displays the selected value of the Cluster Size parameter and the available range for this value.

Sectors/Cluster

This *spinner* control contains the selected value of the *Cluster Size* file system's parameter. The text on the right side displays the available range of values.

The *Cluster Size* value is expressed in Sectors Per Cluster. To obtain the Cluster Size in Kbytes, halve this one.

Volume

This text contains a drive letter that is assigned to the selected partition. It is used for notification purposes only. In case of no drive letter is assigned, the text "***:**" is displayed.

4.24.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed
- Amount of moved and remaining data
- Information about actions being performed.

Change Cluster Size requires much memory. In case of increasing the Cluster Size value, the operation usually takes much time for completion. Its real performance fundamentally depends on amount of data, branching and nesting of directories, hardware and an operating system being used.

4.24.5 Comments

4.24.5.1 Memory requirements

The *Change Cluster size* operation requires large amount of memory for the execution, primarily for building up the map of data that should be moved to another location.

Here are the evaluation formulas:

Filesystem		Memory requirements
FAT16, FAT32		12 bytes per used cluster (maximum between the original and resulting filesystem states)
NTFS	Increasing	= (<**clus_amo**> * 3/8) bytes + 4MB where **clus_amo** is the original amount of clusters
	Decreasing	= (<**clus_amo**> /8 (1+ 2 * **new** / **old**)) bytes + 4MB where **clus_amo** is the original amount of clusters **new** and **old** are new and old *Cluster Size* values, respectively

The *Change Cluster Size* operation temporarily distorts the original filesystem information in order to adapt data to the new partition layout. A breaking of the operation usually makes a processed partition damaged and unusable.

4.24.5.2 About interruption of the Change Cluster Size operation

The *Change Cluster Size* operation temporarily distorts the original filesystem information in order to adapt data to the new partition layout. A breaking of the *Change Cluster Size* operation usually makes a processed partition damaged and unusable.

However, it is possible avoiding data loss in case of executing Change Cluster Size operation in the *fail-safe mode*, which can be activated in program's Settings (see the section Settings overview ➔ General Page ➔ Data loss protection). In this mode, the program is able continuing operations, which were interrupted by hardware malfunction, power break or operating system failure.

A common procedure of resuming operations in the fail-safe mode is described in the section Executing operations in the fail-safe mode.

In case of an operation was interrupted by a user, the program will not provide an ability of resuming the operation, even if it works in the fail-safe mode. Remember, an interruption of the *Change Cluster Size* operation usually leads to the irrevocable corruption of a processed partition.

4.24.6 Changing Cluster Size on locked and system partitions

The difference between *locked* and *unlocked* partitions is that locked ones are used by other programs for file input-output activity (see Glossary).

Hard Disk Manager requires the exclusive controlling of a partition when changing the Cluster Size value. The *Change Cluster Size* operation is incompatible with any other data activity, because the program needs temporarily distort filesystem data in order to successfully complete the operation.

Hard Disk Manager requires rebooting the system to a single-tasking environment in order to complete the *Change Cluster Size* operation, in case of a locked partition is selected.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

The detailed explanation of stepwise actions is given in following sections:

- Working with locked/system partitions and hard disks
- Backup system and locked partitions

4.25 Change Primary Slot

This chapter explains how to change the enumeration of Primary Partitions in the Partition Table with using Hard Disk Manager.

The operation of changing the enumeration of Primary Partitions is named *Change Primary Slot*.

4.25.1 Overview

The enumeration of partitions according to the order of appropriate records in the Partition Table is used in the *DOS partitioning scheme* (see the chapter Glossary for more details). Operating systems use this kind of partitions enumeration in the following manner:

Using the enumeration of partitions in Linux

In Linux, every partition has a special symbolic name that encodes a hard disk containing a partition, and a partition itself. Partitions are addressed and accessed by using their symbolic names.

Symbolic names are automatically generated by Linux, in accordance with the order of hard disks in BIOS and the order of partition records in the Partition Table.

The changing of primary partitions enumeration can lead to the changing of paths to some important resources.

Using the enumeration of partitions in Windows NT, 2000 and XP

Windows NT, 2000 and XP distinguish partitions by their starting position on a hard disk and usually ignore partitions enumeration. However, there some cases when Windows rely on the partitions enumeration:

1. the NTLDR (NT loading module) distinguishes partitions by their enumeration. It reads the number of the Windows system partition from the configuration file BOOT.INI.
   The changing of primary partitions enumeration can lead to inability Windows NT/2000/XP to boot.
2. Windows NT/2000/XP uses the enumeration of partitions when automatically assigning drive letters to partitions on a hard disk that is connected to the system at most first time.

The changing of primary partitions enumeration affects on drive letters assignment in this case.

Using the enumeration of partitions in DOS and Windows 95, 98 and ME

Windows 95, 98 and ME and last versions of MS-DOS use the rather sophisticated algorithm of drive letters assignment, which is briefly explained in the section Mount Partition ➔ Comments ➔ Manipulating drive letters in Windows 95, 98 and ME. A drive letter assigned to a partition depends on the order of records in the Partition Table.

The changing of primary partitions enumeration affects on drive letters assignment. In early versions of MS-DOS, it can even lead to the unavailability of a partition.

Well, a user may have a need of changing the enumeration of primary partitions.

Common disk management tools do not provide an ability of changing partitions enumeration. Moreover, various utilities work in different manner, so that there is a possibility of messing partitions enumeration by seemingly inoffensive actions.

Hard Disk Manager provides the ability changing the enumeration of primary partitions. This feature allows fixing some problems concerning inappropriate order of partitions. In particular, the program can restore the ability of starting from Windows bootable partition, in case of mismatching settings of boot configurations in the BOOT.INI file.

4.25.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a hard disk, where the enumeration of partitions should be changed

Select a hard disk in the Tree Layout panel or in the UDP Layout panel. The selected disk will be highlighted in both panels.

Step 2. Select the operation to perform

Variants:

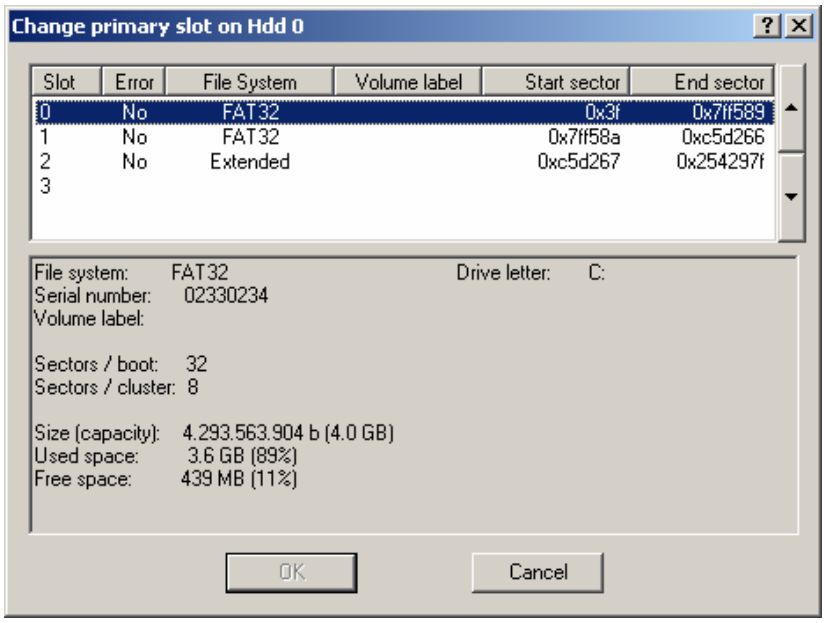
- Select in the main menu:

  Hard disk ➔ Change primary slot…

- call the *popup menu* for the selected hard disk and select the item:

  Change primary slot…

Step 3. Change the order of records in the Partition Table



A user is able changing the enumeration of Primary Partitions and the Extended Partition.

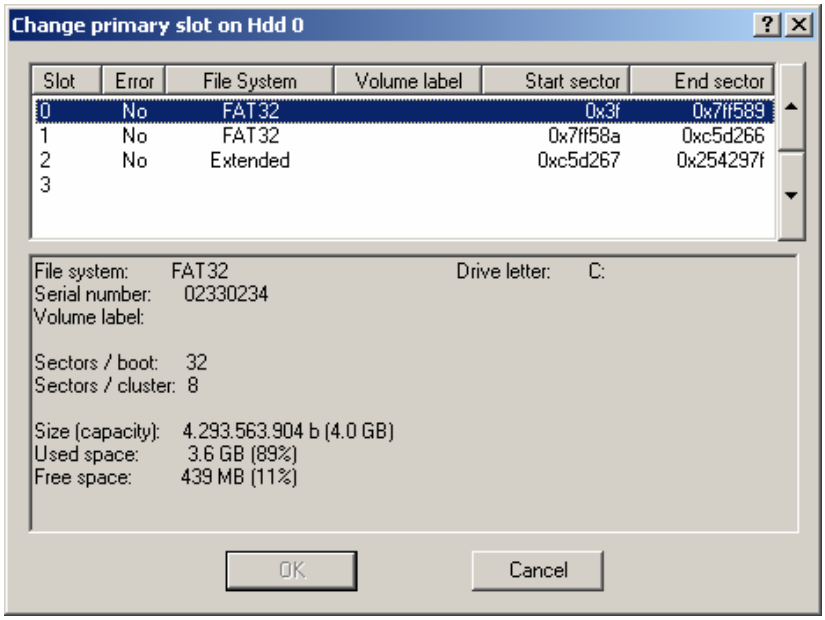
Initially, the program displays the actual state of the Partition Table. After making desired changes, one can press **OK** button to confirm the operation.

Step 4. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Change Primary Slot* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.25.3 Description of the dialog functionality



4.25.3.1 Layout

The dialog displays the actual enumeration of *Primary Partitions* in the Partition Table (exactly, it exhibits the order of appropriate records, which refer to primary partitions, in the primary part of the *Partition Table* referencing records. See the chapter Glossary for more details about the structure of the *DOS partitioning scheme*).

The top part of the dialog displays the enumeration order of partitions, with some parameters that can help in distinguishing partitions:

Slot	The current position of the record, which refers to the selected partition, in the Partition Table
Error	No – the selected partition has parameters consistent with the hard disk geometry and parameters of other partitions Yes – the selected partition
Filesystem	The filesystem that is placed on the selected partition
Volume label	The Volume Label of the selected partition
Start Sector	The starting position of the partition on the hard disk (expressed in Sectors, in the hexadecimal form)
End Sector	The ending position of the partition on the hard disk (expressed in Sectors, in the hexadecimal form)

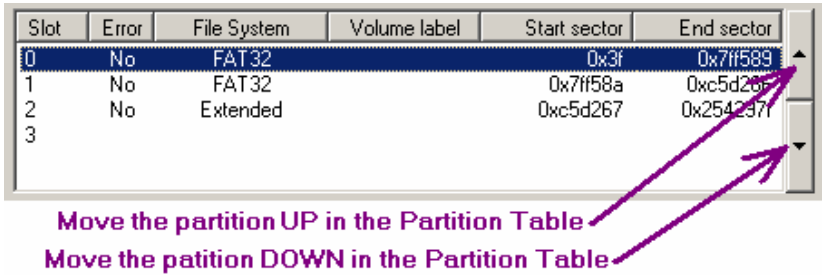
The bottom part displays additional information about a selected partition:

- ⇨ Filesystem type
- ⇨ Serial Number
- ⇨ Volume Label
- ⇨ Assigned Drive Letter
- ⇨ Capacity
- ⇨ Used Space (value and percentage to the Capacity)
- ⇨ Free Space (value and percentage to the Capacity)

- ⇨ Cluster Size (Sectors per Cluster)
- ⇨ Size of Boot Area (Sectors per Boot)

4.25.3.2 Usage

There are two buttons on the right of the list of primary partitions, which allow moving a selected partition up and down within the primary part of the Partition Table:



- Select a partition, which needs changing its number.
- Click on the "Up" button to move the selected partition one position up. If the upper position is occupied with other partition, these records trade places.
- Click on the "Down" button to move the selected partition one position down. If the lower position is occupied with other partition, these records trade places.

The program allows re-positioning a partition's record to any of four available positions.

4.25.4 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time

The *Change Primary Slot* operation itself takes just a fraction of a second. Still, the program should waiting while an operating system accommodates changes in the Partition Table. This action may take 5-20 seconds in Windows 2000 and XP.

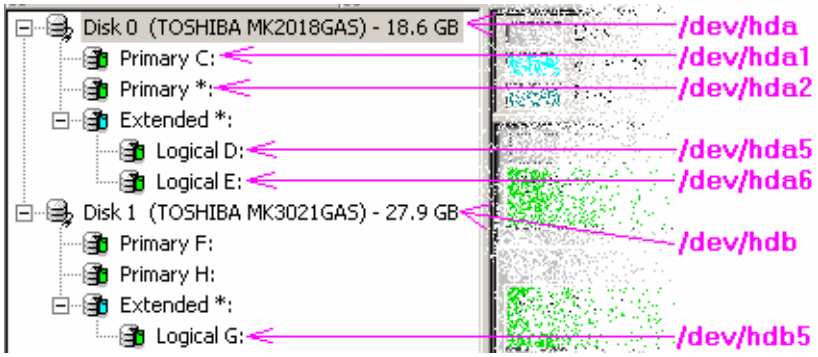
4.25.5 Comments

4.25.5.1 Partition enumeration rules in Linux

Linux lists detected hardware and resources in the `/dev` directory. All detected hard disks and partitions are included in this list as specifically named entries (see also the section Create Partition➔Comments➔Order of Partitions).

Hard disks are listed in the following manner:

- The 1st hard disk in the system is named `/dev/hda`
- The 2nd hard disk in the system is named `/dev/hdb`
- The 3rd hard disk in the system is named `/dev/hdc`
- And so on

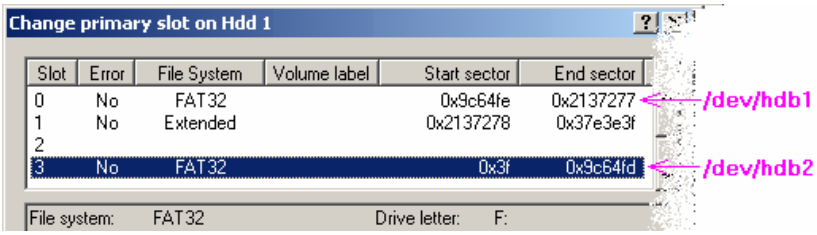


Partitions have symbolic names that encode both the hard disk number and the relative number of a partition on a disk:

- A first primary partition, which is referenced by the 1st used record in the Partition Table, is named:

- ⇨ `/dev/hda1` – on the 1st hard disk
- ⇨ `/dev/hdb1` – on the 2nd hard disk etc

- A second primary partition, which is referenced by the 2nd used record in the Partition Table, is named `/dev/hda2` (on the 1st hard disk), other primary partitions are named in similar fashion.
- A first logical partition is always named `/dev/hda5` (on the 1st hard disk), regardless of the actual amount of primary partitions.
- Next logical partitions are named in similar fashion, in accordance with their position: `/dev/hda6`, `/dev/hda7` ... `/dev/hdb5` …



4.25.5.2 Partition enumeration rules in Windows NT, 2000 and XP

Windows NT, 2000 and XP use enumeration rules that are similar with ones used in Linux, with some minor differences.

In Windows, hard disks are enumerated in accordance with BIOS standards. In fact, Windows generates a symbolic name for each hard disk that includes the disk number. These symbolic names can be used in Windows-based programs for direct accessing to disk contents. However, they are useless for PC users.

Partitions are enumerated separately for each hard disk, in the following manner:

- A first primary partition, which is referenced by the 1st used record in the Partition Table, is indexed with 1.
- A second primary partition, which is referenced by the 2nd used record in the Partition Table, is indexed with 2, other primary partitions are indexed in similar fashion.
- Logical partitions continue indexing of partitions.

As mentioned before, Windows NT, 2000 and XP take into account the enumeration of partitions only specific situations:

- In case of the most first connection of a hard disk, to automatically assign drive letters to new partitions.
- At the Windows startup, NTLDR uses partition's number to select the Windows system partition.

4.25.5.3 Fixing problems concerning incorrect settings in the BOOT.INI file

Operations of creating, deleting, restoring, copying and undeleting of primary partitions in Hard Disk Manager may lead to changing the number of the Windows system partition, which in turn may prevent Windows from a successful startup. The same problems can appear in case of using standard disk managing tools such as FDISK or Windows Disk Administrator.

This problem can be solved either by editing the BOOT.INI file or by using the *Change Primary Slot* function in Hard Disk Manager (see also the chapter Glossary).

BOOT.INI is the textual configuration file for NTLDR, a boot-managing module in Windows NT, 2000 and XP. This file contains the reference to the *Windows system partition* among other parameters.

The section `[operating systems]` of the BOOT.INI file lists bootable partitions, in the following manner:

```
[operating systems]

multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft Windows XP Professional" /fastdetect

C:\="Microsoft Windows"
```

The combination

```
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
```

defines the **Windows system partition**.

In case of the number of the Windows system partition is incorrectly assigned, the following error message appears after choosing the Windows boot configuration:

```
Windows could not start because the following file is missing
or corrupt:
<Windows root>\system32\hal.dll.
Please re-install a copy of the above file.
```

To solve the problem by editing the BOOT.INI,

one should change the number of the Windows system partition (a fragment `partition(x)`).

Another (and better) solution is adding an alternative bootable configuration, containing another partition number, one needs simply copy the boot configuration line around the original one, and change the partition number in the fragment `partition(x)` to another value. From this time, the startup menu will contain multiple boot configurations, so that one can try another boot configuration in case of the first one fails.

To solve the problem by using Hard Disk Manager,

One should run the program, select the Windows system partition and change its position in the Partition Table by using the function *Change Primary Slot*.

4.26 Make Logical / Make Primary

This chapter explains how to change a type of a partition (Primary to Logical or vise versa) with using Hard Disk Manager.

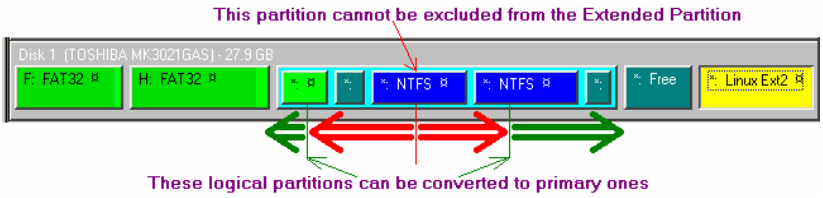
4.26.1 Overview

Hard Disk Manager provides the ability of including a Primary Partition in the Extended Partition, or excluding a Logical Partition from the Extended Partition, without duplicating a partition. This feature can be used for re-organization of the disk layout, swapping or changing amount of Primary Partitions.

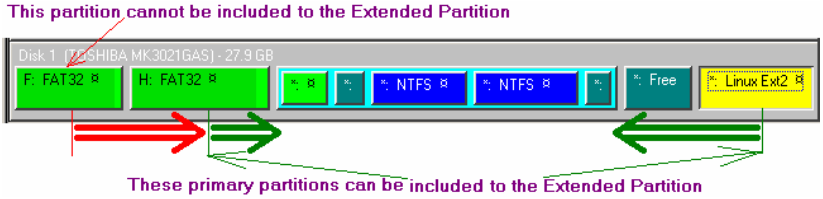
Formally, any primary partition can be copied inside the Extended Partition, or a logical partition can be copied outside the Extended Partition (so that its duplicate becomes a primary one). The disadvantage of this method is that a hard disk should contain enough unallocated free space to fit a duplicate of a partition.

The function *Make Primary / Make Logical* allows moving a partition inside or outside the Extended Partition without making a duplicate. The function is available only for partitions that are adjoining to the boundaries of the Extended Partition.

Converting a logical partition to a primary one:



Converting a primary partition to a logical one:



4.26.2 Initiating the operation

The actions you should make are similar to ones required for initiating other operations. See the chapter Initiating the backup operation in the Windows-based version for the detailed explanation of every step.

Step 1. Select a partition, which needs changing partition type

Select a *Primary* or a *Logical* partition in the Tree Layout panel, in the UDP Layout panel or in the List of Partitions.

The rules, which Hard Disk Manager uses in changing partition type, are explained in the section Rules of changing partition type.

Briefly, Hard Disk Manager allows:

- changing the first and the last Logical Partitions to Primary ones
- Primary Partitions, which are adjacent to the Extended Partition, changing to Logical ones (i.e. including them to the Extended Partition).

Step 2. Select the operation to perform

Variants:

for Logical Partitions:

- Select in the main menu:

  Partition ➔ Modify ➔ Make primary

- call the *popup menu* for the selected hard disk and select the item:

  Modify ➔ Make primary

for Primary Partitions

- Select in the main menu:

  Partition ➔ Modify ➔ Make logical

- call the *popup menu* for the selected hard disk and select the item:

  Modify ➔ Make logical

The function *Make Primary/Logical* does not require assigning any parameters other than selecting the targeted partition.

Step 3. Apply the operation

Hard Disk Manager supports both *Immediate* and *Virtual* execution of the *Make Primary/Make Logical* operation (see the chapter Virtual operations for more details). In the "*smart mode*", the program supports only Virtual execution, i.e. it schedules the operation in the List of Pending Operations.

- If *Virtual Execution* is disabled, the program starts the operation immediately after submitting parameters.
- If *Virtual Execution* is enabled, the program only schedules the operation in the *List of Pending Operations* for the future execution. For immediate execution of accumulated virtual operations, one should press the **Apply** button in the *Virtual Operations Toolbar* (see the section Applying operations).

4.26.3 Running the operation

During the real operation execution, the **Progress Information** window appears (see the section Progress Information for more details).

The program displays information about operation performance:

- Elapsed time and estimated remaining time
- Averaged read & write speed
- Amount of moved and remaining data
- Information about actions being performed.

Time required for completion of the operation *Make Primary / Make Logical* depends on parameters of a processed partition. Usually, it takes just several seconds for making changes in the Partition Table and waiting for accommodating changes in an operating system.

However, in some cases the operation requires additional time for changing size and the starting position of the processed partition. In this case, the operation takes noticeable amount of time that is fundamentally depends on partition properties (filesystem type, used space and data distribution).

4.26.4 Comments

4.26.4.1 Rules of changing partition type

The selected partition should adjoin to borders of the Extended Partition. The program permits a block of free space between the partition and the border of the Extended Partition:

- The first Logical Partition can be changed to a Primary one, which will be located before the Extended Partition.
  (!) The operation is available only in case of the actual amount of Primary Partitions is less than 3 (three).
- The last Logical Partition can be changed to the Primary one, which will be located just after the Extended Partition.
  (!) The operation is available only in case of the actual amount of Primary Partitions is less than 3 (three).
- The Primary Partition, which is left adjacent to the Extended Partition, can be changed to the Logical one. It will become the first Logical Partition.
- The Primary Partition, which is right adjacent to the Extended Partition, can be changed to the Logical one. It will become the last Logical Partition.
- In case of there is no Extended Partition on a disk, a Primary Partition still can be made a Logical one. The program slightly shrinks the selected partition, then it creates the Extended Partition and finally, it includes the selected partition to the Extended Partition.

Hard Disk Manager takes into account rules of the *DOS partitioning scheme* (see the chapter Glossary for more details).

1. The total amount of Primary Partitions plus the Extended Partition should not exceed four (4). It is the fundamental limitation of the DOS partitioning scheme.
2. Hard Disk Manager follows the partition alignment rules that are effective in the DOS partitioning scheme. If required, the program makes corrections in the starting position and capacity of a partition.

The operation of converting a logical partition to a primary one usually takes a little time because it requires only modifying few records in the Partition Table.

On the other hand, the operation of converting a primary partition to a logical one usually requires re-aligning a partition in accordance with the alignment rules of the DOS partitioning scheme. This action requires some time, because the partition needs to be slightly *resized* and *moved*.

4.26.4.2 Solving issues with misaligned primary partitions

Partitions, which were converted from logical to primary by using Hard Disk Manager, have starting positions being not aligned to the beginning of corresponding *cylinders*. So that these partitions do not meet accurately requirements of the DOS partitioning scheme.

In spite of this minor discordance with partitioning standards, operating systems usually detect and access such partitions without problems.

However, there can be situations, when an operating system fails correctly accessing a misaligned primary partition. There is an easy solution of this problem: one should apply the Resize & Move Partition operation on the misaligned primary partition: its starting position should be displaced to any allowable value, even to the least admissible one.

The thing is that Hard Disk Manager suggests only correctly aligned positions of partition's edges in the Resize & Move Partition operation. So that after completion of this operation the resulting partition will be aligned correctly.

4.26.5 Working with locked partitions

The difference between *locked* and *unlocked* partitions is that locked ones are used by other programs for file input-output activity (see Glossary).

The program cannot complete this operation on locked partitions, because it may interfere disk input-output activity of an operating system or other applications.

Hard Disk Manager requires rebooting the system to a single-tasking environment in order to complete the *Make Primary/Logical* operation, in case of a locked partition is selected.

- In Windows 95 and 98, Hard Disk Manager uses the "true" DOS session as the single-tasking environment.
- In Windows NT, 2000 and XP, Hard Disk Manager uses the so-called "*startup Bluescreen*" phase.
- In Windows ME, Hard Disk Manager requires to reboot from a bootable diskette or CD-ROM.

The detailed explanation of step-by-step actions is given in following sections:

- Working with locked/system partitions and hard disks
- Backup system and locked partitions

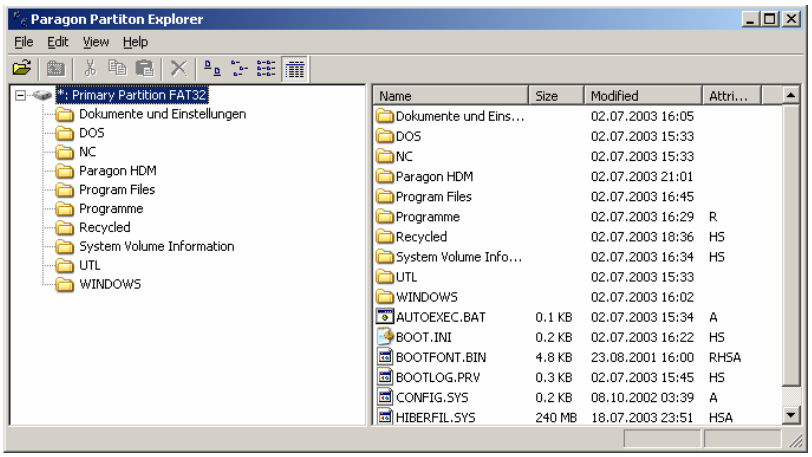
4.27 Supplementary Utilities

4.27.1 Paragon Partition Explorer

4.27.1.1 Overview

The Windows-based version of Hard Disk Manager includes the *Paragon Partition Explorer* utility.

Partition Explorer provides the ability of browsing and editing contents of FAT16, FAT32, NTFS, and Ext2/Ext3 partitions.



The program uses built-in OS-independent drivers for accessing partitions, it can work with both mounted and unmounted partitions.

Partition Explorer supports the following functionality:

1. Browsing contents of single partitions or entire hard disks.
2. Exporting single and multiple files and folders
3. Importing single and multiple files and folders
4. Creating new folders and files
5. Renaming and deleting files and folders
6. Opening documents from browsed partitions

In addition, the utility allows accessing NTFS metadata as if they were ordinary files.

4.27.1.2 Opening and editing documents

When a user opens a document in the Partition Explorer, the utility actually copies a file to the Windows TEMP directory, and then it opens the duplicate file by an appropriate application, which is registered in Windows for this type of files.

One should take into account that the utility is unable copying back a modified document, which was opened in such a way. To modify the document, one should follow the procedure:

1. Export the appropriate file
2. Open the document in Windows, in an usual fashion
3. Complete editing the document
4. Import back the appropriate file by using the Partition Explorer.

4.27.2 Paragon Image Explorer

4.27.2.1 Overview

The Windows-based version of Hard Disk Manager includes the *Paragon Image Explorer* utility.

Image Explorer provides the ability of browsing contents of backup archives made by Hard Disk Manager. Exactly, the program supports only FAT16, FAT32, NTFS, and Ext2/Ext3 partitions saved in backup images.

Image Explorer supports the following functionality:

1. Browsing contents of images of single partitions or hard disks.
2. Exporting single and multiple files and folders from any images

3. Opening documents from browsed partitions.

4.27.2.2 Opening and editing documents

When a user opens a document in the Image Explorer, the utility actually copies a file to the Windows TEMP directory, and then it opens the duplicate file by an appropriate application, which is registered in Windows for this type of files.

One should take into account that any changes made in a duplicate file, will not saved back to the original document.

4.27.3 Diskette Build Wizard

4.27.3.1 Overview

Diskette Build Wizard is purposed for creating DOS-based bootable diskettes that contain the DOS-based version of Hard Disk Manager.

Such diskettes are traditionally aimed for solving following tasks:

1. Continuing interrupted operations when Hard Disk Manager worked in the *fail-safe mode* (see the section Settings overview➔Data loss protection).
2. Completing operations in Windows ME, in case of a locked partition was targeted.

The current version of Hard Disk Manager includes the Linux-based bootable CD, which can be used for the same purposes. The use of the Linux-based bootable CD is preferred, as it provides better performance and supplies some advanced abilities such as network and CD/DVD burning drives support.

4.27.4 SID Changer

4.27.4.1 Overview

Hard Disk Manager includes the *SID Changer* utility that allows effectively changing computer's Security IDentifiers in Windows based configurations.

Briefly, computer SID is used for distinguishing user *access privileges* in workgroup local networks. When workstations are cloned from a template one, they will have the same SID. Later, when user accounts will be created on these stations, there can be a situation of duplicating security identifiers of different users registered in various workstations.

In this case, the workgroup cannot distinguish between these users: they will have the same privileges even if this is not the aim.

Hard Disk Manager uses the built-in SID Changer module that functionally equal to the standalone version.

4.27.4.2 Description of command-line parameters

By default, the utility searches BOOT.INI files on all partitions, picks up information about Windows installations from these files and then performs changing of SID in all found Windows instances. The program automatically generates random SID values.

However, this behavior can be corrected. The built-in SID Changer can be governed through the command line.

The general layout of the command line:

```
[/d<dirlist>|/e<dirlist>] [/s]
```

Comments:

- Any and even all of these keys can be omitted
- Keys `/d` and `/e` cannot be used together.

Description

`/d<dirlist>`	Look-up extra directories in addition to ones mentioned in BOOT.INI files.
`/e<dirlist>`	Look-up only these directories, do not analyze BOOT.INI files at all.

/s	The program should ask a user for SID value to be assigned, for each Windows instance.

5 Glossary

Backup Image

A backup image of a partition is the "momentary photography" of the filesystem state. Hard Disk Manager creates backup images that contain both user files and *metadata*. The image keeps contents of all files and the structure of directories, the location map for all files and directories, file attributes and other related information. For example, the image of NTFS partition contains also security information for each file, quotas, encryption information (for encrypted files and folders), compression information (for compressed files and folders), all extended attributes that NTFS support: multiple data streams, hard links, reparse points and so on.

Locked partitions

In terms of Hard Disk Manager, locked partitions are ones having files open for writing. In fact, locked partitions are ones that Hard Disk Manager cannot lock itself for the exclusive use.

The thing is that under multi-tasking environments such as Windows or Linux, several applications can simultaneously access to the same on-disk objects. The uncoordinated multiple access can damage or destroy the object. The mechanism of locks allows avoiding troubles of that kind.

Unlocked partitions

Partitions, which can be locked for exclusive use by Hard Disk Manager. Unmounted partitions are usually unlocked ones.

Locked and unlocked hard disks

From the point of view of Hard Disk Manager, hard disks are placeholders of partitions. If any of on-disk partitions is locked, the entire hard disk is treated as locked. On the contrary, if there are no locked partitions on the disk, it is the unlocked one.

Multi-partition image

Backup images that include (or may include) more than one partition. Backup images of entire hard disks and images of Extended Partitions are multi-partition ones.

With this version of Hard Disk Manager, multi-partition images have the hierarchical structure, and nested partitions are placed to subordinate images. Generally, these sub-images can be isolated and used independently.

Multivolumic image

Multivolumic images are distributed over multiple files that are named "volumes of a multivolumic archive". The primary reasons of using multivolumic images are inability or inconvenience of fitting the image in one file.

For example, recordable CD discs have capacity 650-800Mb each, so that the image that occupies 2Gb of data should be broken in 3-4 parts in order to be placed on CD-R.

Another situation: it is known that FAT32 limits the size of a single file to the value of 4Gb although the size of FAT32 partition formally may reach 8Tb. To place the very large image on FAT32 partition one should split the image in multiple volumes.

Partitioning schemes

Generally, *Partitioning scheme* is the set of rules, constraints and the format of on-disk structures that keep information about the partitions that are located on the hard disk.

There are several partitioning schemes used in practice. The most widely used partitioning scheme is the so-called *DOS partitioning scheme*. It was introduced by IBM and Microsoft for the purposes of using multiple partitions in the disk subsystems on IBM PC compatible computers.

Another popular partitioning scheme is the so-called *LDM* (Logical Disks Model) that originates from UNIX mainframe systems. The Veritas Executive accommodates the simplified version of LDM to the Windows 2000 operating system.

Windows 2000 and XP support two quite different partitioning schemes: the old DOS partitioning scheme and the new one Dynamic Disk Management (*DDM*). The problem is that older versions of Windows do not support DDM. In addition, most hard disk utilities do not support it, too.

Hard disk Geometry

Traditionally, usable space of a hard disk is logically divided in *Cylinders*, Cylinders are divided in *Tracks* (or *Heads*), and Tracks are divided in *Sectors*.

The triad of values {[Sectors-per-Track], [Tracks-per-Cylinder], [Amount-of-Cylinders]} is usually named *Hard Disk Geometry* or *C/H/S geometry*.

Tracks and Cylinders are enumerated from "0", while Sectors are enumerated from "1".

These disk parameters play important role in the *DOS Partitioning scheme*. The *alignment* of partitions should take into account parameters of the hard disk geometry.

Modern hardware use an advanced scheme of *linear addressing* of Sectors, which assumes all on-disk sectors continuously enumerated from 0. For backward compatibility with old standards, modern hard disks should additionally emulate C/H/S geometry.

MBR & 1st track of the Hard Disk

The 0th sector of the disk is named *MBR* (Master Boot Record). MBR contains the important information about the disk layout:

- The mention of the used partitioning scheme.
- The starting records of the Partition Table.
- The standard bootstrap code (or the initial code of boot managers, disk overlay software or boot viruses).

Generally, the 0th sector is used for the very similar purposes in all existing partitioning schemes.

The capacity of the MBR is not enough to place rather sophisticated boot programs. so that the on-boot software uses the entire 0th track of the hard disk in addition to the 0th sector, because it's never included to any partition.

For example, boot managing utilities such as LILO, GRUB and Paragon Boot Manager are located in the 0th track.

SID & SID Changer

SID – Security IDentifier, the binary structure that is associated with some object in the system. Generally, SID is used for assigning valid security attributes to various objects. There are several types of SIDs that are used by operating system(s) for various purposes. Concerning to the Hard Disk Manager functionality, the Computer SID requires the special attention.

The Computer SID value is used in the Microsoft Workgroups network environment for identifying workgroup users and some types of network resources. In case of duplicating computer SID within a Workgroup, there can be a situation of inability to distinguish between some users (or network resources).

The SID Changer utility is used to avoid such a problem by re-generating a unique value for computer SID.

Extended Partition

The Extended Partition is the exceptional one. Ordinary partitions are aimed to reserve a block of disk space for some filesystem. The Extended Partition is not purposed for keeping filesystems, it is actually used for extending the Partition Table of the disk. Formally, the extended Partition is the container of so-called Logical Partitions. The fundamental feature of the Extended Partition is that it contains many partitions inside.

Active partition

Active partition (or *bootable* partition) is the one an operating system will boot from at the PC startup, in case of the system starts from a hard disk containing this partition.

In the DOS partitioning scheme, only Primary Partitions can be active, because of limitations of the standard bootstrap.

Hidden partitions

The concept of hidden partitions was introduced in the IBM OS/2 Boot Manager. Operating systems do not mount "hidden" partitions, preventing getting access to their contents.

A method of hiding partitions consists in changing the Partition ID value that is saved in an appropriate entry of the Partition Table, by XOR-ing the Partition ID with the 0x10 hexadecimal value.

This method is workable only in case of a set of usable Partition ID values is very limited. For large sets of usable Partition Ids, it can lead to confusing filesystem types; for example, Ext2 partitions are marked with 0x83 Partition ID value. A hidden Ext2 partition would be marked with 0x93 Partition ID value, which is same with the Amoeba Filesystem Partition ID.

Partition ID

Partition ID (or Filesystem ID) is an identifier of a filesystem that is placed in the partition. Partition ID is purposed for fast detecting partitions of supported types. Some of operating systems completely rely on the Partition ID in distinguishing supported partitions, while other are not.

Partition ID is saved in appropriate entries of the Partition Table. It takes only 1 byte.

Partition Label

The *Partition Label* (sometimes it is also named *Volume Label*) is a small textual field (up to 11 characters) that is located in the partition's boot sector. This value is used for notification purposes only. It is detectable by any partitioning tool including DOS's FDISK utility.

Modern operating systems use other methods of saving the Volume Label within the filesystem, as a special hidden file. The Volume Label can contain relatively large text, in multiple languages.

Generally, the Volume Label and the Partition Label are different.

BOOT.INI file

BOOT.INI is the textual configuration file for NTLDR, which is the specific boot-managing tool in Windows NT, 2000 and XP. This file contains the reference to the Windows system partition among other parameters.

By editing BOOT.INI, it is possible fixing some problems of disorganizing Windows startup process. Potentially, troubles of this kind may appear in case of creating or deleting primary partitions in Windows'NT/2000/XP managed systems.

The section `[operating systems]` of the BOOT.INI file lists bootable partitions, in the following manner:

```
[operating systems]

multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft Windows XP Professional" /fastdetect

C:\="Microsoft Windows"
```

The combination

```
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
```

defines Windows system partition. To create an alternative bootable configuration, just copy this line around the original one, and change the *partition's number* `partition(x)` to another value.

From this point, the startup menu will contain multiple boot configurations, so that one can try another boot configuration in case of the first one fails.

Filesystem metadata

The servicing structures of a filesystem, which contain information about allocating files and directories, security information etc, are named *filesystem metadata*.

Filesystem metadata are invisible for users and ordinary applications, because incompetent changes in metadata usually make a partition unworkable.

Cluster

The *Cluster* is the minimum unit of disk space picked up for a file.

Traditionally, a filesystem divides disk space of a partition in clusters of fixed size, for the purpose of performance improvement. Each cluster is a group of sectors, the amount of sectors should be a power of two, ranging from 1 to 128.

An operating system reserves a whole number of clusters for every file, regardless of the actual file size. The remainder of the last file's cluster is not used, and this unused space is named *waste space* (or *slack space*).

The waste space may take a noticeable part of on-partition usable space. For example, Temporary Internet Files (having a lot of small files) usually have 20% to 70% of slack space.

The way of reducing the amount of slack space is to diminish the *Cluster Size* value. On the contrary, a reducing of the *Cluster Size* value leads to slowing of file input-output operations.

Serial Number

In the DOS partitioning scheme, every hard disk and every partition has the *Serial Number*, it takes 32 bits and is expressed by an 8-figure hexadecimal value.

The hard disk's Serial Number is stored in the MBR. Its value is assigned at the initializing of the MBR sector by standard disk managing tools from Microsoft such as Windows Disk Administrator and FDISK utility.

In fact, the hard disk's Serial Number is unimportant for most operating systems and software. It is known that Windows NT, 2000 and XP keep hard disk's Serial Number values in the database of assigned drive letters.

The partition's Serial Number is stored in the partition's Boot Sector (in FAT16, FAT32 and NFTS filesystems). Its value is assigned at the partition's formatting.

In fact, the partition's Serial Number is unimportant for most operating systems and software.

Root Directory

The top-level directory of a formatted logical drive is named the *Root Directory*. The Root Directory includes other files and directories.

In modern filesystems (e.g. Ext2/Ext3, FNTFS and even FAT32), the Root Directory does not differ from other directories in properties. This is not the truth for old FAT12 and FAT16 filesystems.

On FAT12 and FAT16 partitions, the Root Directory is placed outside the common space proposed for saving files. The place for the Root Directory is allocated at the moment of formatting a partition. The maximum amount of files and directories, which can be placed in the Root Directory, is limited by a value depending on the size of the Root Directory, and this value cannot be reduced or enlarged until the partition is re-formatted.

According to FAT16 standards, the Root Directory takes a whole number of sectors, and each directory entry takes 32 bytes. On modern disks, the sector size is usually 512bytes. So that the capacity of the Root Directory is a number that is divisible by 16.