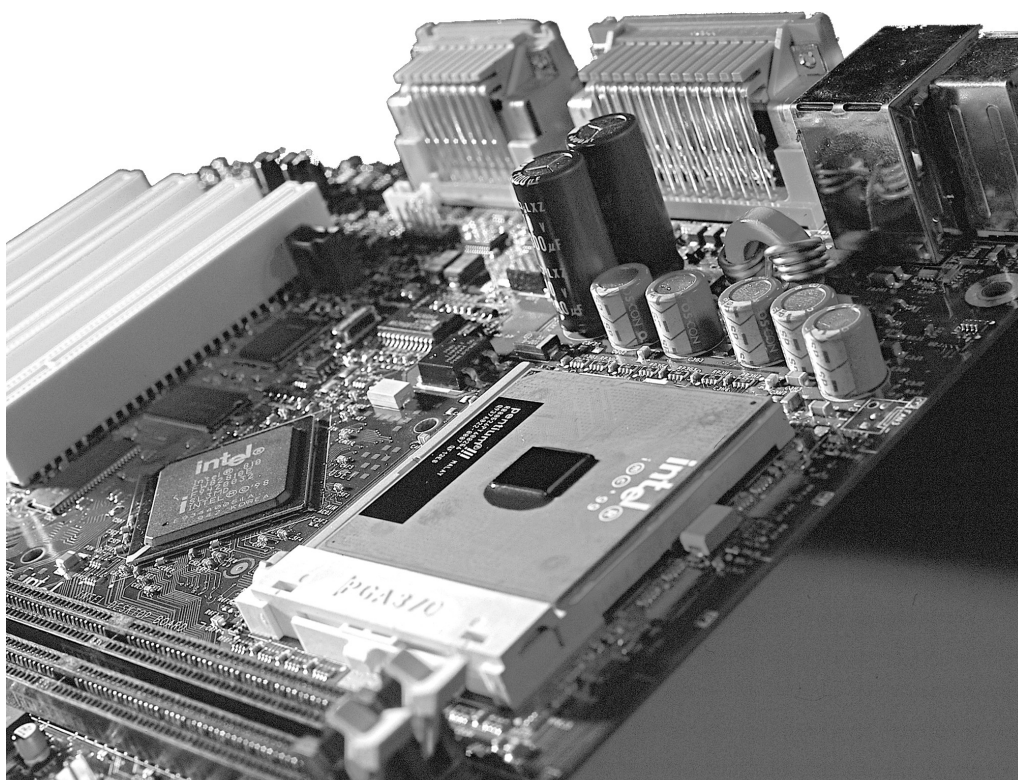


Глава 25

Файловые системы и восстановление данных



Загрузочные секторы

Для обеспечения пользовательским приложениям доступа к файлам независимо от типа используемого диска в операционной системе предусмотрено несколько структур. Среди них наибольшую роль играют загрузочные секторы (также называемые *загрузочными записями*). Загрузочные секторы бывают двух типов: главные загрузочные записи (Master Boot Record — MBR) и загрузочные записи тома (Volume Boot Record — VBR).

На физическом жестком диске может присутствовать только одна главная загрузочная запись, расположенная всегда в начале диска. Диск может содержать несколько разделов (также называемых *томами*); в начале каждого тома находится его загрузочная запись. Структура главной загрузочной записи MBR неизменна для различных операционных и файловых систем, в то время как структура загрузочной записи тома в значительной мере зависит от файловой системы, используемой на томе. Подробно структуры главной загрузочной записи MBR и загрузочной записи VBR рассматриваются в следующих разделах.

На рис. 25.1 представлено взаимоотношение этих структур на жестком диске объемом 8,4 Мбайт при использовании файловой системы FAT16.

Замечание

Некоторые съемные носители, например Iomega Zip, функционируют подобно “высокоемким дискетам”, т.е. на них нет загрузочных секторов главного и дополнительного разделов, а также диагностического цилиндра. Однако такие устройства, как Iomega Jaz, похожи по структуре на жесткие диски.

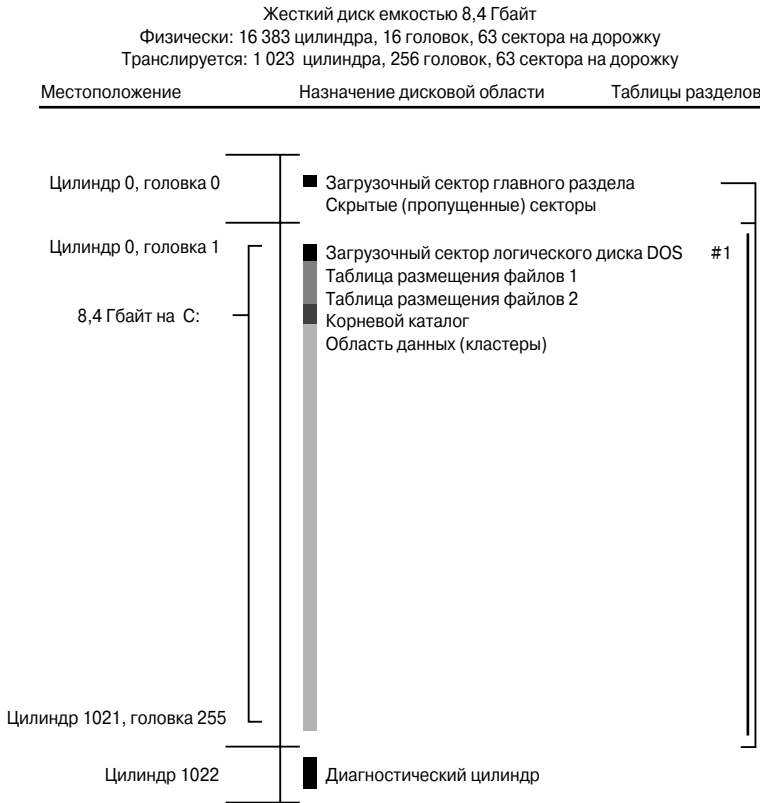


Рис. 25.1. Принцип размещения файлов на обычном жестком диске объемом 8,4 Гбайт при использовании файловой системы FAT16

Каждая дисковая область применяется для конкретной цели. Повреждение одной из перечисленных областей обычно приводит к ограничению доступа к другим областям, вызывая сбой в работе. Например, операционная система не сможет получить доступ к диску, если повреждена главная загрузочная запись. Таким образом, понимание логики работы каждой структуры и их взаимодействия оказывает существенную помощь в устранении неполадок.

Загрузочный сектор главного раздела

Впервые поддержка жестких дисков была реализована в DOS 2.0, выпущенной в 1983 году. В этой операционной системе впервые использовалась 16-разрядная файловая система FAT и поддерживалось деление диска на разделы, т.е. создание на диске логических томов. Выполнять разделение диска необходимо даже в том случае, если вы собираетесь использовать только один раздел. Разделы диска иногда называют логическими томами, поскольку операционная система присваивает каждому из них отдельную букву.

В настоящее время практически все накопители на жестких дисках делятся на несколько разделов, с которыми работает установленная операционная система. Однако, разделив диск, можно установить несколько операционных систем — по одной в каждый раздел, и эти системы смогут нормально сосуществовать на одном диске. Такая мультизагрузка становится все более популярной в настоящее время.

Совет

Для настройки системы с двойной загрузкой (например, с операционными системами Windows 9x/Me и Windows 2000/XP), не приобретая для этого диспетчер загрузки, установите сначала Windows 9x/Me, а затем на неиспользуемом пространстве первого жесткого диска или в первичном разделе дополнительного жесткого диска Windows 2000/XP. При установке Windows 2000/XP будет установлен также диспетчер начальной загрузки.

Чтобы установить на один жесткий диск несколько операционных систем, его надо разбить на разделы. Например, вы можете с помощью программы FDISK создать на диске один или несколько разделов для установки на них DOS или Windows 9x, а оставшуюся часть диска предоставить для другой операционной системы, например Linux. Каждый раздел в операционной системе будет иметь вид отдельного диска. Операционные системы Windows 9x/Me игнорируют неиспользуемые разделы и разделы, отформатированные не в файловой системе FAT, тогда как Windows 2000/XP “видят” разделы FAT и NTFS, но игнорируют разделы других форматов, таких, как Linux или HPFS операционной системы OS/2.

Несмотря на то что операционные системы Windows NT/2000/XP имеют программу создания разделов с интерфейсом командной строки, которая называется DISKPART, подготовка разделов обычно проводится с помощью программы с графическим интерфейсом Управление дисками (Disk Management) или программы Disk Administrator.

Информация о каждом разделе сохраняется в загрузочном секторе раздела (или логического диска) в начале каждого раздела. Существует также основная таблица списка разделов, помещенная в загрузочный сектор главного раздела.

Загрузочный сектор главного раздела, или *главная загрузочная запись* (*Master Boot Record — MBR*), является первым сектором на жестком диске (цилиндр 0, головка 0, сектор 1) и состоит из нескольких элементов.

- *Код начальной загрузки.* Команды, используемые для определения местоположения и загрузки загрузочной записи раздела (VBR) с активного (загрузочного) раздела.
- *Главная таблица разделов.* Таблица, состоящая из четырех 16-байтовых записей для четырех первичных разделов или трех первичных и одного расширенного разделов. Каждый первичный раздел определяет один логический диск, а расширенный раздел может быть разбит на несколько логических дисков. В записях определяется тип раздела и его физическое положение на диске, а также указывается, является ли данный раздел загрузочным и сколько секторов он занимает.

- *Байты сигнатуры.* Двухбайтовая сигнатура (55AAh), используемая ПЗУ системной платы и другими устройствами для подтверждения, что сектор не поврежден.

Основные и дополнительные разделы FAT

Количество разделов на всех жестких дисках в системе может достигать 24. Это означает, что в компьютере может быть установлено либо 24 отдельных накопителя, в каждом из которых имеется по одному разделу, либо один жесткий диск с 24 разделами, либо несколько накопителей с разным количеством разделов, но при условии, что общее количество разделов не превышает 24. Если общее количество разделов превысит эту цифру, DOS просто проигнорирует их, хотя другие операционные системы могут работать и с большим количеством томов. Такое ограничение DOS связано с тем, что в латинском алфавите от C до Z всего 24 буквы.

Замечание

Для создания разделов на жестком диске вместо программы `FDISK` обычно используется `Disk Administrator` в Windows NT или Управление дисками (`Disk Management`) в Windows 2000/XP. Тем не менее далее речь пойдет о программе `FDISK`. Разделы FAT емкостью до 2 Гиб могут создаваться с помощью любой из описываемых операционных систем.

Расширенный раздел представлен в главной таблице разделов аналогично основному, однако дисковое пространство расширенного раздела можно использовать для создания многочисленных логических разделов, или томов (`volumes`). На одном диске возможно создание только одного расширенного раздела, т.е. в таблице разделов не может быть более двух записей, относящихся к томам FAT.

Логическим томам, созданным в расширенном разделе, операционная система выделяет отдельные буквы дисков, которые, однако не указываются в главной таблице разделов. Логические тома не могут быть активными разделами и поэтому не позволяют осуществлять с них загрузку системы. На основе одного расширенного раздела можно создать до 23 томов (т.е., учитывая главный раздел FAT, всего их может быть 24).

Каждый подраздел (или том) в расширенном разделе содержит таблицу раздела, которая располагается в первом секторе подраздела. Первый сектор расширенного раздела содержит таблицу расширенного раздела, которая указывает на первый подраздел и иногда на другой расширенный раздел. Первый сектор этого расширенного раздела содержит таблицу другого расширенного раздела, которая, в свою очередь, может указывать (ссылаться) на другой том или дополнительный расширенный раздел. Цепочка ссылок продолжается, связывая все тома расширенного раздела в таблицу главного раздела. Обратите внимание: если запись расширенного раздела в главной загрузочной записи (`MBR`) будет повреждена или потеряна, то цепочка ссылок прервется в самом начале и все тома, содержащиеся в разделе, станут недоступными, т.е., по сути, исчезнут.

Безусловно, вряд ли кому-нибудь понадобится создавать 24 раздела на одном диске, но использование расширенного раздела позволит создать цепочку связанных разделов, чтобы преодолеть ограничение главной таблицы разделов, в соответствии с которым можно создавать не более четырех разделов.

Поскольку в главной загрузочной записи содержится первая программа, выполняемая системой при загрузке ПК, эта запись часто становится мишенью для создателей компьютерных вирусов. Если главная загрузочная запись будет уничтожена или заражена вирусом, то BIOS не найдет активный раздел и операционная система не будет загружена. Расположенный в загрузочной записи вирус будет загружаться раньше, чем антивирусная программа. Для его удаления необходимо запустить систему с неинфицированного гибкого диска, загрузочного компакт-диска или жесткого диска, после чего воспользоваться антивирусной программой.

В начале каждого раздела содержится загрузочный сектор логического диска. Разбивая диск на разделы, необходимо создать активный (или загрузочный) раздел. Программа, со-

держащаяся в самом первом секторе на жестком диске, определяет, какой раздел активен, и передает управление его загрузочному сектору.

Разделы FAT 12, FAT 16, FAT 32 или NTFS используются главным образом в операционных системах Windows, но можно также создавать дополнительные разделы для Linux, Novell NetWare, HPFS (OS/2), AIX (Unix), XENIX и других файловых/операционных систем. Для этого обычно используются дисковые утилиты альтернативных операционных систем или инструментальные средства сторонних разработчиков, например программа Partition Magic от компании PowerQuest. Разделы, которые не распознаются определенной операционной системой, игнорируются. Если на одном компьютере установлено несколько операционных систем, диспетчер загрузки (который поставляется в комплекте с операционной системой или устанавливается отдельно) позволяет выбрать раздел, который будет активным при каждой загрузке системы. В качестве альтернативы можно установить несколько операционных систем в различных первичных разделах, а затем с помощью FDISK, DISKPART или какой-нибудь другой программы указать тот или иной раздел в качестве активного.

В табл. 25.1 приведен формат таблицы разделов, которая хранится в секторе главной загрузочной записи.

Таблица 25.1. Формат главной загрузочной записи

Смещение в шестнадцатеричной системе)	Смещение (в десятичной системе)	Название	Длина	Описание
000h	0	Код загрузки	446	Код начальной загрузки; загружает VBR из активного раздела
Первая запись в таблице разделов				
1BEh	446	Индикатор загрузки	1 байт	Статус загрузки; 80h — активная (загрузочная). В других случаях — 00h
1BFh	447	Первая головка	1 байт	Первая головка (или сторона) раздела в режиме CHS
1C0h	448	Первый цилиндр/сектор	16 бит	Первые цилиндр (10 бит) и сектор (6 бит) в режиме CHS
1C2h	450	Системный идентификатор	1 байт	Тип раздела/файловая система
1C3h	451	Последняя головка	1 байт	Последняя головка (или сторона) раздела в режиме CHS
1C4h	452	Последний цилиндр/сектор	16 бит	Последние цилиндр (10 бит) и сектор (6 бит) в режиме CHS
1C6h	454	Относительный сектор	4 байт	Счетчик начальных секторов раздела в режиме LBA
1CAh	458	Общее число секторов	1 байт	Общее количество секторов в разделе в режиме LBA
Вторая запись в таблице разделов				
1CEh	462	Индикатор загрузки	1 байт	Статус загрузки; 80h — активная (загрузочная). В других случаях — 00h
1CFh	463	Первая головка	1 байт	Первая головка (или сторона) раздела в режиме CHS
1D0h	464	Первый цилиндр/сектор	16 бит	Первые цилиндр (10 бит) и сектор (6 бит) в режиме CHS
1D2h	466	Системный идентификатор	1 байт	Тип раздела/файловая система
1D3h	467	Последняя головка	1 байт	Последняя головка (или сторона) раздела в режиме CHS
1D4h	468	Последний цилиндр/сектор	16 бит	Последние цилиндр (10 бит) и сектор (6 бит) в режиме CHS
1D6h	470	Относительный сектор	4 байт	Счетчик начальных секторов раздела в режиме LBA
1DAh	474	Общее число секторов	1 байт	Общее количество секторов в разделе в режиме LBA

Смещение в шестнадцатеричной системе)	Смещение (в десятичной системе)	Название	Длина	Описание
Третья запись в таблице разделов				
1DEh	478	Индикатор загрузки	1 байт	Статус загрузки; 80h — активная (загрузочная). В других случаях — 00h
1DFh	479	Первая головка	1 байт	Первая головка (или сторона) раздела в режиме CHS
1E0h	480	Первый цилиндр/сектор	16 бит	Первые цилиндр (10 бит) и сектор (6 бит) в режиме CHS
1E2h	482	Системный идентификатор	1 байт	Тип раздела/файловая система
1E3h	483	Последняя головка	1 байт	Последняя головка (или сторона) раздела в режиме CHS
1E4h	484	Последний цилиндр/сектор	16 бит	Последние цилиндр (10 бит) и сектор (6 бит) в режиме CHS
1E6h	486	Относительный сектор	4 байт	Счетчик начальных секторов раздела в режиме LBA
1EAh	490	Общее число секторов	1 байт	Общее количество секторов в разделе в режиме LBA
Четвертая запись в таблице разделов				
1EEh	494	Индикатор загрузки	1 байт	Статус загрузки; 80h — активная (загрузочная). В других случаях — 00h
1EFh	495	Первая головка	1 байт	Первая головка (или сторона) раздела в режиме CHS
1F0h	496	Первый цилиндр/сектор	16 бит	Первые цилиндр (10 бит) и сектор (6 бит) в режиме CHS
1F2h	498	Системный идентификатор	1 байт	Тип раздела/файловая система
1F3h	499	Последняя головка	1 байт	Последняя головка (или сторона) раздела в режиме CHS
1F4h	500	Последний цилиндр/сектор	16 бит	Последние цилиндр (10 бит) и сектор (6 бит) в режиме CHS
1F6h	502	Относительный сектор	4 байт	Счетчик начальных секторов раздела в режиме LBA
1FAh	506	Общее число секторов	1 байт	Общее количество секторов в разделе в режиме LBA
Байты сигнатуры				
1FEh	510	Сигнатура	2 байт	Сигнатура загрузочного сектора; должна быть 55Aahh

CHS — сектор головки цилиндра (*Cylinder head sector*).

LBA — логический адрес блока (*Logical block address*).

Данные, которые хранятся в таблице разделов, указывают системе, где начинается и где заканчивается раздел, каков его объем, является ли он загрузочным, а также какой тип файловой системы на нем используется. Сведения о начальном цилиндре, головке и секторе используются только системами, работающими в режиме CHS, который является стандартным для всех жестких дисков объемом 8,4 Гбайт и меньше. Значения CHS при работе с дисками объемом больше 8,4 Гбайт неприменимы. При работе с дисками подобного объема адресация возможна только в режиме LBA. В данном случае такие значения начального цилиндра, головки и сектора игнорируются, а используются только поля *Relative Sector* и *Total Sectors*. Поле *Relative Sector* позволяет определить начало раздела, а поле *Total Sectors* — длину. На основании данных двух значений система и определяет, где именно на жестком диске расположен раздел.

Замечание

Процессоры ПК имеют конструктивные особенности, о которых необходимо знать каждому пользователю, занимающемуся редактированием или обработкой секторов начальной загрузки. Числа размером более чем 1 байт фактически читаются “задом наперед”! Эти данные записываются в так называемом *формате с прямым порядком байтов* (little endian format), в котором вначале читается младший (наименее значимый) бит. Люди обычно читают числа слева направо, т.е. в формате с обратным порядком байтов, начиная чтение со старшего (наиболее значимого) бита. Но, поскольку процессоры, используемые в ПК, считывают байты в прямом порядке, все числовые значения, превышающие 1 байт, записываются так, что в начале считываются наименее значимые (младшие) байты, а в конце — наиболее значимые (старшие). Например, поле *Relative Sector* в главной загрузочной записи MBR первого раздела обычно содержит числовое значение 63, которое в шестнадцатеричном формате записывается как 3Fh, а в стандартном шестнадцатеричном формате с обратным порядком байтов — как 0000003Fh (длиной 4 байт). Тем не менее это же числовое значение, записанное в формате с прямым порядком байтов, выглядело бы как 3F000000h. Приведем другой пример. В разделе содержится в общей сложности 23 567 292 сектора (около 12 Гбайт). Это числовое значение, записанное в шестнадцатеричном формате, будет выглядеть как 01679BVBCh. Эти же данные, сохраненные в поле *Total Sectors* таблицы разделов MBR и записанные в шестнадцатеричном формате с прямым порядком байтов, будут выглядеть как BC9B6701h.

Кстати говоря, использование чисел, записанных в формате с прямым порядком байтов, стало препятствием для перехода от 8-разрядной (1-байтовой) конструкции процессоров к 16-разрядной (2-байтовой) и 32-разрядной (4-байтовой). Методы организации и реализации внутренних регистров определяют технологию обработки чисел, используемую процессором. Во многих процессорах, например в микросхемах Motorola PowerPC, используемых в системах Macintosh, чтение чисел выполняется в формате с обратным порядком байтов. В свою очередь, все процессоры, используемые в персональных компьютерах, создаются по технологии Intel, впервые реализованной в процессоре Intel 8088, разработанном для первых компьютеров IBM PC. Конечно, способ чтения чисел, используемый в определенном процессоре, для обычных пользователей не имеет никакого значения. Иметь дело с числовыми значениями, записанными в формате с прямым или обратным порядком байтов, приходится лишь специалистам, пишущим машинные программы или программы на языке ассемблера, а также пользователям, занимающимся редактированием или интерпретацией необработанных секторов начальной загрузки.

В табл. 25.2 приведены стандартные байты идентификации системы, а в табл. 25.3 — нестандартные.

Таблица 25.2. Байт идентификации системы в таблице разделов (стандартные значения)

Значение	Тип раздела	Режим трансляции	Размер раздела
00h	Нет	—	—
01h	Основной, FAT 12	CHS	0–15 Мбайт
04h	Основной, FAT 16	CHS	16–32 Мбайт
05h	Дополнительный	CHS	16–32 Мбайт
06h	Основной, FAT 16	CHS	32 Мбайт–2 Гбайт
07h	NTFS/HPFS	Все	Любой
0Ah	Диспетчер загрузки OS/2	Все	Любой
0Bh	Основной, FAT 32	LBA	512 Мбайт–2 Тбайт
0Ch	Основной, FAT 32	LBA	512 Мбайт–2 Тбайт
0Eh	Основной, FAT 16	LBA	32 Мбайт–2 Гбайт
0Fh	Дополнительный	LBA	32 Мбайт–2 Гбайт

CHS — сектор головки цилиндра (*Cylinder head sector*).

LBA — логический адрес блока (*Logical block address*).

Таблица 25.3. Байт идентификации системы в таблице разделов (нестандартные значения)

Значение	Тип раздела
02h	Корневой раздел MS-XENIX
03h	Пользовательский раздел MS-XENIX
08h	Раздел файловой системы AIX
09h	Загрузочный раздел AIX
12h	Диагностический раздел Compaq
40h	Раздел ENIX 80286
50h	Раздел Ontrack Disk Manager только для чтения (DOS)
51h	Раздел Ontrack Disk Manager для чтения и записи (DOS)
52h	Раздел CP/M или Mucrosoft System V/386
54h	Раздел Ontrack Disk Manager (не DOS)
55h	Раздел Micro House EZ-Drive (не DOS)
56h	Раздел Golden Bow Vfeature
61h	Раздел Storage Dimensions Speedstor
63h	Раздел IBM 386/ix или UNIX System V/386
64h	Раздел Novell NetWare 286
65h	Раздел Novell NetWare 386
75h	Раздел IBM PCIX
80h	Раздел Minix v.1.1 – 1.4a
80h	Раздел Minix v.1.1b или Linux
82h	Файл подкачки Linux
83h	Файловая система Linux
93h	Файловая система Amoeba
94h	Таблица неверных блоков Amoeba
B7h	Файловая система BSDI (дополнительный файл подкачки)
B8h	Файловая система BSDI (дополнительная)
DBh	Раздел Digital Research Concurrent DOS/CPM-86
E1h	Дополнительный раздел FAT 12 SpeedStor
E4h	Дополнительный раздел FAT 16 SpeedStor
F2h	Дополнительный раздел DOS 3.3+
F4h	Первичный раздел SpeedStor
FFh	Раздел дефектных блоков UNIX

При восстановлении поврежденного диска приведенные в табл. 25.2 и 25.3 значения можно ввести с помощью программы Diskedit из пакета Norton Utilities.

Недокументированные возможности программы FDISK

Это программа с очень большими возможностями, которые были еще более расширены в DOS 5 и следующих версиях. К сожалению, эти возможности никогда не документировались в руководстве по DOS и не были описаны даже в Windows. Самым важным из недокументированных параметров является /MBR. С его помощью программа FDISK перезаписывает данные в главном загрузочном секторе, оставляя неизменными таблицы разделов диска.

Параметр /MBR словно специально предназначен для уничтожения вирусов, “заражающих” главный загрузочный сектор диска (цилиндр 0, головка 0, сектор 1). Чтобы воспользоваться этой возможностью, введите следующую команду:

FDISK/MBR

После этого FDISK перезапишет код загрузочного сектора, оставляя таблицы разделов неизменными. В нормально работающей системе это не приведет к проблемам, но на всякий случай создайте резервную копию таблиц разделов на дискете.

Имейте в виду: таблицы разделов будут перезаписаны в том случае, если 2 байта контрольного кода (сигнатуры) 55AAh в конце сектора окажутся поврежденными. Но эта ситуация маловероятна. На самом деле в случае повреждения байтов сигнатуры вы сразу же об этом узнаете: система перестанет загружаться и будет вести себя так, как будто разделения диска вообще не существует. Если система после загрузки с проверенной дискеты или оптического диска не распознает жесткий диск, то его загрузочный сектор может быть инфицирован. В этом случае следует воспользоваться какой-либо современной антивирусной программой и попытаться восстановить поврежденные данные.

Внимание!

Команду `FDISK/MBR` следует использовать **только** в системах с нормальной структурой главной загрузочной записи. Если в системе для доступа ко всем разделам диска используются такие программы, как Disk Manager, EZ-Drive, MaxBlast, Data Lifeguard Tools и т.п., **не применяйте** команду `FDISK/MBR`. Это связано с тем, что в указанных программах для доступа к диску используется модифицированная версия MBR. Команда `FDISK/MBR` может привести к уничтожению изменений, сделанных этими программами в структуре диска, в результате чего данные станут недоступными.

В служебной программе Recovery Console операционной системы Windows NT/2000 существует функция, эквивалентная команде `FDISK/MBR`, которая называется `FIXMBR`. Аналогом команды `FDISK` в Recovery Console является функция `DISKPART`. Для получения более подробной информации по использованию этих функций загрузите программу Recovery Console и введите команду `HELP`.

Загрузочный сектор

Загрузочный сектор — это первый сектор на любой области диска, адресуемого в качестве раздела/тома (или же логического диска DOS). Например, на дискете или на диске Zip это самый первый физический сектор, так как дискету нельзя разбить на разделы и она имеет только один логический диск. На жестком диске загрузочный сектор (секторы) располагается в начале каждого раздела, не являющегося дополнительным, или в начале любой области диска, распознаваемой как логический диск DOS.

Эти секторы немного похожи на загрузочные секторы разделов, так как содержат таблицы со специальной информацией о логическом диске.

- *Команда перехода к коду загрузки.* Трехбайтовая команда безусловного ветвления (или перехода) Intel x86, при выполнении которой происходит переход к началу кода загрузки операционной системы.
- *Блок параметров BIOS.* Содержит специфическую информацию, относящуюся к данному разделу, например размер раздела, количество используемых секторов диска, размер кластеров и метка тома. Используется драйвером файловой системы для определения типа и состояния носителей. Зависит от типа файловой системы, используемой данными носителями.
- *Код загрузки.* Команды, используемые для определения местоположения и последующей загрузки ядра операционной системы или загрузочного файла, которым является обычно `IO.SYS` или `NTLDR`.
- *Байты сигнатуры.* Двухбайтовая сигнатура (55AAh), используемая ПЗУ системной платы и другим кодом для проверки правильности загрузочного сектора.

Загрузочный сектор дискеты загружается ROM BIOS, а при загрузке системы с жесткого диска MBR передает управление загрузочному сектору активного раздела. В обоих случаях загрузочный сектор логического диска получает управление. Он выполняет некоторые проверки и затем пытается прочитать с диска первый системный файл (файл `IO.SYS` в DOS/

Windows 9x/Me или NTLDR в Windows NT/2000/XP). Загрузочный сектор не виден, так как находится вне области хранения файлов логического диска.

Загрузочный сектор раздела (VBR), как правило, создается в процессе форматирования высокого уровня. Эта операция обычно выполняется с помощью программы FORMAT, включенной в операционные системы DOS и Windows. Можно также воспользоваться программой Disk Administrator в Windows NT или оснасткой Управление дисками (Disk Management) в Windows 2000/XP, где загрузочный сектор создается после разбивки диска на разделы. Загрузочный сектор, начинающийся с первого сектора, есть во всех разделах.

Замечание

Большинство современных систем поддерживают загрузку с других устройств, а не только с дискеты. Эта возможность обеспечивается системной BIOS. Например, некоторые системы, помимо загрузки с жесткого диска и дискеты, могут загружаться с накопителя CD-ROM или диска Zip. Во многих случаях можно загружаться даже с помощью дисководов, подключенных к портам USB, что позволяет значительно повысить функциональные возможности компьютерных систем.

Картриджи Zip и диски LS-120 также можно сделать загрузочными. При соответствующей настройке BIOS дисковод LS-120 заменит дисковод для гибких дисков в качестве диска A:. В ряде новых систем поддерживается загрузка с USB-устройств. Типы накопителей, используемые для загрузки системы, указаны в соответствующем меню системной BIOS.

Сектор VBR состоит из программы (выполняемого кода) и области данных. Единственная таблица данных, находящаяся в этом секторе, называется блоком параметров носителя или блоком параметров диска. Информация, содержащаяся в этой таблице, требуется операционной системе для проверки емкости логического раздела диска и размещения таких важных структур, как таблицы размещения файлов (FAT) в разделах FAT или главной файловой таблицы (MFT) в разделах NTFS. Формат этих данных весьма специфичен.

Несмотря на то что во всех загрузочных секторах помимо кода загрузки содержится также блок параметров BIOS (BPB) и другие структуры, в загрузочном разделе из VBR выполняется только код начальной загрузки. Другие данные считываются операционной системой во время загрузки лишь для определения параметров раздела (тома).

В табл. 25.4 приведены форматы загрузочной записи для томов FAT12 и FAT16.

Таблица 25.4. Форматы загрузочной записи для томов FAT12 и FAT16

Смещение (в шестнадцатеричной системе)	Смещение (в десятичной системе)	Название	Длина, байт	Описание
000h	0	BS_jmpBoot	3	Команда перехода на код загрузки, как правило EB3C90h
003h	3	BS_OEMName	8	Имя производителя и версия DOS. Как правило, это имя MSWIN4. 1. После форматирования операционной системой не используется
00Bh	11	BPB_BytsPerSec	2	Размер сектора в байтах (обычно 512)
00Dh	13	BPB_SecPerClus	1	Размер кластера в секторах (степень числа 2). Как правило, это значения 1, 2, 4, 8, 16, 32 или 64
00Eh	14	BPB_RsvdSecCnt	2	Количество зарезервированных секторов (обычно 1)
010h	16	BPB_NumFATs	1	Количество копий FAT (обычно 2)
011h	17	BPB_RootEntCnt	Одно слово	Максимальное количество записей в корневом каталоге (обычно 512)
013h	19	BPB_TotSec16	2	16-битовое значение, определяющее количество секторов на томе, если оно не превышает 65 536. В противном случае используется значение BPB_TotSec32

Смещение (в шестнадцатеричной системе)	Смещение (в десятичной системе)	Название	Длина, байт	Описание
015h	21	BPB_Media	1	Байт описания диска (F8h для жесткого диска)
016h	22	BPB_FATSz16	2	Размер FAT в секторах
018h	24	BPB_SecPerTrk	2	Количество секторов на дорожке для прерывания 13h; для жестких дисков, как правило, равно 63
01Ah	26	BPB_NumHeads	2	Количество головок
01Ch	28	BPB_HiddSec	4	Количество скрытых секторов (для первого раздела, как правило, это значение равно 63)
020h	32	BPB_TotSec32	4	32-битовое значение, определяющее количество секторов на томе, если оно равно или превышает 65 536. В противном случае используется значение BPB_TotSec16
024h	36	BS_DrvNum	1	Физический номер диска (00h — дисковод, 80h — жесткий диск)
025h	37	BS_Reserved1	1	Зарезервировано (используется Windows NT) (как правило, равно 00h)
026h	38	BS_BootSig	1	Расширенная сигнатура загрузки; равна 29h при наличии следующих трех полей. В противном случае равна 00h
027h	39	BS_VolID	4	Серийный номер тома; используется вместе со значением BS_VolLab для отслеживания томов на съемных носителях. Как правило, формируется на основе даты и времени форматирования тома
02Bh	43	BS_VolLab	11	Метка тома. Соответствует 11-байтовой метке, записанной в корневом каталоге; при отсутствии метки тома данное значение равно NO NAME
036h	54	BS_FilSysType	8	Идентификатор файловой системы (FAT12 или FAT16). После форматирования операционной системой не используется
03Eh	62	BS_BootCode	448	Код программы загрузки
1FEh	510	BS_Signature	2	Байты сигнатуры (55AAh)

Загрузочный сектор (VBR) в разделе FAT 32 занимает три сектора, несмотря на то что в начале тома зарезервированы 32 сектора, предназначенные для базовой загрузочной записи раздела и ее резервной копии. Созданный по умолчанию VBR находится в секторах 0, 1 и 2, а его резервная копия занимает секторы 6, 7 и 8. Эти записи создаются во время форматирования раздела и при стандартном использовании не изменяются. Первый сектор содержит команду перехода, блок параметров BIOS (BPB), исходный код начальной загрузки и байты сигнатуры. Второй сектор называется FSInfo (File System Information) и содержит байты сигнатуры и информацию, которая используется программным обеспечением файловой системы; третий сектор содержит только дополнительный код загрузки и байты сигнатуры. В табл. 25.5 приведен формат и описана компоновка первого сектора загрузочной записи раздела FAT 32, которая занимает три сектора.

Таблица 25.5. Формат загрузочной записи раздела FAT 32, блок параметров BIOS (BPB), сектор 0

Смещение (в шестнадцатеричной системе)	Смещение (в десятичной системе)	Название	Длина, байт	Описание
000h	0	BS_jumpBoot	3	Команда перехода к коду загрузки; обычно EB5890h
003h	3	BS_OEMName	8	Идентификатор OEM ID; указывает, какая система форматировала раздел; обычно MSWIN4.1. После форматирования операционной системой не используется
00Bh	11	BPB_BytsPerSec	2	Количество байт в секторе; обычно 512
00Dh	13	BPB_SecPerClus	1	Количество секторов в кластере; должно быть число 2 в определенной степени. Обычно 1, 2, 4, 8, 16, 32 или 64
00Eh	14	BPB_RsvdSecCnt	2	Количество секторов, зарезервированных для загрузочной записи (записей); для разделов FAT 32 должно быть 32 сектора
010h	16	BPB_NumFATs	1	Количество структур FAT в разделе; обычно две
011h	17	BPB_RootEntCnt	2	Количество 32-байтовых записей каталога в корневом каталоге разделов FAT 12 и FAT 16; в разделах FAT 32 должно быть равно 0
013h	19	BPB_TotSec16	2	16-разрядное число, указывающее общее количество секторов в разделах, содержащих менее 65 536 секторов. Если число равно 0, то значение указано в поле BPB_TotSec32. Для разделов FAT 32 должно быть равно 0
015h	21	BPB_Media	1	Байт-дескриптор носителя; для всех несменных носителей обычно F8h, для большинства сменных носителей — F0h
016h	22	BPB_FATsSz16	2	16-разрядное число, указывающее общее количество секторов, занятых разделами FAT 12/16; в разделах FAT 32 это число должно быть равно 0. Количество секторов, занятых разделами FAT, указано в поле BPB_FATsSz32
018h	24	BPB_SecPerTrk	2	Количество секторов на дорожке для прерывания 13h; на жестких дисках обычно 63
01Ah	26	BPB_NumHeads	2	Число головок для прерывания 13h; на жестких дисках обычно 255
01Ch	28	BPB_HiddSec	4	Количество скрытых секторов, предшествующих разделу, который содержит этот том; для первого тома обычно 63
020h	32	BPB_TotSec32	4	32-разрядное число, указывающее общее количество секторов в разделах, содержащих 65 536 или более секторов. Если 0, то значение указано в поле BPB_TotSec16. В разделах FAT 32 не должно быть равно 0
024h	36	BPB_FATsSz32	4	32-разрядное число, указывающее количество секторов, занятых разделами FAT 32. Число, указанное в поле BPB_FATsSz16, должно быть равно 0
028h	40	BPB_ExtFlags	2	Только FAT 32. Биты 0–3. Номер активного раздела FAT с отсчетом от нуля. Действителен только в том случае, если зеркальное отражение отключено (бит 7 = 1). Биты 4–6. Зарезервированные. Бит 7. “0” указывает, что раздел FAT зеркально отражен, “1” — что раздел FAT активен.
02Ah	42	BPB_FSVer	2	Биты 8–15. Зарезервированные Номер версии раздела FAT 32. Старший байт является основным номером редакции, а младший — вторым номером. Это значение должно иметь вид 00h:00h

Окончание табл. 25.5

Смещение (в шестнадцатеричной системе)	Смещение (в десятичной системе)	Название	Длина, байт	Описание
02Ch	44	BPB_RootClus	4	Номер первого кластера корневого каталога; обычно 2
030h	48	BPB_FSInfo	2	Номер сектора структуры загрузочного сектора FSInfo в зарезервированной области раздела FAT 32; обычно 1
032h	50	BPB_BkBootSec	2	Номер сектора резервной копии загрузочной записи; обычно 6
034h	52	BPB_Reserved	12	Зарезервированное число; должно быть равно 0
040h	64	BS_DrvNum	1	Номер дисководов для прерывания 13h; обычно 00h для гибких дисков, 80h — для жестких дисков
041h	65	BS_Reserved	1	Зарезервированный номер (используемый Windows NT); должен быть равен 0
042h	66	BS_BootSig	1	Расширенная сигнатура начальной загрузки; при наличии трех следующих полей число должно быть равно 29h; в противном случае — 00h
043h	67	BS_VolID	4	Серийный номер раздела; используется со значением BS_VolLab, поддерживает отслеживание разделов на сменных носителях. В качестве начального числа обычно используется дата и время форматирования данного раздела
047h	71	BS_VolLab	11	Метка тома. Соответствует 11-байтовой метке тома, записанной в корневом каталоге; если метки нет, должно быть выражение NO NAME
052h	82	BS_FilSysType	8	Должно быть FAT 32. Не используется операционной системой после форматирования
05Ah	90	BS_BootCode	420	Код программы начальной загрузки
1FEh	510	BS_Signature	2	Байты сигнатуры; значение должно быть равно 55Aah

Загрузочная запись раздела FAT 32 занимает три сектора, вторым из которых является FSInfo. Формат и компоновка этого сектора представлены в табл. 25.6.

Таблица 25.6. Формат загрузочной записи раздела FAT 32, FSInfo, сектор 1

Смещение (в шестнадцатеричной системе)	Смещение (в десятичной системе)	Название	Длина, байт	Описание
000h	0	FSI_LeadSig	4	Ведущая сигнатура, используемая для проверки сектора; значение должно быть равно 52526141h
004h	4	FSI_Reserved1	1	Зарезервированное значение; должно быть равно 0
1E4h	484	FSI_StrucSig	4	Сигнатура структуры, используемая для проверки правильности сектора; значение должно быть равно 72724161h
1E8h	484	FSI_Free.Count	4	Последнее известное количество свободных кластеров в разделе. Если FFFFFFFFh, количество свободных кластеров неизвестно и должно быть еще раз вычислено операционной системой
1ECh	492	FSI_Nxt_Free	4	Следующий свободный кластер; это значение указывает на то, где система должна начинать поиск свободных кластеров. Обычно указывается номер последнего распределенного кластера. Если указано значение FFFFFFFFh, система должна начинать поиск с кластера 2
1F0h	496	FSI_Reserved2	12	Зарезервированное значение; должно быть равно 0
1FCh	508	FSI_TrailSig4	4	Конечная сигнатура; значение должно быть равно 000055Aah

Как уже отмечалось, загрузочная запись раздела FAT 32 занимает три сектора. В табл. 25.7 приведены формат и компоновка сектора Boot Code, который является третьим, и последним, сектором этой записи.

Таблица 25.7. Формат загрузочной записи раздела FAT 32, Boot Code, сектор 2

Смещение (в шестнадцатерич- ной системе)	Смещение (в десятичной системе)	Название	Длина, байт	Описание
000h	0	BS_BootCode	510	Код программы начальной загрузки
1FEh	510	BS_Signature	2	Байты сигнатуры; значение должно быть равно 55AAh

Следует заметить, что в третьем секторе загрузочной записи раздела (VBR) нет никаких данных, относящихся к определенной системе, т.е. его содержание в разных системах является одним и тем же. Таким образом, если этот сектор (и его резервная копия в LBA 8) будет поврежден, его можно скопировать в любом разделе FAT 32 и использовать полученную копию для восстановления поврежденного сектора.

Загрузочная запись (VBR) в разделах NTFS занимает 7 секторов, несмотря на то что в начале диска для VBR зарезервированы 16 секторов. В конце раздела находится зарезервированная 16-секторная область, предназначенная для резервной копии VBR. Первым из семи является сектор BPB, содержащий команду передачи управления, код BPB и байты сигнатуры. В секторах 2–7 содержится только дополнительный код начальной загрузки, без байтов сигнатуры и других структур. Код начальной загрузки не зависит от системы, поэтому содержание загрузочной записи любого раздела NTFS должно быть одним и тем же, за исключением ее первого сектора. В табл. 25.8 показаны формат и компоновка первого сектора загрузочной записи раздела NTFS, занимающей 7 секторов.

Таблица 25.8. Формат загрузочной записи раздела NTFS, блок параметров BIOS (BPB), сектор 0

Смещение (в шестнадцатерич- ной системе)	Смещение (в десятичной системе)	Название	Длина, байт	Описание
000h	0	BS_jumpBoot	3	Команда перехода к коду загрузки; обычно EB5890h
003h	3	BS_OEMName	8	Идентификатор OEM ID; указывает, какая система форматировала раздел; обычно NTFS. После форматирования операционной системой не используется
00Bh	11	BPB_BytsPerSec	2	Количество байт в секторе; обычно 512
00Dh	13	BPB_SecPerClus	1	Количество секторов в кластере; должно быть число 2 в некоторой степени. Значение обычно составляет 1, 2, 4 или 8
00Eh	14	BPB_RsvdSecCnt	2	Количество зарезервированных секторов, расположенных перед VBR; значение должно быть равно 0, иначе NTFS не сможет выполнить установку тома
010h	16	BPB_Reserved	3	Значение должно быть равно 0, или NTFS не сможет выполнить установку тома
013h	19	BPB_Reserved	2	Значение должно быть равно 0, или NTFS не сможет выполнить установку тома
015h	21	BPB_Media	1	Байт-дескриптор носителя; для всех несменных носителей обычно F8h, для большинства сменных носителей — F0h
016h	22	BPB_Reserved	2	Значение должно быть равно 0, или NTFS не сможет выполнить установку тома
018h	24	BPB_SecPerTrk	2	Количество секторов на дорожке для прерывания 13h; для жестких дисков значение обычно составляет 63

Смещение (в шестнадцатерич- ной системе)	Смещение (в десятичной системе)	Название	Длина, байт	Описание
01Ah	26	BPB_NumHeads	2	Число головок для прерывания 13h; на жестких дисках обычно 255
01Ch	28	BPB_HiddSec	4	Количество скрытых секторов, предшествующих разделу, который содержит этот том; для первого тома обычно 63
020h	32	BPB_Reserved	4	Значение должно быть равно 0, или NTFS не сможет выполнить установку тома
024h	36	Reserved	4	Не используется NTFS; обычно значение равно 80008000h
028h	40	BPB_TotSec64	8	Общее количество секторов в разделе
030h	48	BPB_MftClus	8	Номер логического кластера для запуска файла \$MFT
038h	56	BPB_MirClus	8	Номер логического кластера для запуска файла \$MFTMirr
040h	64	BPB_ClusPerMft	1	Количество кластеров в записи файл/каталог MFT. Положительное число (00h-7Fh) обозначает количество кластеров в записи MFT. Если число отрицательное (80h-FFh), то размер записи равен абсолютной величине этого числа, умноженной на 2
041h	65	Reserved	3	Не используется NTFS
044h	68	BPB_ClusPerIndx	1	Количество кластеров в буферном индексе-регистре; используется для распределения пространства каталогов. Положительное число (00h-7Fh) обозначает количество кластеров в записи MFT. Если число отрицательное (80h-FFh), то размер записи равен абсолютной величине этого числа, умноженной на 2
045h	69	Reserved	3	Не используется NTFS
048h	72	BS_VolID	8	Серийный номер раздела; используется для отслеживания разделов на сменных носителях. В качестве начального числа обычно указаны дата и время форматирования раздела
050h	80	Reserved	4	Не используется NTFS
054h	84	BS_BootCode	426	Код программы начальной загрузки
1FEh	510	BS_Signature	2	Байты сигнатуры; значение должно быть равно 55AAh

Область данных

Область данных диска — это область, следующая за загрузочным сектором, таблицами размещения файлов и корневым каталогом на любом логическом диске. Эта область контролируется с помощью FAT и корневого каталога и делится на ячейки размещения, называемые *кластерами*. В этих кластерах и располагаются сохраняемые на диске файлы.

Цилиндр для диагностических операций чтения и записи

Программа разбивки диска на разделы FDISK всегда резервирует последний цилиндр жесткого диска для выполнения диагностических операций. Из-за этого FDISK указывает меньшее количество цилиндров, чем существует на самом деле. Операционная система не использует этот цилиндр, поскольку он находится вне разделов.

В системах с дисковыми интерфейсами IDE или SCSI контроллер должен выделить дополнительную область после разделов для хранения таблицы испорченных дорожек и запасных секторов. В этом случае разница между фактическим числом цилиндров и тем, что показывает FDISK, будет еще больше.

Область диагностики позволяет выполнять тестирование чтения/записи жесткого диска, не повреждая данных на диске. Программы форматирования жестких дисков на низком уровне обычно используют этот цилиндр для тестирования чередования диска либо для хранения необходимой во время форматирования информации.

Файловые системы

Жесткий диск — основное устройство хранения данных. Причем структура хранения файлов и каталогов, обеспечивающая непосредственное расположение данных на диске, может быть разной. Файловая система чаще всего интегрирована в операционную, а некоторые операционные системы поддерживают несколько файловых систем.

При работе с современными ПК можно выбрать одну из нескольких файловых систем. Для каждой файловой системы характерны определенные ограничения, преимущества и недостатки; кроме того, ряд ограничений накладывается и используемой операционной системой. Не всегда операционная система поддерживает все файловые системы.

К основным файловым системам относятся следующие:

- FAT (File Allocation Table — таблица размещения файлов), которая разделяется на три типа: FAT12, FAT16 и FAT32;
- NTFS (New Technology File System — файловая система новых технологий).

Хотя при работе с ПК могут использоваться и другие файловые системы, например HPFS (High Performance File System — высокопроизводительная файловая система) операционной системы OS/2, на компьютерах, работающих под управлением Windows, они не используются. Поэтому далее в главе основное внимание уделяется файловым системам FAT и NTFS.

Кластер (ячейка размещения)

Термин *кластер* иногда заменяется термином *ячейка размещения* (*allocation unit*). Новое обозначение — синоним старого, так как кластер является наименьшей ячейкой на диске, которой может оперировать система при чтении или записи файла на диск. Кластер соответствует одному или (чаще всего) нескольким секторам. Это позволяет уменьшить размер FAT и ускорить работу операционной системы, так как ей приходится оперировать меньшим числом распределяемых ячеек. В то же время с увеличением размера кластера на диске растет и размер неиспользуемого дискового пространства, так как его распределение происходит с дискретностью в один кластер.

Поскольку операционная система может оперировать только целыми кластерами, это приводит к потере довольно большого дискового пространства. Файлы редко бывают таких размеров, чтобы занимать целое количество кластеров, поэтому последний кластер, относящийся к файлу, редко оказывается заполненным до предела. Дополнительное пространство между фактическим окончанием файла и концом кластера называется *резервом*. Чем больше размер кластера в разделе, тем больше объем “бесполезного” резервного пространства.

Использование кластеров больших размеров ощутимо сказывается на работе системы. Например, на диске емкостью 2 Гбайт, содержащем 5 000 файлов, со средней потерей дискового пространства в полкластера на один файл суммарные потери дискового пространства составят около 78 Мбайт: $5000 \times (0,5 \times 32)$. Когда файлы размером менее 32 Кбайт хранятся на диске, ячейки размещения которого имеют размер 32 Кбайт, потеря емкости может составить до 40% от общего объема жесткого диска. Поскольку раздел FAT 32 имеет больше кластеров, чем раздел FAT 16, размер кластера уменьшается. Использование меньшего кластера снижает потери дискового пространства. Например, раздел размером 2 Гбайт с 5 000 файлов в FAT 32 использует кластер размером 4 Кбайт, вместо 32 Кбайт в FAT 16. Такое уменьшение размера кластера снижает потери дискового пространства с 78 до 10 Мбайт.

Замечание

Сколько пространства тратится впустую при использовании кластеров того или другого размера? Чтобы получить ответ на этот вопрос, воспользуйтесь бесплатной утилитой Karen's Disk Slack Checker, которую можно получить на Web-сайте Карена Кенворси (Karen Kenworthy), бывшего редактора журнала Windows Magazin. Для этого следует обратиться по адресу: <http://www.karenware.com/powertools/ptslack.asp>.

Типы файловой системы FAT

До появления Windows XP наиболее распространенные файловые системы создавались на основе таблицы размещения файлов (FAT), которая отслеживает данные, хранящиеся в каждом кластере жесткого диска. Кроме того, FAT является универсальной файловой системой, которая распознается практически любой операционной системой, установленной на персональных компьютерах и рабочих станциях. Например, FAT распознается даже системами Apple Macintosh. Поэтому, несмотря на то что для Windows XP обычно рекомендуется NTFS (о которой речь идет в следующих разделах главы), для большей совместимости с различными системами и платформами внешними накопителями на жестких дисках и дисковых со сменными носителями в качестве собственной файловой системы чаще всего используется FAT. Таким образом, если необходимо создать систему с двойной загрузкой, т.е. установить Windows XP и Windows 9x/Me на одном компьютере, придется использовать FAT даже на основных накопителях.

Как уже отмечалось, существует три основные версии системы FAT — FAT 12, FAT 16 и FAT 32, которые отличаются разрядностью чисел, используемых в таблицах размещения файлов. Другими словами, для отслеживания кластеров данных в FAT 16 используются 16-разрядные числа, в FAT 32 — 32-разрядные и т.д. Системы FAT подразделяются следующим образом:

- *FAT 12*, используемая в разделах емкостью не более 16 Мбайт (например, дискета);
- *FAT 16*, используемая в разделах емкостью от 16 Мбайт до 2 Гбайт; операционные системы Windows NT/2000/XP поддерживают разделы FAT 16 емкостью до 4 Гбайт;
- *FAT 32*, используемая в разделах емкостью от 512 Мбайт до 2 Тбайт.

Файловые системы FAT 12 и FAT 16 изначально применяются в DOS и Windows и поддерживаются практически всеми известными сегодня операционными системами. Большинство ПК поставляются с жесткими дисками, на которых установлена одна из файловых систем FAT.

Несмотря на то что все операционные системы ПК поддерживают FAT 12 и FAT 16, Windows 2000 и Windows XP поддерживают FAT 32 и NTFS, причем последняя совершенно не совместима с FAT.

FAT12

Первой файловой системой, используемой в компьютерах IBM PC, представленных 12 августа 1981 года, была FAT12. Она до сих пор используется на дискетах и разделах FAT объемом меньше 16 MiB. Файловая система FAT12 использует таблицу, содержащую 12-битовые значения для управления кластерами (также называемыми *ячейками размещения*). *Кластер* — это единица хранения информации в области данных. Каждому файлу соответствует, как минимум, один кластер, а при увеличении объема файла увеличивается и количество используемых кластеров. Сведения о размерах кластеров FAT12 представлены в табл. 25.9.

Таблица 25.9. Размеры кластеров FAT12

Тип носителя	Размер раздела	Количество секторов в каждом кластере	Размер кластера, КиБ
Дискета 5,25 дюйма двойной плотности (DD)	360 Кбайт	2	1
Дискета 3,5 дюйма двойной плотности (DD)	720 Кбайт	2	1
Дискета 5,25 дюйма высокой плотности (HD)	1,2 Мбайт	1	0,5
Дискета 3,5 дюйма высокой плотности (HD)	1,44 Мбайт	1	0,5
Дискета 3,5 дюйма сверхвысокой плотности (ED)	2,88 Мбайт	2	1
Другие накопители	0–15,9 МиБ	8	4

КиБ (KiB) = кибибайт (Kibibyte) = 1 024 байт. МиБ (MiB) = меbibайт (Mebibyte) = 1 048 576 байт.

Каждый кластер раздела FAT12, как правило, содержит восемь секторов, за исключением дискет, размер кластера которых зависит от типа дискеты. Двенадцатибитовые значения кластеров изменяются в диапазоне от 000h до FFFh (в шестнадцатеричной системе измерений) или от 0 до 4 095 (в десятичной системе измерений). В результате теоретически максимальное количество кластеров равно 4 096; однако 11 значений оказываются зарезервированными, поэтому их нельзя назначать фактическим кластерам на диске. Значения кластеров начинаются с 2 (значения 0 и 1 зарезервированы), значение FF7h зарезервировано и указывает на поврежденный сектор, а значения FF8h – FFFh указывают на конец цепочки FAT, в результате чего остаются доступными 4 085 кластеров (4 096 – 11 = 4 085). Microsoft вычитает из полученного значения еще 1 для исключения определенных проблем, в результате чего на разделе FAT12 может содержаться не более 4 084 секторов.

На разделах FAT12 один сектор используется для хранения загрузочной записи и блока параметров BIOS (BPB), а также хранятся две копии таблицы размещения файлов (каждой из которых соответствует до 12 секторов). Кроме того до 32 секторов отводится под корневой каталог (меньшее количество секторов используется только на дискетах), а область данных ограничивается 4 084 кластерами. Поскольку каждый кластер FAT12 содержит восемь секторов (за исключением дискет), тома (разделы) FAT12 могут содержать максимум 32 729 секторов (1 сектор занимает загрузочная запись, 12 секторов — каждая из двух копий FAT, 32 сектора — корневой каталог, а 4 084 кластера, каждый из которых содержит восемь секторов, — данные). В результате получается значение 16,76 Мбайт, или 15,98 МиБ. Ограничения томов FAT12 представлены в табл. 25.10.

Таблица 25.10. Ограничения разделов FAT12

Ограничение раздела	Количество кластеров	Количество секторов в каждом кластере	Общее количество секторов в разделе	Объем раздела (десятичное значение)	Объем раздела (двоичное значение)
Максимальное значение	4 084	8	32,729	16,76 Мбайт	15,98 МиБ

Мбайт = мегабайт = 1 000 000 байт.

МиБ = меbibайт = 1 048 576 байт.

Операционные системы PC/MS-DOS версий 1.x и 2.x использовали только файловую систему FAT12, однако все последующие версии DOS и все версии Windows автоматически создают файловую систему FAT12 на любых дисках или разделах, содержащих 32 729 или меньшее количество секторов (16,76 Мбайт). При увеличении количества секторов автоматически используется файловая система FAT16, FAT32 или NTFS. Основные характеристики файловой системы FAT12 перечислены ниже.

- Используется на всех дискетах.

- Используется по умолчанию на разделах FAT объемом 16,76 Мбайт (15,98 МиБ) и меньше.
- Поддерживается всеми версиями DOS и Windows.
- Поддерживается всеми операционными системами, способными работать с дисками ПК.

Файловая система FAT12 до сих пор используется на носителях малого объема, поскольку 12-разрядные таблицы занимают меньше места, чем аналогичные таблицы FAT16 и FAT32, что позволяет использовать больший объем дискового пространства для хранения данных.

FAT16

Файловая система FAT16 во многом похожа на FAT12, за исключением того, что для управления кластерами на диске используются 16-разрядные значения. Она была представлена 14 августа 1984 года вместе с операционной системой PC/MS-DOS 3.0, поскольку это было необходимо для обеспечения поддержки жестких дисков большего объема. В отличие от FAT12, файловая система FAT16 используется на носителях, содержащих больше 32 729 секторов (т. е. объемом больше 15,98 МиБ, или 16,76 Мбайт). Теоретически FAT16 может поддерживать накопители объемом до 2 или ГиБ. Однако даже при использовании FAT16 операционная система DOS 3.3 (а также все более ранние версии) поддерживали разделы объемом не более 32 МиБ (33,55 Мбайт), поскольку первые версии DOS использовали внутреннюю 16-разрядную адресацию, в том числе и в блоке параметров BIOS BPB (он содержится в загрузочном секторе тома, который является первым логическим сектором в разделе FAT). Использование 16-битовых значений секторов в DOS 3.3 и более ранних версиях привело к тому, что поддерживались только диски, содержащие не более 65 535 секторов объемом 512 байт, что составляет 32 МиБ (33,55 Мбайт).

В качестве временного решения, позволившего работать с дисками большего чем 32 МиБ объема, в PC/MS-DOS 3.3 (представленной 2 апреля 1987 года), использовалось понятие расширенного раздела, который мог содержать до 23 подразделов (логических дисков) объемом 32 МиБ каждый. С учетом основного раздела диска это позволяло работать с 24 разделами объемом 32 МиБ каждый, которые с точки зрения операционной системы представляли собой логические диски C–Z.

Для того чтобы воспользоваться всеми преимуществами FAT16 и обеспечить поддержку накопителей и разделов большего объема, компании Microsoft и Compaq в ноябре 1987 года представили систему Compaq DOS 3.31. Это была первая операционная система, которая использовала внутреннюю 21-разрядную адресацию, в том числе и в блоке параметров BIOS BPB. Данная возможность получила широкое распространение только с 19 июля 1988 года, когда компании Microsoft и IBM выпустили PC/MS-DOS 4.0. Эта операционная система позволяла при использовании FAT16 работать с разделами объемом до 2 ГиБ (при этом каждый кластер содержал 64 сектора).

Как уже отмечалось, каждый кластер раздела FAT16 содержит 64 сектора. Шестнадцатитрехбитовые значения кластеров изменяются в диапазоне от 0000h до FFFFh (в шестнадцатеричной системе измерений) или от 0 до 65 535 (в десятичной системе измерений). В результате теоретическое максимальное количество кластеров равно 65 536; однако 11 значений оказывается зарезервированными, поэтому их нельзя назначать фактическим кластерам на диске. Значения кластеров начинаются с 2 (значения 0 и 1 зарезервированы), значение FFF7h зарезервировано и указывает на поврежденный сектор, а значения FFF8h– FFFFh указывают на конец цепочки FAT, в результате чего остаются доступными 65 525 кластеров (65 536 – 11 = 65 525). Microsoft вычитает из полученного значения еще 1 для исключения определенных проблем, в результате чего на разделе FAT16 может содержаться не более 5 524 секторов. Сведения о размерах кластеров на разделах (томах) FAT16 представлены в табл. 25.11.

Таблица 25.11. Размеры кластеров FAT16

Объем раздела	Количество секторов в каждом кластере	Размер кластера, КиБ
4, 1–15,96 МиБ ¹	2	1
От 5,96 до 128 МиБ	4	2
От 128 до 256 МиБ	8	4
От 256 до 512 МиБ	16	8
От 512 МиБ до 1 ГиБ	32	16
От 1 до 2 ГиБ	64	32
От 2 ГиБ до 4 ГиБ ²	128	64

1 На разделах объемом меньше 16 МиБ по умолчанию используется файловая система FAT12; однако с помощью специальных параметров можно задать и файловую систему FAT32.

2 Разделы объемом больше 2 ГиБ поддерживаются только Windows NT/2000/XP.

Мбайт = Мегабайт = 1 000 000 байт.

КиБ (KiB) = кибибайт (Kibibyte) = 1 024 байт.

МиБ (MiB) = мебибайт (Mebibyte) = 1 024 КиБ = 1 048 576 байт.

ГиБ (GiB) = гиббибайт (Gibibyte) = 1 024 МиБ = 1 073 741 824 байт.

На разделах FAT16 один сектор используется для хранения загрузочной записи и блока параметров BIOS (BPB), а также хранятся две копии таблицы размещения файлов (используемая по умолчанию и резервная), каждая из которых занимает 256 секторов. Кроме того до 32 секторов отводится под корневой каталог, а область данных ограничивается 65 524 кластерами. Поскольку каждый кластер FAT16 содержит 64 сектора, тома (разделы) FAT16 могут содержать максимум 4 194 081 сектор (один сектор занимает загрузочная запись, 256 секторов — каждая из двух копий FAT, 32 сектора — корневой каталог, а 65 524 кластера, каждый из которых содержит 64 сектора, — данные). В результате получается значение 2,15 Гбайт, или 2 ГиБ. Ограничения томов FAT16 представлены в табл. 25.12.

Таблица 25.12. Ограничения разделов FAT16

Ограничение раздела	Количество кластеров	Количество секторов в каждом кластере	Общее количество секторов в разделе	Объем раздела (десятичное значение)	Объем раздела (двоичное значение)
Минимальное значение	4 167	2	8 401	4,3 Мбайт	4,1 МиБ
Максимальное значение в DOS 3.0-3.3	16 343	4	65 533	33,55 Мбайт	32 МиБ
Максимальное значение в Win9x/Me	65 524	64	4 194 081	2,15 Гбайт	2 ГиБ
Максимальное значение в Windows NT/2000/XP	65 524	128	8 387 617	4,29 Гбайт	4 ГиБ

Замечание

Операционные системы Windows NT/2000/XP могут создавать разделы (тома) FAT16, содержащие 128 секторов в каждом кластере размером 64 КиБ, в результате чего максимальный объем раздела составляет 4,29 Гбайт, или 4 ГиБ. Однако с данными, сохраненными в разделах, отформатированных подобным образом, не смогут работать никакие другие операционные системы. Для обеспечения максимальной совместимости в разделах FAT16 следует использовать кластеры объемом 32 КиБ, в результате чего максимальный объем раздела составляет 2,15 Гбайт, или 2 ГиБ.

Некоторые наиболее значимые характеристики FAT16 перечислены ниже.

- FAT16 полностью поддерживается MS-DOS 3.31 и последующими версиями, всеми версиями Windows, а также некоторыми версиями Unix.
- FAT16 весьма эффективна на разделах объемом меньше 256 MiB, однако при увеличении объема раздела ее эффективность снижается, так как размер кластера оказывается больше, чем при использовании FAT32 и NTFS.
- Информация, которая содержится в загрузочном секторе, автоматически не архивируется, поэтому при его повреждении получить доступ к разделу невозможно.
- В случае возникновения проблемы для ее устранения систему можно загрузить с помощью загрузочной дискеты MS-DOS. Выпускается немало специальных утилит для восстановления данных на разделах FAT16.
- Корневой каталог (папка) может содержать максимум 512 элементов. Количество элементов может уменьшиться при использовании длинных имен файлов.
- FAT16 не поддерживает функции защиты, шифрования и сжатия.
- Размеры файлов в разделах FAT16 ограничены только размерами раздела. Поскольку каждый файл занимает не менее одного кластера, в разделах FAT16 не может содержаться больше 65 524 файлов.

VFAT и длинные имена файлов

В оригинальной операционной системе Windows 95 используется та же файловая система, что и в DOS, но с существенными улучшениями. В Windows 95 поддерживается файловая система FAT, переписанная в 32-разрядный код и названная *виртуальной таблицей размещения файлов* (*virtual file allocation table* — VFAT). Эта таблица используется вместе с 32-разрядной программой VSCACHE (заменившей 16-разрядную программу SMARTDrive из DOS и Windows 3.1), что обеспечивает более высокую производительность файловой системы. Однако основное существенное улучшение новой файловой системы — это поддержка длинных имен файлов. Системы DOS и Windows 3.1 ограничивались стандартом “восемь-точка-три” при именовании файлов, поэтому добавление поддержки длинных имен файлов было приоритетной задачей, которую требовалось решить разработчикам Windows 95, тем более что пользователи операционных систем Macintosh и OS/2 уже всю применяли эти возможности.

Создатели Windows 95 должны были обеспечить обратную совместимость, т.е. необходимо было реализовать в файловой системе все новые свойства и, кроме того, не “обделит” пользователей предыдущих версий DOS и Windows. Кстати, обратная совместимость — одна из самых распространенных проблем в мире ПК.

В системе VFAT файлу или каталогу можно присваивать имя длиной до 255 символов (включая путь к этому файлу или каталогу). В Windows 95 от трехсимвольного расширения не отказались, поскольку в этой операционной системе (как и в предыдущих версиях Windows) с помощью расширения создается ассоциация типа “файл–приложение”. В длинных именах файлов можно использовать пробелы, а также символы + , ; = [], которые нельзя было использовать в стандартных (“восемь-точка-три”) именах файлов DOS.

При создании длинного имени файла создается его псевдоним, удовлетворяющий стандарту “восемь-точка-три”. Каким образом VFAT выполняет это в Windows 9x, описано ниже.

1. Первые три символа после последней точки в длинном имени файла становятся расширением псевдонима.
2. Первые шесть символов длинного имени файла (за исключением пробелов, которые игнорируются) преобразуются в символы верхнего регистра и становятся первыми шестью символами стандартного имени файла. Недопустимые в стандартном имени файла символы (+ , ; = []) преобразуются в символы подчеркивания.

3. VFAT добавляет символы ~1 (седьмой и восьмой) к псевдониму имени файла. Если первые шесть символов нескольких файлов одинаковы, то для разрешения конфликтов имен добавляются символы ~2, ~3 и т.д.

Длинные имена файлов в Windows NT/2000/XP

Обратите внимание, что в Windows NT/2000/XP псевдонимы имен файлов создаются иначе, чем в Windows 9x. Операционная система Windows NT/2000/XP использует для создания “короткого” имени файла первые шесть допустимых символов длинного имени и, если созданное имя уникально, добавляет символы ~1. Если же первые шесть символов уже используются другим файлом, то добавляются символы ~2. Для создания расширения Windows NT/2000/XP использует первые три допустимых символа после последней точки в длинном имени файла. Если после добавления символов ~5 появляется еще одно аналогичное “короткое” имя файла,

то для создания следующих имен файлов используется такой алгоритм: длинное имя файла преобразуется в четыре шестнадцатеричных символа, которые помещаются после двух допустимых символов длинного имени, и добавляются символы ~5. Таким образом, в Windows NT/2000/XP окончание ~5 появляется у всех псевдонимов файлов, а изменяются только шестнадцатеричные значения.

Совет

Вы можете отключить в системе VFAT механизм создания стандартного имени из длинного имени файла. Для этого в системном реестре добавьте двоичный параметр NameNumericTail в ветви `NKEY_LOCAL_MACHINE\ System\CurrentControlSet\Control\FileSystem` и установите его значение равным 0. Для возврата к использованию псевдонимов длинных имен файлов установите значение этого параметра равным 1.

Система VFAT хранит псевдонимы длинных имен в поле стандартных имен файлов записи каталога файлов. Таким образом, все версии DOS и Windows могут получить доступ к файлу под длинным именем с помощью его псевдонима. Остается еще одна проблема: как хранить 255 символов имени файла в 32 байтах записи каталога, ведь каждый символ имени файла — это один байт? Модифицировать структуру записи каталога нельзя, поскольку тогда предыдущие версии DOS не смогут использовать ее.

Разработчики файловой системы решили эту проблему следующим образом: были добавлены дополнительные записи каталога для хранения длинных имен файлов. Чтобы предыдущие версии DOS не повредили этих дополнительных записей каталога, VFAT устанавливает для них атрибуты, которые нельзя использовать для обычного файла: только для чтения, скрытый, системный и метка тома. Такие атрибуты DOS игнорирует, а следовательно, длинные имена файлов остаются “нетронутыми”.

Замечание

При использовании VFAT я рекомендую применять дисковые утилиты, которые поддерживают эту файловую систему. Windows 9x содержит необходимые программы для проверки, восстановления, дефрагментации диска и резервного копирования. Кстати, при запуске старых дисковых утилит в Windows 9x вы будете предупреждены о возможных последствиях.

Если необходимо использовать длинные имена файлов со старыми программами, установите программу `Lfnbk.exe` с компакт-диска Windows 9x. Эта программа восстанавливает длинные имена файлов, но только в том случае, если структура каталога не изменялась. Данный метод не рекомендуется применять, за исключением особых экстраординарных обстоятельств, однако он все же существует. Некоторые программы восстановления операционной системы (позволяющие восстановить содержимое жесткого диска без перезагрузки Windows) когда-то применяли эту функцию для восстановления диска Windows с длинными именами файлов из командной строки DOS (где поддерживаются только имена, соответствующие ограничению 8.3).

Еще одна проблема с длинными именами файлов состоит в том, что VFAT создает новый псевдоним всякий раз при создании или копировании файла в новый каталог. Например, файл `Expenses-January04.doc` сохраняется в папке под псевдонимом `EXPENS~1.DOC`. Если с помощью программы Проводник (Windows Explorer) в Windows 9x скопировать этот файл в папку, в которой уже существует файл `Expenses-December03.doc` с псевдонимом `EXPENS~1.DOC`, то VFAT создаст в этой папке для копируемого файла новый псевдоним `EXPENS~2.DOC`. Причем пользователь не будет уведомлен о таком “самоуправстве”. Для программ, поддерживающих длинные имена файлов, такое копирование не составляет проблемы: все длинные имена файлов сохраняются. Если же запустить приложение, которое не поддерживает длинных имен файлов, то, открыв файл `EXPENS~1.DOC`, пользователь обнаружит, что это файл `Expenses-December03.doc`, а не `Expenses-January04.doc`.

FAT 32

Это расширенная версия файловой системы FAT, поддержка которой была впервые реализована в Windows 95B (известной также как OEM Service Release 2), выпущенной в августе 1996 года. Файловая система FAT 32 поддерживается также Windows 98/Me и Windows 2000/XP. Оригинальная версия Windows 95 и различные версии Windows NT не поддерживают FAT 32.

Эта файловая система работает как стандартная FAT, но имеет отличия в организации хранения файлов. Кроме того, FAT 32 можно установить с помощью программы FDISK, в отличие от VFAT, которая является частью `Vmm.vxd`. Система FAT 32 была впервые реализована в Windows 95 OEM Service Release 2 (OSR2). Она встроена также в Windows 98/Me и в Windows 2000/XP.

Замечание

Поскольку FAT 32 устанавливается только с помощью программы FDISK, вы не сможете использовать ее на дискетах и дисках съемных устройств, которые не имеют файловой системы. Однако подобные устройства имеют структуру жестких дисков и на них можно устанавливать FAT 32.

Одной из основных причин создания FAT 32 была необходимость более эффективного использования дискового пространства. В этой файловой системе используются кластеры меньшего размера (для накопителей, емкость которых не превышает 8 GiB, размер кластеров составляет 4 KiB), что позволяет на 10–15% повысить эффективность использования дискового пространства по сравнению с большими накопителями FAT 16. Кроме того, FAT 32 поддерживает разделы емкостью до 2 TiB (в отличие от 2 GiB в FAT 16). Размеры кластеров FAT 32 приведены в табл. 25.13.

Таблица 25.13. Размеры кластеров FAT 32

Размер раздела	Количество секторов в каждом кластере	Размер кластера, КиБ
От 32,5 до 260 МиБ ¹	1	0,5
От 260 МиБ до 8 ГиБ	8	4
От 8 до 16 ГиБ	16	8
От 16 до 32 ГиБ	32	16
От 32 ГиБ до 8 ТиБ ²	64	32

¹ Разделы, размер которых не превышает 512 МиБ, по умолчанию имеют файловую систему FAT 16. Тем не менее можно изменить параметры форматирования и установить FAT 32.

² Операционные системы Windows 2000/XP форматируют только разделы FAT 32, размер которых не превышает 32 ГиБ; тем не менее они поддерживают существующие разделы FAT 32 емкостью до 2 ТиБ.

ТиБ (TiB) = тебибайт (Tebibyte) = 1 024 ГиБ = 1 099 511 627 776 байт

Судя по названию, в FAT 32 для управления кластерами (ячейками размещения) жесткого диска используются 32-разрядные числа, но фактически FAT 32 использует только первые 28 бит каждого 32-разрядного числа, оставляя зарезервированными 4 бита с высоким логиче-

ским уровнем. Значение битов меняется только при форматировании раздела, во время которого обнуляются все биты 32-разрядных чисел (в том числе и высшие биты). В дальнейшем 4 бита с высоким логическим уровнем при записи или чтении содержимого кластеров FAT 32 игнорируются; таким образом, эти биты сохраняются только в том случае, если не заполнены нулями. Компания Microsoft предусматривала только резервирование битов и никогда не сообщала об их использовании для других целей.

Таким образом, несмотря на то что данные технически являются 32-разрядными числами, в FAT 32 для управления кластерами жесткого диска используются 28-разрядные числа. В каждом кластере раздела FAT 32 содержится от 1 (512 байт) до 64 секторов (32 КиБ); 28-разрядные числа кластера располагаются в диапазоне от 0000000h до FFFFFFFh, или в десятичном исчислении от 0 до 268 435 455. Таким образом, общее число кластеров теоретически составляет 268 435 456. Тем не менее 11 чисел являются зарезервированными и не могут быть присвоены фактическим кластерам жесткого диска. Числа кластеров начинаются с 2 (0 и 1 зарезервированы), номер FFFFFFF7h зарезервирован для обозначения сбойного кластера, а числа от FFFFFFF8h до FFFFFFFFh в системе FAT указывают завершение цепочки. Следовательно, максимальное количество кластеров составляет 268 435 445 ($268\,435\,456 - 11 = 268\,435\,445$).

В разделах FAT 32 первые 32 сектора резервируются для загрузочной записи, которая включает в себя загрузочную запись, созданную по умолчанию (она начинается с логического сектора 0 и занимает три сектора), и резервную копию загрузочной записи (она начинается с логического сектора 6, но также занимает три сектора). Секторы, оставшиеся в этой области, зарезервированы и заполнены нулями (0). Сразу после 32 зарезервированных секторов находятся две таблицы размещения файлов FAT (таблица, заданная по умолчанию, и ее резервная копия), размеры которых могут составлять от 512 до 2 097 152 секторов, с областью данных от 65 525 до 268 435 445 кластеров.

Размер каждого кластера FAT 32 достигает 64 секторов (32 КиБ), поэтому диски или разделы FAT 32 теоретически могут содержать до 17 184 062 816 секторов (32 сектора для загрузочной записи + 2 097 152 сектора на каждую таблицу размещения файлов FAT × 2 FAT + 268 435 445 кластеров × 64 сектора на каждый кластер), что составляет 8,8 Тбайт, или 8 ТиБ. Эта величина является теоретической, потому что 32-разрядная нумерация секторов, используемая в таблицах разделов, расположенных в главной загрузочной записи (MBR), имеет ограничения по емкости диска, объем которого не должен превышать $4\,294\,967\,295 (2^{32} - 1)$ секторов, что составляет 2,2 Тбайт, или 2 ТиБ. Таким образом, хотя FAT 32 теоретически может обрабатывать до 8,8 Тбайт, в действительности из-за существующих ограничений формата таблицы разделов MBR емкость дисков не может превышать 2,2 Тбайт. С учетом скорости повышения емкости жестких дисков отдельные модели накопителей такого объема появятся примерно между 2009 и 2011 годами. Думаю, что к этому времени проблемы с ограничением MBR будут уже решены.

Отдельные файлы могут иметь размер чуть менее 4 ГиБ (без одного байта) и ограничены размерами поля в записи каталога, которая занимает 4 байта. Так как при нумерации кластеров используются не 16-разрядные, а 32-разрядные числа, то формат записей каталога в разделе FAT 32 может быть немного изменен. Двухбайтовое поле *Link to Start Cluster* увеличилось до 4 байт, для чего было использовано 2 из 10 байт (байты 12–21) в записи каталога, зарезервированных для будущего использования.

Замечание

Если вы попытаетесь отформатировать раздел FAT 32 емкостью более 32 ГиБ в системе с Windows 2000/XP, где-то на последней стадии процесс форматирования прервется и на экране появится сообщение об ошибке следующего содержания:

Logical Disk Manager: Volume size too big.

Это стандартное сообщение об ошибке. Инструменты форматирования, включенные в Windows 2000/XP, не позволяют форматировать разделы емкостью более 32 ГиБ, использующие FAT 32. Эти ограничения

относятся только к средствам форматирования; операционные системы полностью поддерживают существующие разделы, емкость которых не превышает 2 ТиБ. Таким образом Microsoft пытается сделать все для того, чтобы NTFS использовалась на любых недавно созданных разделах, объем которых превышает 32 ГиБ. При форматировании внешних накопителей USB или FireWire, которые будут подключены к различным системам, в том числе к Windows 98/Me, необходимо отформатировать жесткий диск в системе Windows 98/Me.

Операционные системы Windows 95 OSR2 и Windows 98 имеют дополнительные ограничения по использованию FAT 32. Программа ScanDisk, поставляемая в комплекте с этими операционными системами, является 16-разрядным приложением, объем пространства размещения для единичного 16-мегабайтового блока памяти которого составляет менее 64 Кбайт. Это означает, что ScanDisk не может обрабатывать разделы с файловой системой FAT 32, которая использует таблицу размещения файлов (FAT) размером менее 64 Кбайт. Для размещения любого элемента FAT 32 используется 4 байта, поэтому ScanDisk не может обрабатывать FAT в разделе с файловой системой FAT 32, определяющей более 4 177 920 кластеров (фактический объем раздела после вычитания двух зарезервированных кластеров составляет 4 177 918 кластеров). С учетом того что на каждый кластер приходится 32 КиБ данных, включая зарезервированную область загрузочного сектора и саму таблицу размещения файлов (FAT), получаем максимальный размер раздела, равный 136,94 Гбайт, или 127,53 ГиБ. В Windows Me и более новых версиях подобного ограничения не существует.

Ограничения по емкости разделов для файловой системы FAT 32 приведены в табл. 25.14.

Таблица 25.14. Ограничения емкости разделов для FAT 32

Ограничение емкости	Количество кластеров	Количество секторов в кластере	Общее количество секторов	Емкость раздела (в десятичной системе)	Емкость раздела (в двоичной системе)
Минимальный размер	65 535	1	66 601	34,1 Мбайт	32,52 МиБ
Максимальный размер для Win 9x	4 177 918	64	267 452 072	136,94 Гбайт	127,53 ГиБ
Максимальный объем MBR	67 092 481	64	4 294 967 266	2,2 Тбайт	2 ТиБ
Теоретический максимальный размер	268 435 445	64	17 184 062 816	8,8 Тбайт	8 ТиБ

Файловая система FAT 32 более функциональна, чем FAT 12 или FAT 16, в которых первый сектор раздела используется для размещения загрузочной записи тома, имеющей важнейшее значение. Повреждение или разрушение загрузочного сектора приводит к потере доступа ко всем данным, которые хранятся в этом разделе. В FAT 32 в первых 32 секторах раздела находится загрузочная запись раздела, создаваемая по умолчанию, и ее резервная копия. Каждая загрузочная запись раздела FAT 32 занимает три сектора. Загрузочная запись раздела находится в логических секторах 0–2 (в первых трех секторах), а ее резервная копия — в секторах 6–8. Эта особенность неоднократно спасала меня при повреждении базовой загрузочной записи, что приводило к потере доступа ко всему разделу. Мне удавалось в подобных ситуациях восстановить весь раздел, заменяя поврежденную загрузочную запись ее резервной копией с помощью редактора секторов, в частности программы Norton Diskedit, входящей в пакет Norton SystemWorks от компании Symantec. Поскольку FAT 12 и FAT 16 не позволяют создавать резервную копию загрузочного сектора, при его повреждении придется вручную воссоздавать загрузочный сектор с самого начала, что сделать значительно труднее.

Существует еще одно отличие FAT 32 от ее предшественниц — положение корневого каталога: он не занимает фиксированного места на диске, как в FAT 16. Корневой каталог в FAT 32 может располагаться в любом месте раздела и иметь любой размер. Устранение ограничений записей корневого каталога обеспечивает динамичное изменение размера раздела FAT 32. Однако Microsoft не реализовала это замечательное свойство в операционных систе-

мах Windows 9x, чем и воспользовались независимые разработчики, например компания PowerQuest, создавшая программу PartitionMagic.

Файловая система FAT 32, как и FAT 12/16, поддерживает две копии таблицы размещения файлов (FAT) и, когда сектор в заданной по умолчанию таблице становится нечитабельным, автоматически переключается на резервную таблицу FAT. В системе FAT 16 корневой каталог занимает ровно 32 сектора и располагается сразу за двумя копиями таблицы FAT. В свою очередь, в FAT 32 корневой каталог фактически создается в виде подкаталога (папки), который сохраняется точно так же, как и обычный файл, может находиться в любом месте жесткого диска и иметь переменную длину. Таким образом, в корневом каталоге FAT 32 не существует 512-байтового ограничения, свойственного файловым системам FAT 12/16.

Основной недостаток FAT 32 — несовместимость с предыдущими версиями DOS и Windows 95. Вы не сможете загрузить предыдущую версию DOS или оригинальную Windows 95 с диска с файловой системой FAT 32; кроме того, раздел с FAT 32 будет недоступен этим системам при их загрузке с другого диска.

Файловая система FAT 32 обладает перечисленными ниже характеристиками.

- FAT 32 полностью поддерживается операционными системами Windows 95B (OSR2)/98/Me/2000/XP и более новыми версиями.
- Операционные системы MS-DOS 6.22 и более ранние версии, Windows 95a и Windows NT не поддерживают FAT 32 и не могут считывать или записывать разделы FAT 32.
- Операционная система Mac OS 8.1 и ее более поздние версии поддерживают жесткие диски FAT 32.
- FAT 32 является идеальным форматом для внешних накопителей USB и FireWire, которые подключаются к различным моделям PC и Macintosh.
- Корневой каталог (папка) сохраняется в виде обычного файла, который может быть расположен в любом месте жесткого диска. Подкаталоги (папки), в том числе и корневой каталог, могут обрабатывать до 65 534 записей. Если в записях используются длинные файловые имена, количество записей заметно сокращается.
- В FAT 32 используют кластеры меньшего размера (4 KiB для разделов емкостью до 8 GiB), что приводит к более эффективному, чем в FAT 16, распределению дискового пространства.
- Создается резервная копия критического загрузочного сектора, которая хранится в логическом секторе 6 того же раздела.
- Операционные системы Windows 2000/XP форматируют только те разделы FAT 32, емкость которых не превышает 32 GiB. Для форматирования разделов FAT 32 большей емкости (т.е. более 32 GiB) придется отформатировать разделы в Windows 98/Me.
- При появлении проблем, связанных с загрузочным сектором диска, следует загрузить компьютер с помощью загрузочной дискеты, записанной в Windows 95B (OSR2)/98/Me/2000/XP. Существуют программные средства сторонних разработчиков, с помощью которых можно восстановить данные, хранящиеся в разделах FAT 32.
- FAT 32 не имеет встроенных функций, обеспечивающих безопасность, шифрование или сжатие данных.
- Размеры файлов в разделах FAT 32 не могут превышать $4\,294\,967\,295$ байт ($2^{32} - 1$), что на 1 байт меньше 4 GiB.

Зеркальная копия файловой системы

Система FAT 32 также использует преимущества двух копий FAT в разделе диска. Как и в FAT 16, в FAT 32 первая копия является основной и периодически копирует данные в дополнительную копию FAT. Если в FAT 32 появляются проблемы с главной копией FAT, система пе-

реключается на дополнительную копию, которая становится главной. Помимо этого, система прерывает процесс создания зеркальной копии FAT, чтобы предотвратить потерю данных.

Создание раздела FAT 32

Для создания раздела с FAT 32 в Windows 9x необходимо использовать программу FDISK в командной строке так же, как при создании раздела с FAT 16. При запуске этой программы будет выполнено тестирование диска и, если его размер превышает 512 Мбайт, появится приведенное ниже сообщение.

Компьютер имеет диск емкостью более 512 Мбайт. Данная версия Windows включает поддержку больших дисков и позволяет эффективнее использовать место на таких дисках, а также форматировать диски размером более 2 Гбайт как один диск.

ВНИМАНИЕ! Если включить поддержку больших дисков и создать на них новый диск, невозможно будет получить доступ к новому диску из другой операционной системы, включая некоторые версии Windows 95 и Windows NT, а также более ранние версии Windows и MS-DOS. Кроме того, дисковые служебные программы, которые не поддерживают явно файловую систему FAT32, не смогут работать с этим диском. Если собираетесь обращаться к этому диску из других операционных систем или более старых служебных программ, не включайте поддержку больших дисков.

Включить поддержку больших дисков (Y/N)? [N]

Если вы ответите на этот вопрос утвердительно, все разделы размером более 512 Мбайт будут иметь файловую систему FAT 32. Кроме того, утвердительный ответ требуется для создания раздела размером более 2 Гбайт. Последующие окна работы программы FDISK аналогичны окнам предыдущих версий этой программы.

Программа FDISK автоматически определяет размер кластера на основе выбранной файловой системы и размера раздела. Однако существует недокументированный параметр команды Format, позволяющий явно указать размер кластера: Format / Z:n, где *n* — размер кластера в байтах, кратный 512. С помощью этой команды вы можете создать файловую систему с размером кластера, меньшим установленного по умолчанию.

Внимание!

При использовании переключателя /z необходимо помнить об ограничении FAT 16 на количество кластеров (65 536). Используйте этот переключатель с файловой системой FAT 32. Не забывайте, что модификация размера кластера может отразиться на производительности файловой системы.

Преобразование FAT 16 в FAT 32

В Windows 98 и более новых версиях существует программа-мастер для преобразования раздела в FAT 32 без потери данных. При запуске программа преобразования диска отображает информацию о существующих разделах и установленных файловых системах. Вам необходимо лишь выделить диск и выполнить все операции мастера.

Поскольку преобразование затрагивает данные существующих разделов и создание загрузочных записей нового тома, параметров FAT и кластеров, эта процедура может оказаться более длительной, чем при разделе и форматировании пустого диска. В зависимости от объема содержащихся на диске данных и размера кластера, преобразование может продолжаться несколько часов.

Обратите внимание, что после преобразования диска в FAT 32 выполнить обратное преобразование нельзя. Необходимо принимать “радикальные” меры, т.е. сохранить данные, запустить программу FDISK, удалить раздел с FAT 32 и заново создать раздел с FAT 16.

Замечание

Операционная система Windows содержит базовые инструменты для создания разделов с файловой системой FAT 32. Тем не менее компания VCOM создала программу Partition Commander (www.v-com.com), а PowerQuest — программу PartitionMagic, которые обладают более широкими возможностями работы с файловыми системами. С помощью этих программ можно выполнить преобразование FAT 16 в FAT 32 и обратно, а также изменить размеры разделов без потери данных.

Основные сведения о FAT

Электронная таблица FAT предназначена для управления распределением дискового пространства. Каждая ее ячейка связана с определенным кластером на диске. Число, содержащееся в ячейке, сообщает, использован ли данный кластер под какой-либо файл, и если использован, то указывает, где находится следующий кластер этого файла. Каждая ячейка FAT хранит шестнадцатеричное значение длиной 12 или 16 бит. Шестнадцатиразрядные FAT более удобны в работе, так как значительно легче редактировать поля размером в два байта, чем в полтора. Чтобы самостоятельно отредактировать FAT, необходимо выполнить некоторые математические преобразования для получения номера кластера. К счастью, многие программы позволяют отредактировать FAT автоматически. В большинстве этих программ номера кластеров представлены в десятичном виде, наиболее удобном для пользователей. В табл. 25.15 приведены данные о каталоге и FAT (файл не фрагментирован).

Таблица 25.15. Записи файлов в FAT 16

Каталог		
Имя	Начальный кластер	Размер
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значение	Назначение
00002	0	Первый доступный кластер
...
00999	0	Кластер доступен
01000	1001	Используется; ссылка на следующий кластер
01001	1002	Используется; ссылка на следующий кластер
01002	1003	Используется; ссылка на следующий кластер
01003	FFFFh	Конец файла
01004	0	Кластер доступен
01005	0	Кластер доступен
...
65526	0	Последний доступный кластер

В данном примере запись каталога указывает начальный кластер (1000), в котором размещается файл. В FAT кластеры с ненулевыми значениями используются, а специальное значение указывает дальнейшее расположение файла. В рассматриваемом примере в кластере 1000 указывается кластер 1001, в 1001 — 1002, в 1002 — 1003, а в 1003 записано значение FFFFh, т.е. на этом кластере файл заканчивается.

Рассмотрим пример с фрагментированным файлом. Пусть файл Usconst.txt записан, начиная с кластера 1000, а файл Pledge.txt начинается с кластера 1002. Таким образом, файл Usconst.txt становится фрагментированным. Описанная ситуация иллюстрируется данными, приведенными в табл. 25.16.

В данном примере в файл Usconst.txt “внедряется” файл Pledge.txt, что приводит к непоследовательному расположению файлов на диске, т.е. фрагментации. В системах DOS и Windows есть программы дефрагментации, которые перемещают файлы для их последовательного размещения на диске.

Таблица 25.16. Записи о фрагментированном файле в каталоге и FAT

Каталог		
Имя	Начальный кластер	Размер
Pledge.txt	1002	2
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значение	Назначение
00002	0	Первый доступный кластер
...
00999	0	Кластер доступен
01000	1001	Используется; ссылка на следующий кластер
01001	1004	Используется; ссылка на следующий кластер
01002	1003	Используется; ссылка на следующий кластер
01003	FFFFh	Конец файла
01004	1005	Используется; ссылка на следующий кластер
01005	FFFFh	Конец файла
...
65526	0	Последний доступный кластер

Первые две записи FAT зарезервированы и содержат информацию о самой FAT, все остальные указывают на соответствующие кластеры диска. Большинство записей FAT состоят из ссылок на кластеры, в которых содержатся части определенного файла, а некоторые включают специальные шестнадцатеричные значения:

- 0000h — кластер не используется;
- FFF7h — как минимум, один сектор в кластере поврежден и не может быть использован для хранения данных;
- FFF8h–FFFFh — кластер содержит конец файла.

Для сравнения FAT 16 и FAT 32 необходимо посмотреть, как в этих файловых системах организовано хранение данных. Номера кластеров в FAT 16 хранятся в виде 16-разрядных записей (0000h–FFFFh). Максимальное значение FFFFh соответствует десятичному 65 536, но несколько значений зарезервированы для специальных целей. Реальное число кластеров в FAT 16 лежит в диапазоне 0002h–FFF6h, или 2–65 526. Таким образом, для хранения файлов используется 65 524 кластера. В FAT 32 количество кластеров лежит в диапазоне 00000000h–FFFFFFFFh, или 0–4 294 967 295. Как и в FAT 16, верхние и нижние кластеры зарезервированы для специальных целей и их номера находятся в диапазоне 00000002h–FFFFFFFF6h, или 2–4 294 967 286. Таким образом, для хранения файлов можно использовать 4 294 967 284 кластера. Накопитель на жестких дисках разбит на большее количество кластеров, каждый из которых становится меньше, что снижает потери дискового пространства. Пример записей о файле в FAT 32 приведен в табл. 25.17.

Таблица 25.17. Записи файлов в системе FAT 32

Каталог		
Имя	Начальный кластер	Размер
Usconst.txt	1000	8
FAT 32		
Номер кластера	Значение	Назначение
0000000002	0	Первый доступный кластер
...
0000000999	0	Кластер доступен
0000001000	1001	Используется; ссылка на следующий кластер

FAT 32		
Номер кластера	Значение	Назначение
0000001001	1002	Используется; ссылка на следующий кластер
0000001002	1003	Используется; ссылка на следующий кластер
0000001003	1004	Используется; ссылка на следующий кластер
0000001004	1005	Используется; ссылка на следующий кластер
0000001005	1006	Используется; ссылка на следующий кластер
0000001006	1007	Используется; ссылка на следующий кластер
0000001007	0FFFFFFFh	Конец файла
0000001008	0	Кластер доступен
...
268435446	0	Последний доступный кластер

Поскольку раздел FAT 32 имеет больше кластеров, чем раздел FAT 16, размер кластера уменьшается. Использование меньшего кластера снижает потери дискового пространства.

Некоторые дополнительные ограничения характерны и для томов FAT 32. Для томов объемом меньше 512 MiB по умолчанию задается файловая система FAT 16, однако с помощью специальных команд файловую систему FAT 32 можно указать для томов объемом минимум 32,52 MiB. Однако для томов еще меньшего объема доступны только FAT 16 или FAT 12.

Уменьшение размера кластера приводит к увеличению записей в FAT. Раздел размером 2 Гбайт с FAT 32 использует 524 288 записей, в то время как аналогичный раздел с FAT 16 использует 65 536 записей. Следовательно, таблица FAT 16 имеет размер 128 Кбайт (65 536 записей \times 16 бит = 1 048 576 бит, или 131 072 байт, или 128 Кбайт), а таблица FAT 32 — 2 Мбайт.

Размер FAT существенно влияет на производительность файловой системы. В Windows 9x/Me модуль VCACHE пытается загрузить FAT в оперативную память для повышения производительности системы. Выбор кластера размером 4 Кбайт на дисках емкостью до 8 Гбайт обеспечивает компромисс между производительностью и размером FAT в оперативной памяти.

Несмотря на то что размер FAT в файловой системе FAT 32 практически в 20 раз больше, чем в FAT 16, появляется незначительный (менее 5%) прирост производительности FAT 32 в Windows. Это отчасти достигается использованием самых современных накопителей на жестких дисках.

Тип используемых FAT определяется программой создания разделов, хотя записываются они в процессе форматирования высокого уровня программой Format. На всех дискетах применяется 12-разрядная FAT, а на жестком диске может использоваться как 12-, так и 16-разрядная FAT, в зависимости от размера логического диска. На дисках размером меньше 16 Мбайт (32 768 секторов) применяется 12-разрядная FAT, на дисках большего размера — 16-разрядная, а на дисках размером более 512 Мбайт при использовании Windows 95 OSR2 и Windows 98 — 32-разрядная FAT. При работе с программой управления жесткими дисками в Windows 2000/XP также можно выбрать FAT 32.

Программа создания разделов обычно размещает на одном диске две копии FAT. Каждая копия занимает несколько последовательных секторов на диске, и вторая копия записывается непосредственно после первой. К сожалению, операционная система использует вторую копию FAT только в том случае, когда невозможно прочитать секторы, содержащие первую копию. Таким образом, если первая копия FAT пропадет (весьма распространенная ситуация), операционная система не будет использовать вторую копию. Даже команда Chkdsk не проверяет вторую копию FAT. Кроме того, каждый раз, когда операционная система обновляет первую копию FAT, большие участки первой копии автоматически копируются во вторую. Если же первая копия повреждена, то и вторая окажется поврежденной: после обновления FAT вторая копия отражает все изменения в первой копии, включая и ошибки. Обе копии FAT редко отличаются одна от другой, по крайней мере в течение продолжительного срока: при обновлении первая копия

FAT автоматически копируется во вторую. Учитывая все это, можно сказать, что применение второй копии FAT ограничивается только операциями по восстановлению дефектных данных. Но даже в такой ситуации использовать вторую копию FAT можно только в том случае, когда проблема решается *немедленно*, не дожидаясь очередного обновления FAT.

Каталоги

Каталог — это база данных, содержащая информацию о записанных на диске файлах. Каждая запись в ней имеет длину 32 байт, и между записями не должно быть никаких разделителей. В каталоге сохраняется практически вся информация о файле, которой располагает операционная система.

- *Имя файла и расширение* — восемь символов имени и три символа расширения; точка между именем и расширением файла подразумевается, но не включается в эту запись.

Замечание

В Windows 9x/Me имя файла может состоять из 255 символов в структуре каталога 8.3.

- *Байт атрибутов файла*, содержащий флаг, который представляет стандартные атрибуты файла.
- *Время и дата* создания файла или его модификации.
- *Размер файла* в байтах.
- *Ссылка на начальный кластер* — номер кластера, с которого начинается файл.

Информация о расположении файла, т.е. расположении оставшихся кластеров, содержится в FAT.

Существует два основных типа каталогов: *корневой каталог* и *подкаталог*. Различаются они максимальным количеством хранящихся файлов. На каждом логическом диске в фиксированном месте, сразу же за копиями FAT, располагается корневой каталог. Размеры корневых каталогов варьируются в зависимости от размера диска, но каждый конкретный корневой каталог имеет *фиксированное* максимальное число файлов. Длина корневого каталога фиксируется при создании логического диска и не может быть изменена в процессе работы. Размер корневого каталога различных накопителей приведен в табл. 25.18. В отличие от корневого каталога, подкаталог может хранить произвольное количество файлов и расширяться по мере необходимости.

Таблица 25.18. Размер корневого каталога

Тип накопителя	Максимальное количество записей
Жесткий диск	512
Дисковод 1,44 Мбайт	224
Дисковод 2,88 Мбайт	448
Jaz и Zip	512
LS-120/LS-240	512

Замечание

Одно из преимуществ FAT 32 заключается в том, что корневой каталог может располагаться в любом месте диска и содержать неограниченное количество записей.

Все каталоги имеют одинаковую структуру. Записи в этой базе данных сохраняют важную информацию о файлах, которая связана с информацией, хранящейся в FAT, посредством одного из полей записи — номера первого занимаемого файлом кластера на диске. Если бы все файлы на диске не превышали размеров одного кластера, потребности в FAT вообще бы не возникло. В FAT содержится информация о файле, отсутствующая в каталоге, — номера кластеров, в которых расположен *весь* файл.

Чтобы отследить расположение всего файла на диске, обратитесь к каталогу и выясните номер первого кластера и длину файла. Затем, используя таблицу размещения файлов, просмотрите *цепочку* кластеров, занимаемых файлом, пока не дойдете до конца файла.

Формат 32-байтовой записи в каталоге приведен в табл. 25.19.

Таблица 25.19. Формат каталога

Смещение			
Hex	Dec	Длина поля	Описание
00h	0	8 байт	Имя файла
08h	8	3 байт	Расширение файла
0Bh	11	1 байт	Атрибуты файла
0Ch	12	10 байт	Зарезервировано (00h)
16h	22	Одно слово	Время создания
18h	24	Одно слово	Дата создания
1Ah	26	Одно слово	Начальный кластер
1Ch	28	Одно двойное слово	Размер файла в байтах

Примечание. Слово соответствует двум байтам в обратном порядке, двойное слово — двум словам в обратном порядке.

Имена файлов и их расширения записаны с привязкой к левому краю и дополнены до максимальной длины пробелами, т.е. имя файла *AL* будет реально сохранено как *AL.....*, где точки обозначают пробелы. Первый байт имени файла также может обозначать его состояние (табл. 25.20).

Таблица 25.20. Байт состояния записи каталога (первый байт)

Hex	Состояние файла
00h	Запись никогда не использовалась; ниже этой записи поиск не выполняется
05h	Первый символ имени файла в настоящее время — E5h
E5h	Файл удален
2Eh	Точка (.) показывает, что запись является каталогом. Если и второй байт — 2Eh, то поле начального кластера содержит номер кластера родительского каталога (0000h, если родительский каталог корневой)

В табл. 25.21 приведены используемые в записях каталогов атрибуты файлов.

Таблица 25.21. Атрибуты файлов

Позиция бита в шестнадцатеричном формате									Значение	Описание
7	6	5	4	3	2	1	0			
0	0	0	0	0	0	0	1	01h	Только для чтения	
0	0	0	0	0	0	1	0	02h	Скрытый	
0	0	0	0	0	1	0	0	04h	Системный	
0	0	0	0	1	0	0	0	08h	Метка тома	
0	0	0	1	0	0	0	0	10h	Подкаталог	
0	0	1	0	0	0	0	0	20h	Архивный (измененный)	
0	1	0	0	0	0	0	0	40h	Зарезервировано	
1	0	0	0	0	0	0	0	80h	Зарезервировано	
Примеры										
0	0	0	0	0	1	1	1	07h	Системный, скрытый, только для чтения	
0	0	1	0	0	0	0	1	21h	Только для чтения, архивный	
0	0	1	1	0	0	1	0	32h	Скрытый, подкаталог, архивный	
0	0	1	0	0	1	1	1	27h	Только для чтения, скрытый, системный, архивный	

Замечание

Однако вы можете установить FAT-атрибуты файлов в NTFS с помощью стандартных инструментов Windows NT/2000, например программы Windows NT Explorer, или команды DOS `Attrib`. При копировании файлов из раздела NTFS в FAT все атрибуты файла сохраняются, и пользователь с правами полного доступа не сможет удалить файл с атрибутом “только для чтения”.

Ошибки файловой системы FAT

Ошибки в файловой системе появляются скорее из-за программных, нежели из-за аппаратных сбоев (например, при неверном завершении работы Windows). Некоторые программные ошибки описаны ниже.

Потерянные кластеры

Это наиболее распространенная ошибка файловой системы, при которой кластеры в FAT помечаются как используемые, хотя на самом деле таковыми не являются. Потерянные кластеры появляются при неверном завершении работы приложения или крахе системы. Программы восстановления диска могут обнаружить эти кластеры и восстановить их.

Потерянные кластеры появляются в файловой структуре (табл. 25.22).

Таблица 25.22. Потерянные кластеры в файловой структуре

Каталог		
Имя	Начальный кластер	Размер
(Нет записей)	0	0
FAT 16		
Номер кластера	Значение	Назначение
00002	0	Первый доступный кластер
...
00999	0	Кластер доступен
01000	1001	Используется; ссылка на следующий кластер
01001	1002	Используется; ссылка на следующий кластер
01002	1003	Используется; ссылка на следующий кластер
01003	FFFFh	Конец файла
01004	0	Кластер доступен
...
65526	0	Последний доступный кластер

Операционной системой воспринимается действительная цепочка кластеров FAT, но не соответствующая запись каталога, поскольку выполнение программы было остановлено до сохранения ее открытых рабочих файлов. Операционная система обычно модифицирует цепочку FAT при записи файлов, а после закрытия программы создает запись каталога. Если система прерывает работу перед закрытием файла (например, внезапно выключается компьютер), то появляются потерянные кластеры. Программы администрирования жестких дисков отслеживают цепочку FAT для каждого файла и подкаталога в разделе, после чего воссоздают в памяти точную копию FAT. После компиляции списка всех записей FAT, указывающих на правильное расположение кластеров, программа сравнивает копию с реальной структурой FAT. Записи копии, не указывающие на текущие расположенные кластеры FAT, являются потерянными кластерами, поскольку не входят в действительную цепочку FAT.

Программы восстановления диска просматривают диск и создают копию FAT в оперативной памяти. Затем эта копия сравнивается с “настоящей” FAT, и таким образом выявляются потерянные кластеры, т.е. не принадлежащие ни одному из существующих файлов. Практически все программы восстановления могут сохранять информацию из потерянных кластеров в файл, а затем обнулять их.

Например, программа Chkdsk или Scandisk из цепочек потерянных кластеров создает файлы с именами FILE0001.CHK, FILE0002.CHK и т.д. Программа Chkdsk/Scandisk преобразует потерянные кластеры в файлы (табл. 25.23).

Таблица 25.23. Поиск потерянных кластеров

Каталог		
Имя	Начальный кластер	Размер
FILE0001.CHK	1000	4
FAT 16		
Номер кластера	Значение	Назначение
00002	0	Первый доступный кластер
...
00999	0	Кластер доступен
01000	1001	Используется; ссылка на следующий кластер
01001	1002	Используется; ссылка на следующий кластер
01002	1003	Используется; ссылка на следующий кластер
01003	FFFFh	Конец файла
01004	0	Кластер доступен
...
65526	0	Последний доступный кластер

Как видно из приведенного примера, оригинальное имя файла не восстанавливается. Однако его можно восстановить, просмотрев содержимое файлов, которые созданы программой восстановления диска.

Пересекающиеся файлы

Такие файлы появляются, когда две записи каталога неправильно указывают на один кластер. В результате кластер “содержит” данные из нескольких файлов, что, естественно, недопустимо.

В табл. 25.24 приведен пример записи файловой системы с пересекающимися файлами.

Таблица 25.24. Пересекающиеся файлы

Каталог		
Имя	Начальный кластер	Размер
Usconst.txt	1000	4
Pledge.txt	1002	2
FAT 16		
Номер кластера	Значение	Назначение
00002	0	Первый доступный кластер
...
00999	0	Кластер доступен
01000	1001	Используется; ссылка на следующий кластер
01001	1002	Используется; ссылка на следующий кластер
01002	1003	Используется; ссылка на следующий кластер
01003	FFFFh	Конец файла
01004	0	Кластер доступен
...
65526	0	Последний доступный кластер

В рассматриваемом примере два файла занимают два кластера — 1002 и 1003. Это пересечение файлов начинается с кластера 1002. Чаще всего один из пересекающихся файлов поврежден. Программы восстановления данных обычно решают проблему пересекающихся

файлов следующим образом: файлы копируются с новыми именами в свободное место диска, а пересекающаяся область *обоих* файлов (и их остальные части) удаляется. Обратите внимание, что удаляются оба файла, т.е. устранение подобной ошибки не порождает новых проблем: например, запись в каталоге указывает на несуществующий файл. Просмотрев два восстановленных файла, можно определить, какой из них поврежден.

Для программ восстановления диска поиск пересекающихся файлов — очень простая задача, и практически все дисковые утилиты могут устранить эту проблему.

Неверный файл или каталог

Иногда информация в записи каталога для файла или подкаталога не соответствует действительности: запись содержит кластер с неверной датой или неправильным форматом. Практически все программы восстановления диска устраняют и эту проблему.

Ошибки FAT

Как уже отмечалось выше, при повреждении основной FAT доступ к файлам осуществляется с помощью дополнительной FAT. Программы восстановления диска возвращают поврежденную FAT в ее оригинальное местоположение и активизируют зеркальное копирование. FAT 32 обладает большими способностями к восстановлению, поскольку в ней используются более развитые средства зеркального копирования.

Пример поврежденной FAT приведен в табл. 25.25.

Таблица 25.25. Поврежденная FAT

Каталог		
Имя	Начальный кластер	Размер
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значение	Назначение
00002	0	Первый доступный кластер
...
00999	0	Кластер доступен
01000	1001	Используется; ссылка на следующий кластер
01001	0	Кластер доступен
01002	1003	Используется; ссылка на следующий кластер
01003	FFFFh	Конец файла
01004	0	Кластер доступен
...
65526	0	Последний доступный кластер

В рассматриваемом примере размер файла в каталоге не соответствует количеству кластеров в FAT (потерян кластер 1001), а кроме того, кластеры 1002 и 1003 являются потерянными. При восстановлении данных поврежденная FAT восстанавливается из резервной копии. Практически все программы восстановления данных успешно справляются с таким типом ошибки FAT. Обычные утилиты сокращают объем файла и создают запись для второго файла, состоящую из утерянных кластеров. Затем уже пользователю придется разобраться, какому файлу принадлежат найденные кластеры; в данном случае знание технологий восстановления данных поможет там, где не справляются автоматические утилиты.

NTFS

Операционная система Windows NT 3.1 (несмотря на обозначение 3.1, это первая версия NT), выпущенная в августе 1993 года, продемонстрировала файловую систему новой технологии (New Technology File System — NTFS), которая предназначена только для операцион-

ных систем, созданных на основе NT (включая Windows 2000/XP), и не поддерживается Windows 9x/Me. Она включает в себя множество новых возможностей, не поддерживаемых в файловых системах FAT.

В отличие от FAT, NTFS поддерживает разделы большого объема (до 16 TiB), большие файлы и количество файлов, хранящихся в разделе. Кроме того, в NTFS используются также кластеры меньшего размера, чем в FAT 32, что позволяет более эффективно использовать дисковое пространство. Например, в разделе NTFS объемом 30 GiB используются кластеры размером 4 KiB, тогда как в разделе такого же объема, отформатированном в FAT 32, используются кластеры размером 16 KiB. Использование кластеров меньшего размера позволяет снизить потери дискового пространства. Размеры кластеров NTFS приведены в табл. 25.26.

Таблица 25.26. Размеры кластеров NTFS

Размер раздела	Количество секторов в каждом кластере	Размер кластера
От 16 до 512 MiB	1	0,5 KiB
От 512 MiB до 1 GiB	2	1 KiB
От 1 до 2 GiB	4	2 KiB
От 2 GiB до 2 TiB*	8	4 KiB

**Изменение параметров форматирования позволяет увеличить размер кластеров; однако использование кластеров размером более 8 секторов (4 KiB) приводит к блокированию функции сжатия файлов.*

В NTFS используется специальная файловая структура, которая называется главной файловой таблицей (Master File Table — MFT), и файлы метаданных. В сущности, MFT представляет собой реляционную базу данных, состоящую из строк и столбцов, в которых содержатся записи и атрибуты файлов. Эта таблица содержит записи практически всех файлов, расположенных в разделе NTFS. Файловая система NTFS создает записи файлов и каталогов для каждого файла или каталога, созданного в разделе NTFS. Эти записи хранятся в MFT, причем каждая из них занимает 1 KiB. Записи файлов содержат данные о местоположении записи в MFT, а также атрибуты файлов и другую информацию, относящуюся к этим файлам.

Файловая система NTFS была предназначена для управления кластерами с помощью 64-разрядных чисел, представляющих собой астрономические величины, но в существующих версиях используются только 32-разрядные числа. Использование 32-разрядных чисел позволяет обеспечить адресацию до 4 294 967 295 кластеров, каждый из которых обычно занимает 4 KiB.

В общей сложности NTFS резервирует 32 сектора, 16 из которых занимает созданный по умолчанию загрузочный сектор раздела, а следующие 16 — его резервная копия. Загрузочный сектор раздела, созданный по умолчанию, размещается в начале раздела (в логическом секторе 0), тогда как его резервная копия записывается либо в логическом центре (если раздел был отформатирован с помощью NT 3.51 или более ранней версии), либо в конце тома раздела (если последний был отформатирован с помощью операционной системы NT 4.0 или более поздней версии, включая Windows 2000/XP).

Следовательно, раздел NTFS может содержать в общей сложности до 34 359 738 392 секторов (32 сектора, зарезервированные для базового загрузочного сектора раздела и его резервной копии, плюс 4 294 967 295 кластеров \times 8 секторов), что составляет 17,59 Тбайт, или 16 TiB. Следует заметить, что полученное значение является теоретическим, так как 32-разрядная нумерация секторов, используемая в таблицах разделов, размещенных в главной загрузочной записи (MBR), имеет ограничения по емкости диска. Поэтому максимальная емкость диска не может превышать 4 294 967 295 ($2^{32} - 1$) секторов, что составляет 2,2 Тбайт, или 2 TiB. Таким образом, несмотря на то что NTFS теоретически может обрабатывать разделы объемом до 17,59 Тбайт, из-за ограничений, налагаемых форматом таблицы разделов MBR, эта величина уменьшается до 2,2 Тбайт.

Операционные системы Windows 2000/XP позволяют обойти это ограничение, используя в незагрузочных накопителях новый формат хранения данных, который называется *динами-*

ческим диском. Операционные системы Windows 2000 и Windows XP Professional (но не XP Home) предлагают два метода хранения данных: основные и динамические диски. В *основных дисках* используются те же файловые структуры, что и ранее. На диске находится главная загрузочная запись (MBR), содержащая таблицу разделов, из-за ограничений которой можно создавать не более четырех первичных разделов на жестком диске или трех первичных разделов и одного расширенного раздела с неограниченными логическими дисковымидами. Первичные разделы и логические дисководы основных дисков называются *основными разделами*.

Динамические диски, впервые появившиеся в Windows 2000, обеспечивают возможность создания динамических разделов, которые могут быть простыми (использующими только один диск), составными (в которых используется несколько накопителей) или чередующимися (для повышения производительности одновременно используется несколько накопителей). В целях отслеживания информации о динамических разделах, содержащихся на диске, и динамических дисках, имеющихся в компьютере, в динамических дисках используется скрытая база данных, которая содержится в последнем мегабайте диска. Все динамические диски, имеющиеся в компьютере, содержат точную копию базы данных, что позволяет использовать базу данных одного динамического диска для восстановления поврежденной базы данных другого диска. Применяя несколько составных или чередующихся накопителей с динамическим форматом, можно преодолеть ограничение в 2,2 Тбайт, свойственное разделам, содержащим только одну главную загрузочную запись (MBR).

Как уже отмечалось, 32-разрядная нумерация секторов в таблицах разделов на дисках MBR не позволяет создавать основные диски NTFS емкостью более 2 ТиБ. Тем не менее использование нескольких составных или чередующихся накопителей дает возможность создавать разделы NTFS более высокой емкости и, следовательно, динамические диски большего размера. Поскольку управление динамическими разделами осуществляется с помощью скрытой базы данных, то на них совершенно не влияет ограничение в 2,2 Тбайт, которое налагается таблицами разделов главной загрузочной записи. В сущности, динамические диски дают возможность операционным системам Windows 2000/XP Pro создавать разделы NTFS с емкостью до 16 ТиБ. Ограничения по емкости разделов NTFS приведены в табл. 25.27.

Таблица 25.27. Ограничения емкости разделов для NTFS

Ограничение емкости	Количество кластеров	Количество секторов в кластере	Общее количество секторов	Емкость раздела (в десятичной системе)	Емкость раздела (в двоичной системе)
Минимальный размер	32 698	1	32 730	16,76 Мбайт	15,98 МиБ
Максимальный размер основного диска	536 870 908	8	4 294 967 296	2,2 Тбайт	2 ТиБ
Максимальный размер динамического диска	4 294 967 295	8	34 359 738 392	17,59 Тбайт	16 ТиБ

Ниже перечислены характеристики NTFS.

- *Размер файлов не превышает 16 ТиБ (без 64 КиБ) или ограничен размером раздела меньшей емкости.* NTFS поддерживает до 4 294 967 295 ($2^{32} - 1$) файлов в разделе.
- *Обычно NTFS не используется на сменных носителях, так как не позволяет сразу же передать данные на жесткий диск.* Кроме того, отключение носителей, отформатированных в NTFS, без использования приложения Safe Removal может привести к потере данных. Для сменных носителей, которые могут быть неожиданно отключены, лучше использовать файловые системы FAT 12, FAT 16 или FAT 32.
- *NTFS поддерживает регистрацию транзакций и функции восстановления данных.* В случае отказа в работе NTFS восстанавливает после перезагрузки целостность файловой системы, используя для этого системный журнал и данные контрольных точек.

- NTFS динамически перераспределяет кластеры, содержащиеся в свободных секторах, и отмечает дефектные кластеры как поврежденные, что препятствует их дальнейшему использованию.
- NTFS имеет встроенные средства защиты, которые дают возможность устанавливать разрешения для каждого файла или каталога.
- NTFS имеет встроенную файловую систему кодирования (EFS). Эта система проводит динамическое кодирование и декодирование в процессе работы с зашифрованными файлами или папками, не позволяя другим пользователям обращаться к этим файлам.
- NTFS предоставляет возможность указывать дисковые квоты. Существует возможность отслеживать и контролировать использование различными пользователями дискового пространства в разделах NTFS.
- NTFS имеет функцию встроенного динамического сжатия, которая позволяет сжимать и разархивировать файлы по мере их использования.

Архитектура NTFS

Несмотря на существующие различия в структуре раздела файловых систем FAT и NTFS, они имеют подобные элементы, например загрузочную область. Раздел NTFS состоит из *главной таблицы файлов* (*master file table — MFT*). Однако MFT — это не то же самое, что FAT. Вместо использования таблицы со ссылками на кластеры, MFT содержит больше информации о файлах и каталогах в разделе. В некоторых случаях MFT может даже содержать файлы и каталоги.

При организации раздела NTFS система создает 10 системных файлов NTFS (табл. 25.28).

Таблица 25.28. Системные файлы NTFS

Имя файла	Назначение	Описание
\$mft	Master File Table (MFT)	Содержит запись для каждого файла в разделе NTFS в его атрибуте Data
\$mftmirr	Master File Table2 (MFT2)	Зеркальная копия MFT, используемая для восстановления
\$badclust	Файл поврежденных секторов	Содержит все поврежденные секторы раздела
\$bitmap	Карта распределения кластеров	Содержит карту всего раздела, указывающую на занятые кластеры
\$boot	Загрузочный файл	Содержит загрузочную информацию (если раздел загрузочный)
\$attrdef	Таблица определения атрибутов	Содержит определение всех системных и пользовательских атрибутов раздела
\$logfile	Файл журнала	Представляет собой файл журнала транзакций, используемый для восстановления
\$quota	Таблица квот	Представляет собой таблицу квот пользователей на данном разделе (используется только в NTFS 5)
\$upcase	Таблица символов	Используется для преобразования символов верхнего и нижнего регистров в символы верхнего регистра Unicode
\$volume	Раздел	Содержит информацию о разделе, например имя раздела и версию
\$extend	Файл расширения NTFS	Используется для хранения дополнительных расширений, таких, как квоты, идентификаторы объектов и параметры точек монтирования
Без имени	Индекс корневого файла	Представляет собой корневой каталог

Первая запись в MFT называется *дескриптором* (*descriptor*) и содержит информацию о расположении самой MFT. Загрузочный сектор в разделе NTFS содержит ссылку на расположение записи дескриптора. Вторая запись в MFT — это зеркальная копия дескриптора. Такое избыточное хранение данных обеспечивает большую устойчивость к ошибкам.

Третья запись — это запись файла журнала. Все операции (транзакции) в NTFS записываются в специальный файл журнала, что позволяет восстановить данные после сбоя. Остальная часть MFT состоит из записей для файлов и каталогов, которые хранятся в разделе. В файле NTFS хранятся атрибуты, определенные пользователем и системой. Атрибуты в разделе NTFS — это не простые флаги из раздела FAT. Вся информация о файле, т.е. атрибуты, в NTFS сохраняется вместе с файлом и является частью самого файла. Каталоги в NTFS со-

стоят в основном из индексов файлов в этом каталоге и не содержат такой информации о файле, как размер, дата, время и др.

Таким образом, MFT — это не просто список кластеров, это основная структура хранения данных в разделе. Если файл или каталог относительно небольшой (около 1 500 байт), его запись может храниться в MFT. Для больших массивов данных в MFT помещается указатель на файл или каталог, а сами данные располагаются в других кластерах в разделе. Эти кластеры называются *экстендами* (*extents*). Все записи в MFT, включая дескрипторы и файл журнала, могут использовать экстенды для хранения дополнительных атрибутов. Атрибуты файла, которые являются частью записи MFT, называются *резидентными* (*resident*), а атрибуты, расположенные в экстендах, — *нерезидентными* (*nonresident*).

NTFS 5.0 (NTFS 2000)

В Windows 2000 используется новая версия NTFS — файловая система NTFS 5. При установке Windows 2000 все существующие разделы NTFS автоматически обновляются до NTFS 5. Если на компьютере также используется Windows NT (мультизагрузка), то необходимо установить пакет обновления Service Pack 4 (SP4) или последующий, чтобы эта система могла работать с разделами NTFS 5. В процессе обновления изменяется версия драйвера NTFS.SYS.

Файловая система NTFS 5 обладает несколькими новыми свойствами.

- *Квотирование диска.* Администраторы системы могут устанавливать для пользователей ограничения на использование диска. Эти квоты могут быть нескольких уровней: Off, Tracking и Enforced.
- *Шифрование.* NTFS 5 поддерживает автоматическое шифрование и дешифрование файлов при их записи и считывании с диска.
- *Особые объекты файловой системы.* Позволяют использовать точки монтирования, т.е. перенаправление записи и считывания данных из папки на другой раздел или физический диск.
- *Поддержка больших файлов.* Позволяет более экономно расходовать дисковое пространство.
- *Журнал номеров последовательных обновлений.* Обеспечивает ведение журнала всех изменений файлов раздела.
- *Монтирование накопителей.* Позволяет “привязывать” дисководы и каталоги к пустому каталогу, расположенному на диске NTFS. Например, можно создать каталог на жестком диске, который позволит обращаться к данным, хранящимся на другом диске или накопителе.

Обратите внимание, что большинство этих свойств поддерживаются только операционной системой Windows 2000.

Изменения, внесенные в Windows XP

По сравнению с Windows 2000/NT в Windows XP изменилось местоположение главной таблицы файлов (MFT). В Windows 2000/NT таблица MFT обычно располагается в начале дискового пространства, занимаемого файловой системой NTFS. В Windows XP файлы метаданных \$logfile и \$bitmap располагаются в трех гигабайтах от начала дискового пространства, которое используется NTFS. Благодаря этому быстродействие системы в Windows XP по сравнению с Windows 2000/NT увеличилось на 5–8%.

Другая особенность реализации NTFS в Windows XP — уменьшение объема данных MFT, считываемых из памяти. Если все диски отформатированы в NTFS, то во время загрузки Windows считывает из таблицы MFT всего несколько сотен килобайт данных. При загрузке системы, жесткие диски которой (все или несколько) отформатированы в FAT 32, объем счи-

тываемых данных достигает нескольких мегабайт (количество данных зависит от числа накопителей и размера дисков). Таким образом, NTFS повышает эффективность использования системной памяти.

Совместимость NTFS

Получить доступ к разделу NTFS из DOS и других операционных систем нельзя. Windows NT предназначена для использования в качестве сетевой операционной системы, поэтому доступ к файлам в разделе NTFS можно получить с помощью сети. Для этого в NTFS поддерживаются имена файлов, удовлетворяющие стандарту “восемь-точка-три”.

Основное преимущество NTFS — обеспечение безопасности файлов и каталогов. Атрибуты безопасности в NTFS называются *разрешениями (permissions)* и устанавливаются системным администратором посредством предоставления доступа к данным на уровне прав пользователей и групп пользователей.

Алгоритм создания коротких имен файлов в Windows NT практически такой же, как и в файловой системе VFAT Windows 9x. Процесс создания имени файла, удовлетворяющего стандарту 8.3 для Windows 9x, а также особенности этого процесса в Windows NT/2000 описаны выше в главе.

Создание раздела NTFS

Создать раздел NTFS можно только на жестком диске. Его нельзя создать на дискете, а на сменном устройстве, например Iomega Zip или Jaz, можно. Существует три способа создания раздела NTFS:

- при установке Windows NT/2000 или после установки с помощью программ работы с диском;
- путем форматирования существующего раздела в NTFS (с удалением всех данных) с помощью команды Format системы Windows NT/2000 (параметр /fs:ntfs);
- посредством преобразования существующего раздела FAT в NTFS (с сохранением всех данных) при установке или после установки Windows NT/2000 с помощью программы Convert.

Инструменты для NTFS

Программы и методы работы с разделами FAT не имеют никакого отношения к NTFS, поскольку эти две файловые системы кардинально отличаются друг от друга. В Windows NT проверить диск на наличие возможных ошибок файловой системы и порченных секторов можно с помощью собственной версии программы CHKDISK. В Windows 2000/XP есть две версии CHKDISK — для запуска из командной строки и с графической оболочкой, а также специальная утилита дефрагментации DSKPROBE, включенная в Windows XP Professional, и пакет Resource Kit для Windows 2000.

Программы CHKDISK (Windows 2000/XP) и SCANDISK (Windows 9x/Me) отличаются тем, что CHKDISK, запущенная с помощью графического интерфейса Windows, не позволяет исправлять ошибки файловой системы. Если при запуске программы CHKDISK установлен флажок Автоматически исправлять ошибки файловой системы (Automatically Fix File System Errors), необходимо настроить автоматический запуск CHKDISK при следующей загрузке операционной системы. Программа CHKDISK, запущенная без использования этого параметра, позволяет выявить ошибки файловой системы. Для того чтобы найти и попытаться исправить сбойные секторы, необходимо запустить CHKDISK с помощью графического интерфейса Windows.

Совет

Для того чтобы получить дополнительную информацию о файловой системе того или иного жесткого диска, в Windows 2000/XP можно использовать программу FSUTIL, управление которой осуществляется с помощью командной строки. В частности, можно узнать, нужно ли запускать CHKDSK или какую-нибудь программу восстановления диска, созданную сторонними разработчиками. Например, для того чтобы определить, является ли диск D: неисправным (имеются ли ошибки файловой системы, которые необходимо исправить), достаточно запустить сеанс командной строки и ввести следующую команду:

```
FSUTIL DIRTY QUERY D:
```

Чтобы познакомиться с синтаксисом и примерами использования программы FSUTIL, введите в командной строке FSUTIL без каких-либо параметров.

Файловая система NTFS оснащена собственной автоматической системой восстановления диска. Кроме таких отказоустойчивых функций Windows NT/2000/XP, как зеркальное отображение диска (одновременное хранение данных сразу на двух жестких дисках) и разделение диска (данные распределяются по нескольким жестким дискам с добавлением данных четности для восстановления информации), эти операционные системы обладают двумя специальными технологиями, а именно:

- управление транзакциями;
- перераспределение кластеров.

Файловая система NTFS позволяет вернуться к любой точке *транзакции* (этим термином определяют изменения файла, хранящегося в разделе NTFS), если транзакция не была правильно завершена из-за ошибок диска, переполнения памяти или таких ошибок, как извлечение носителя или отключение устройства до завершения процесса транзакции. Каждая транзакция включает пять этапов.

1. NTFS создает системный журнал, содержащий записи операций с метаданными (обновления файлов, удаление и т.д.), после чего размещает файл в системной памяти.
2. NTFS сохраняет в памяти операции с фактическими метаданными.
3. NTFS отмечает запись транзакции в системном журнале как зафиксированную.
4. После завершения транзакции NTFS сохраняет системный журнал на жестком диске.
5. После выполнения транзакции NTFS сохраняет операции с фактическими метаданными на диске.

Этот процесс позволяет избежать появления случайных данных (потерянных кластеров) на жестких дисках NTFS.

Когда Windows NT/2000/XP обнаруживает испорченный сектор в разделе NTFS, данные испорченного кластера воссоздаются в другом кластере. Если жесткий диск входит в отказоустойчивый дисковый массив, то любые утраченные данные восстанавливаются посредством дублирования данных с других жестких дисков.

Несмотря на все эти возможности, Windows NT/2000/XP не хватает универсальной программы дефрагментации и восстановления данных NTFS-разделов. Рекомендуется использовать утилиты сторонних производителей, например Norton Utilities 2002 компании Symantec, совместимую с операционными системами Windows Me/NT 4.0/2000/XP.

Файловая система HPFS

Файловая система HPFS (High Performance File System) впервые была представлена в OS/2 версии 1.2 и содержала целый ряд улучшений по сравнению с существующей в то время файловой системой FAT 16. Система HPFS также поддерживалась Windows NT версий 3.x, однако в последующих версиях Windows ее поддержка была исключена. Файловую систему

HPFS со многих точек зрения можно рассматривать как предшественницу NTFS, впоследствии представленной в Windows NT.

Подробные сведения об HPFS приведены в разделе *Technical Reference* на прилагаемом компакт-диске.

Восстановление диска и данных

Команды Chkdsk, Recover и Scandisk — это “реанимационная бригада” DOS, занимающаяся восстановлением поврежденных данных на диске. Эти команды имеют очень простой и не слишком дружелюбный интерфейс, их применение зачастую оказывает значительное воздействие на систему, но иногда только они и могут помочь. Из перечисленных утилит наиболее известны, пожалуй, Recover, которая восстанавливает программы, и Chkdsk, используемая для проверки файловой структуры диска. Многие пользователи даже не подозревают, что Chkdsk может не только проверять, но и восстанавливать поврежденную файловую структуру диска. Еще одна программа — простая утилита Debug — может помочь вам в беде, но только в том случае, если вы точно знаете, что и как делать.

Утилита Scandisk является более мощной, чем Chkdsk и Recover, и заменяет эти две утилиты в DOS 6 и более поздних версиях, а также в Windows 9x.

Версия MS-DOS 5.0 и более ранние поддерживают только две утилиты, используемые для проверки диска, — CHKDSK и RECOVER. Использование и принципы работы команд CHKDSK и Recover описаны в 11-м издании книги, представленном на прилагаемом компакт-диске.

Программа RECOVER

Улучшенная версия программы RECOVER, которая позволяет восстанавливать данные из определенного файла, является единственной программой с интерфейсом командной строки, входящей в операционные системы Windows NT, 2000 и XP. Для использования этой версии программы RECOVER, которая поддерживает файловые системы FAT и NTFS, запустите сеанс командной строки и введите следующую команду:

```
RECOVER (диск\каталог\имя_файла)
```

Например, для того чтобы восстановить все читабельные секторы в файле Mynovel.txt, который находится в каталоге C:\My Documents\Writings, необходимо ввести следующую команду:

```
RECOVER C:\My Documents\Writings\Mynovel.txt
```

Поскольку для версий NT/2000/XP программы RECOVER требуется указать путь и имя файла, ее использование, в отличие от программы RECOVER для DOS, не может привести к повреждению файловой системы. К сожалению, эта версия не позволяет использовать групповые символы. Вместо них придется вводить в строку ввода имена файлов по одному или проверить жесткий диск, после чего попытаться восстановить поврежденные файлы с помощью утилиты сторонних разработчиков, например Norton Disk Doctor.

Программа Scandisk

Эта программа входит в поставку DOS версии 6 и более поздних, а также в Windows 9x. Она значительно мощнее утилит Chkdsk и Recover и выполняет функции их обеих. Программа Scandisk из Windows 95 OSR2 и Windows 98 может работать с FAT 32.

Замечание

В Windows NT 4.0 и Windows 2000/XP используется программа CHKDSK, которая является более мощным аналогом утилиты SCANDISK. Для получения дополнительной информации о программе SCANDISK, используемой в MS-DOS, обратитесь к 12-му изданию книги, представленному на прилагаемом компакт-диске.

Программа Scandisk больше похожа на упрощенную версию Norton Disk Doctor и позволяет проверять как целостность файловой структуры, так и работу секторов на физическом уровне. Обнаружив ошибки в каталогах или в FAT, Scandisk может их исправить. После определения дефектного сектора в FAT помечается дефектный кластер, содержащий этот сектор. При этом программа пытается восстановить поврежденный файл, причем сохраняются данные как до дефектного участка, так и после него.

В Windows 9x есть программа Scandisk для DOS и Windows. Файлы этих программ называются Scandisk.exe и Scandiskw.exe соответственно. Windows 9x проверяет диск в процессе установки операционной системы, а также после неверного завершения работы с системой. Вы можете также запустить программу Scandisk и ее “оконную” версию из командной строки.

Особенности работы программы Scandisk вы можете найти в книгах по операционным системам или в справочной системе Windows 9x.

Дефрагментация диска

Структура файловой системы FAT основана на хранении данных в виде кластеров, которые могут быть размещены в любой области диска, что позволяет компьютеру сохранять файлы практически любых размеров. Следование по цепочке FAT для обнаружения всех кластеров, содержащих данные конкретного файла, может привести к тому, что жесткому диску придется вести поиск во многих областях. Поскольку при этом перемещаются головки дисков, считывание файла, кластеры которого “разбросаны” по всему диску, существенно замедляет скорость работы жесткого диска, в отличие от файла, состоящего из последовательно расположенных кластеров.

Регулярное добавление, перемещение и удаление файлов приводит к фрагментации файлов, что замедляет скорость их обработки. Для решения этой проблемы необходимо регулярно проводить дефрагментацию жесткого диска. Программа дефрагментации считывает каждый файл на диске, используя таблицу FAT для доступа к кластерам, где бы они не находились.

Как уже упоминалось выше, в файловой системе FAT данные в кластерах могут располагаться в любом месте диска. И при поиске файла последний считывается из нескольких мест, что, естественно, снижает производительность системы. Для перемещения файла в одно место служат программы дефрагментации диска.

В Windows есть программа дефрагментации диска, которая работает с FAT 16 и FAT 32. В Windows 98/Me в программу дефрагментации была добавлена функция ускорения запуска приложений — перемещение часто запускаемых программ к началу диска.

Рассмотрим работу программы дефрагментации диска на примере. В табл. 25.29 приведены данные о расположении файлов в FAT.

Таблица 25.29. Фрагментированные файлы

Каталог		
Имя	Начальный кластер	Размер
Pledge.txt	1002	2
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значение	Назначение
00002	0	Первый доступный кластер
...
00999	0	Кластер доступен
01000	1001	Используется; ссылка на следующий кластер
01001	1004	Используется; ссылка на следующий кластер
01002	1003	Используется; ссылка на следующий кластер
01003	FFFFh	Конец файла
01004	1005	Используется; ссылка на следующий кластер
01005	FFFFh	Конец файла
...
65526	0	Последний доступный кластер

В рассматриваемом примере файл `Usconst.txt` фрагментирован на две части. После запуска программы дефрагментации этот файл может быть расположен на диске так, как продемонстрировано в табл. 25.30.

Таблица 25.30. Дефрагментированный файл

Каталог		
Имя	Начальный кластер	Размер
Pledge.txt	1004	2
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значение	Назначение
00002	0	Первый доступный кластер
...
00999	0	Кластер доступен
01000	1001	Используется; ссылка на следующий кластер
01001	1002	Используется; ссылка на следующий кластер
01002	1003	Используется; ссылка на следующий кластер
01003	FFFFh	Конец файла
01004	1005	Используется; ссылка на следующий кластер
01005	FFFFh	Конец файла
...
65526	0	Последний доступный кластер

В процессе дефрагментации части файла, разбросанные по всему диску, были соединены. Дефрагментация диска — довольно продолжительный процесс, поскольку необходимо выполнить большое количество операций чтения и записи.

Кроме программы дефрагментации диска, поставляемой с операционной системой Windows, существуют программы независимых разработчиков. Примером может служить программа Speed Disk из комплекта Norton Utilities.

Внимание!

Помните, что процесс дефрагментации — довольно опасная процедура. При дефрагментации происходит считывание, удаление и перезапись данных. Сбой в системе питания при выполнении этой операции может привести к плачевным последствиям. Поэтому перед дефрагментацией критически важных данных следует выполнять резервное копирование.

Операционная система Windows NT 4.0 не содержит программу дефрагментации, но подобная утилита имеется в Windows 2000/XP. Существуют также утилиты сторонних разработчиков, в частности программа VoptXP от компании Golden Bow System (www.vopt.com), которые могут использоваться с текущими и современными версиями Windows, но имеют более высокую скорость и больше возможностей, чем базовая программа дефрагментации Windows.

Программы независимых разработчиков

Если при чтении диска C: появляется сообщение `Sector not found` (сектор не найден), самое время воспользоваться программой, отличной от традиционных SCANDISK или RECOVER (в DOS). Кроме стандартных программ для работы с диском, поставляемых с операционной системой, существует огромное количество дисковых программ независимых разработчиков. Самый известный пакет таких программ — Norton Utilities, разработанный компанией Symantec. Большинство подобных программ созданы для DOS и Windows и могут работать с FAT 32. Все эти программы имеют существенный недостаток: их необходимо приобретать дополнительно; однако, как правило, игра стоит свеч.

Norton Utilities и Norton System Works

Утилита Norton Diagnostics, входящая в состав Norton Utilities, позволяет идентифицировать и проверить существующее аппаратное обеспечение, создать загрузочную дискету, используемую для перезапуска системы и тестирования жесткого диска при его выходе из строя, а также для восстановления ошибочно удаленных файлов и отмены случайного форматирования жесткого диска.

В настоящее время пакет Norton Utilities версии 2002 поставляется в качестве самостоятельного программного продукта или же в виде одного из компонентов программы Norton System Works 2003 (для операционных систем Windows 95 OSR 2.x/98/Me/NT/2000/XP). Программа Norton System Works 2003 также включает в себя Norton Anti-Virus и деинсталлятор Norton CleanSweep; в версию Professional вошли программа архивирования системной информации Norton Ghost, приложения WinFax Basic и Web Services. Некоторые программы, входящие в состав Norton Utilities, предназначены для запуска из командной строки или работы в режиме MS-DOS:

- Norton Disk Doctor (NDD . EXE);
- Disk Editor (DISKEDIT . EXE);
- UnErase (UNERASE . EXE);
- UnFormat (UNFORMAT . EXE);
- Rescue Restore (RESCUE . EXE).

Тем не менее большинство программ, вошедших в Norton Utilities версии 2002, предназначены непосредственно для работы в графической среде Windows. Не следует использовать более ранние версии Norton Utilities, например 8.0 (для Windows 3.1 и MS-DOS), с 32-разрядными версиями Windows, так как это может привести к потере данных из-за отсутствия поддержки длинных имен файлов и дисков большой емкости.

Файловые системы и программы независимых разработчиков

Приобретая программы для работы с жестким диском, следует обращать внимание в первую очередь на поддержку используемой файловой системы. Например, программа Norton Utilities 2001 поддерживает файловые системы FAT (включая FAT 32) и NTFS, в то время как ее ранние версии поддерживают только FAT 16. Программа SpinRite 5 поддерживает все разновидности FAT, однако не работает с NTFS. Обратите особое внимание на то, что никогда не следует применять программу для работы с диском, предназначенную для операционной системы более ранней версии. Это может привести к непоправимым последствиям.

Восстановление данных

Для восстановления потерянных данных иногда достаточно открыть окно приложения Корзина (Recycle Bin) и извлечь оттуда удаленный файл, а иногда приходится тратить сотни долларов на приобретение специализированного программного обеспечения или сервисных услуг. В наихудшем случае придется даже отправить дисковод в центр восстановления данных. Существует несколько факторов, которые определяют степень сложности процесса восстановления данных, в том числе:

- способ удаления данных;
- файловая система накопителя, на котором хранились удаленные данные;
- форма хранения данных: магнитные ленты, оптические и магнитооптические носители, флэш-память;
- используемая версия Windows или другой операционной системы;
- наличие программного обеспечения по защите данных, установленного в системе;
- наличие физических повреждений и исправность головок, жестких дисков и монтажной платы диска.

Программа Корзина Windows и удаление файлов

Проще всего восстанавливать файлы, отправленные в “корзину” Windows (начиная с Windows 95, приложение Корзина (Recycle Bin) является стандартным компонентом Windows). Запустите программу Проводник (Windows Explorer) или откройте окно Мой компьютер (My Computer), выделите мышью файл или группу файлов и нажмите клавишу <Delete>. В результате этой операции удаленный файл окажется в окне приложения Корзина. Несмотря на то что файл, отправленный в корзину, исчезает из каталога, в котором он до этого находился, этот файл фактически защищен от перезаписи. По умолчанию Windows 95 и более новые версии резервируют 10% дискового пространства на каждом жестком диске для приложения Корзина (в накопителях на сменных носителях эта программа отсутствует). Таким образом, 10-гигабайтовый накопитель резервирует около 1 Гбайт дискового пространства для корзины. В данном случае, если в корзину отправить файлы общим объемом менее 1 Гбайт, то эти так называемые *удаленные файлы* будут защищены операционной системой Windows. Однако если в корзину отправить более 1 Гбайт данных, то ранее отправленные файлы будут перезаписаны. Чем быстрее выяснится, что в корзину отправлен нужный файл, тем больше шансов его спасти.

Чтобы восстановить “удаленный” файл, запустите программу Корзина, выделите нужный файл, щелкните на нем правой кнопкой мыши и выберите команду Восстановить (Restore). Windows возвратит файл в исходное положение и удалит из корзины.

Если в тот момент, когда вы нажимаете клавишу <Delete> или выбираете в меню команду Удалить (Delete), удерживать нажатой клавишу <Shift>, удаляемый файл не попадет в корзину. В последнем случае для восстановления утраченных данных придется воспользоваться программным обеспечением сторонних разработчиков.

Восстановление файлов, не попавших в корзину

Приложение Корзина (Recycle Bin) является первой “линией обороны” при спасении утраченных данных, но ее возможности ограничены. Как отмечалось в предыдущем разделе, при определенных условиях выделенные файлы либо не попадают в корзину, либо перезаписываются последними удаленными файлами. Кроме того, Корзина не используется для хранения файлов, удаленных с помощью командной строки, или при замене файла его более новой версией.

Для восстановления файлов, не попавших в корзину, обратите внимание на программу Norton UnErase (которая входит в пакеты Norton Utilities и Norton SystemWorks). Следует заметить, что эффективность программы Norton UnErase и способы ее применения зависят от используемой версии Windows и файловой системы жесткого диска.

Использование программ Norton UnErase и Norton Protected Recycle Bin в Windows 9x/Me

В Windows 9x/Me с файловой системой FAT восстановление данных на жестком диске, не имеющем установленной программы Norton Utilities, не представляет собой ничего сложного. Тем не менее установка этой программы до того, как будут удаляться файлы, позволяет еще больше упростить их возможное восстановление. Если программа Norton UnErase еще не установлена, но возникла необходимость в восстановлении удаленных файлов, запустите программу непосредственно с загрузочного компакт-диска Norton Utilities. В этом случае Norton UnErase будет работать как программа с интерфейсом командной строки, поэтому придется ввести первую букву имени каждого восстанавливаемого файла.

Если программа Norton Utilities уже установлена, на рабочем столе вместо обычной пиктограммы Корзина появится значок Norton Protected Recycle Bin. В отличие от стандартной программы Корзина, приложение Norton Protected Recycle Bin сохраняет файлы, которые были заменены более новыми версиями, и файлы, удаленные средствами командной строки. Чтобы восстановить файл, сохраненный приложением Norton Protected Recycle Bin, щелкните мышью на этом значке, выберите нужный файл, щелкните на нем правой кнопкой мыши и выберите команду Восстановить (Restore). Удаленный файл вернется на прежнее место.

Внимание!

Не следует устанавливать программы восстановления данных на жесткий диск, данные которого вы пытаетесь восстановить, так как инсталляция подобных программ может привести к перезаписи утраченных данных. Если необходимо восстановить данные, которые находятся на загрузочном диске Windows, следует подключить к компьютеру второй жесткий диск, сконфигурировать его в системной BIOS как загрузочный, установить рабочую копию Windows, а затем загрузиться с этого диска и установить на него программу восстановления данных. По возможности установите диск большой емкости (по крайней мере 10 Гбайт или больше), что позволит получить несколько гигабайт свободного пространства для хранения восстановленных данных.

В качестве альтернативы попробуйте также запустить утилиту Norton Unerase Wizard из меню программы Norton Utilities. Эта утилита позволяет находить недавно удаленные файлы (они находятся в корзине), все защищенные файлы на локальных дисках (они также хранятся в корзине) и все восстанавливаемые файлы, которые находятся на локальных дисках. При выборе последнего параметра можно ограничить круг поиска, используя групповые символы или указывая типы файлов, которые следует найти. Для поиска файлов, которые не хранятся в корзине, необходимо ввести первую букву имени файла; можно также узнать, какие файлы были удалены той или иной программой. Для восстановления файла с помощью мастера Unerase Wizard следует выделить файл, указав при необходимости первую букву его имени, затем просмотреть содержимое файла, щелкнув на кнопке Quick View (если программа просмотра файлов поддерживает этот формат), после чего щелкнуть на кнопке Восстановить. При этом восстановленный файл вернется на прежнее место.

Операционные системы Windows 9x/Me позволяют находить утраченные файлы как на жестких дисках, так и на сменных носителях (дискеты, флэш-память), хотя программа Корзина используется только для файлов жесткого диска.

Использование программ Norton UnErase и Norton Protected Recycle Bin в Windows 2000/XP

В Windows 2000/XP функции Norton UnErase и Norton Protected Recycle Bin работают примерно так же, как и в Windows 9x/Me, но имеют определенные ограничения: мастер Unerase Wizard выполняет поиск файлов только на жестких дисках. Накопители на сменных дисках не поддерживаются.

Совет

Если вы хотите с помощью мастера Norton Unerase Wizard восстановить данные, записанные на дискете, загрузите компьютер с загрузочного компакт-диска Norton Utilities, а затем запустите соответствующий мастер из меню запуска того же компакт-диска. Этот метод подходит как для Windows 9x/Me, так и для Windows 2000/XP.

Если на компьютере установлены две операционные системы — Windows 9x/Me и Windows 2000/XP — и мастер Norton Unerase Wizard будет использоваться для восстановления данных на Zip-дисководах, накопителях USB и устройствах флэш-памяти, запустите компьютер в Windows 9x/Me и установите Norton Utilities или SystemWorks. И в дальнейшем, если необходимо отыскать утраченные данные на сменных носителях, всегда загружайте Windows 9x/Me.

Альтернатива Norton UnErase

Программный пакет System Suite 4.0 от компании VCOM (ранее реализуемый компанией Ontrack) представляет собой интегрированный набор служебных программ, который обеспечивает примерно такие же возможности, как и Norton UnErase. Но, в отличие от Norton UnErase, функция FileUndelete, включенная в System Suite, позволяет работать с жесткими дисками и с накопителями на сменных носителях во всех поддерживаемых версиях Windows, включая Windows XP.

Несмотря на то что Norton Disk Editor (DISKEDIT.COM) не относится к числу автоматических средств, его можно использовать для восстановления утраченных данных на дискетах, жестких дисках и сменных носителях различных типов. Это приложение работает с любой файловой системой и практически во всех операционных системах, включая Linux.

Восстановление файлов в NTFS

Файловая структура NTFS значительно сложнее, чем любая файловая система той или иной версии FAT. Кроме того, некоторые файлы могут быть сжаты с помощью функции сжатия данных, встроенной в NTFS. Поэтому для восстановления файлов, удаленных с жестких дисков NTFS, необходимо использовать специализированную NTFS-совместимую программу. Для этого подходят, например, программы Norton Utilities и Norton SystemWorks 2002 и более поздних версий, совместимые с NTFS. Следует использовать также функцию Norton Protection, которая позволяет сохранять удаленные файлы в течение определенного времени. Использование Norton Protection значительно расширяет возможности Norton UnErase.

Для восстановления удаленных файлов можно использовать не только такие программы, как Norton Utilities или Norton UnErase, включенную в Norton SystemWorks, но и программы, перечисленные ниже.

- *Active Undelete*. Программные продукты этой серии работают также с картами флэш-памяти. Для получения подробной информации и бесплатной демоверсии этой программы посетите Web-узел <http://www.active-undelete.com>.
- *Restorer 2000*. Существуют следующие версии этой программы: FAT, NTFS и Professional. Для получения подробной информации и бесплатной демоверсии программы посетите Web-узел <http://www.bitinart.net/r2k.shtml>.
- *Ontrack EasyRecovery*. Для получения подробной информации и бесплатной демоверсии программы посетите Web-узел <http://www.ontrack.com>.

Совет

Некоторые программы, используемые в NTFS, позволяют восстанавливать только те файлы, которые были созданы пользователем, вошедшим в данную систему. Другие файлы могут быть восстановлены только администратором. Это особенно важно в том случае, когда вы пытаетесь восстановить удаленные файлы в системе, в которой работает несколько пользователей. Для получения более подробной информации обратитесь к соответствующей документации.

Восстановление данных с разбитых на разделы и отформатированных жестких дисков

Форматирование жестких дисков, дискет или сменных носителей приводит к удалению таблицы размещения файлов, которая используется такими программами, как Norton UnErase или File-Undelete, включенной в пакет VCOM System Suite, для определения местоположения файлов. При повторной разбивке жесткого диска на разделы с помощью программы FDISK или другого подобного приложения (например, Управление диском (Disk Management) в Windows 2000/XP) происходит удаление исходной файловой системы и информации о ранее созданных разделах.

В таких случаях для восстановления данных следует использовать более мощные программные средства. Существует два способа восстановления данных случайно отформатированного жесткого диска.

- Использование программы для восстановления прежнего формата жесткого диска.
- Использование программы, с помощью которой можно проигнорировать вновь созданную файловую систему FAT и обратиться непосредственно к секторам диска, что позволит отыскать и восстановить утраченные данные.

Для восстановления данных, удаленных с жесткого диска при создании разделов, следует использовать программу прямого считывания данных с секторов диска.

Программа Norton Unformat и имеющиеся ограничения

Пакеты Norton Utilities и Norton SystemWorks содержат программу Norton Unformat, которую можно использовать для восстановления прежнего формата случайно отформатированного жесткого диска FAT. К сожалению, эта программа имеет ряд ограничений, которые относятся к современным файловым системам и использованию накопителей определенных типов.

- *Norton Unformat не поддерживает накопители NTFS.* Это означает, что данная программа далеко не всегда может использоваться для восстановления данных в системах на базе Windows 2000/XP.
- *Norton Unformat не может использоваться с дисковыми, для работы которых требуются драйверы устройств, например с накопителями на сменных носителях.*
- *Norton Unformat лучше работает в том случае, если копия файловой системы FAT и корневого каталога создается с помощью Norton Image.* Если образ файла устарел, Norton Unformat зачастую не справляется со своей задачей; при отсутствии образа файла Norton Unformat не позволит восстановить корневой каталог и фактические имена папок в корневом каталоге будут заменены последовательно пронумерованными названиями папки.
- *Norton Unformat не может отправить копию восстановленного файла на другой диск-вод или каталог.* Восстановленные файлы возвращаются на свои прежние места, т.е. на тот же диск и в тот же раздел. В том случае, если для определения местоположения данных Norton Unformat использует устаревший файл, созданный программой Norton Image, восстановление прежнего формата накопителя часто приводит к перезаписи имеющихся достоверных данных.

По этим причинам Norton Unformat не может быть оптимальным приложением для восстановления данных на отформатированном диске. В качестве альтернативы можно использовать мощную, но, к сожалению, требующую ручного управления программу Norton Disk Editor (DISKEDIT), которая позволяет восстановить прежний формат накопителя и данные, хранившиеся на случайно отформатированном диске. Существуют и более простые варианты.

Восстановление утраченных данных на другом дисковом

Разработано множество программ, с помощью которых можно восстановить данные, утраченные из-за случайного форматирования диска или разбивки его на разделы, перенести восстановленную информацию на другой носитель. К числу наилучших и наиболее универсальных программ относится серия программных продуктов EasyRecovery от компании Ontrack DataRecovery Services, которая является подразделением корпорации Kroll Ontrack, Inc. Семейство EasyRecovery объединяет три программы.

- *EasyRecovery DataRecovery.* Позволяет восстановить удаленные данные и файлы, хранившиеся на случайно отформатированных жестких дисках, дискетах и сменных носителях, а также поврежденные или разрушенные файлы Zip и Microsoft Word. Для хранения восстановленных файлов могут использоваться как локальные, так и сетевые каталоги.
- *EasyRecovery FileRepair.* Позволяет исправить или восстановить данные поврежденных или разрушенных файлов Zip и Microsoft Office (Word, Excel, Access, PowerPoint и Outlook). Для хранения восстановленных файлов могут использоваться локальные и сетевые каталоги.
- *EasyRecovery Professional.* Объединяет функции DataRecovery и FileRecovery и включает некоторые дополнительные возможности, в частности поиск файлов, функцию RawRecovery и определяемые пользователем параметры раздела, что позволяет восстанавливать данные даже при тяжелом повреждении файловой системы и случайной разбивке диска на разделы. Бесплатная пробная версия отображает перечень файлов, которые могут быть восстановлены (а также бесплатно исправляет и восстанавливает

Zip-файлы); для получения пробной версии обратитесь на Web-узел компании Ontrack (<http://www.ontrack.com>).

Более ранняя версия EasyRecovery Data Recovery Lite, позволяющая восстанавливать до 50 файлов, является одним из компонентов программы System Suite от компании VCOM.

Запустив программу EasyRecovery Professional, можно выбрать один из перечисленных ниже методов восстановления данных.

- *DeletedRecovery*. Восстановление удаленных файлов.
- *FormatRecovery*. Восстановление файлов, утраченных при случайном форматировании накопителей.
- *RawRecovery*. Восстановление файлов путем непосредственного считывания данных из секторов посредством технологии определения соответствий сигнатур файлов.
- *AdvancedRecovery*. Восстановление данных с удаленных или поврежденных разделов.

В том или ином случае необходимо определить другой накопитель, который будет использоваться для хранения восстановленных данных. Метод “только для чтения” позволяет сохранить содержимое исходного накопителя и воспользоваться другим методом восстановления данных, если не удалось решить эту задачу с помощью первого метода.

Какие параметры наиболее оптимальны для восстановления данных? В табл. 25.31 приведены результаты, полученные при различных сценариях потери данных и использовании тех или иных параметров восстановления. Для восстановления данных с логического 19-гигабайтового диска, отформатированного в Windows XP с файловой системой NTFS, использовалась программа EasyRecovery Professional.

Таблица 25.31. Параметры восстановления данных и результаты, полученные при использовании EasyRecovery Professional

Характеристика утраченных данных	Метод восстановления данных	Данные восстановлены?	Подробности	Примечания
Удаленный каталог	DeletedRecovery	Да	Все файлы восстановлены	Сохранились все длинные имена файлов и каталогов
Отформатированный накопитель (полное форматирование)	FormatRecovery	Да	Все файлы восстановлены	Для хранения восстановленных файлов были созданы новые каталоги; сохранились длинные имена файлов и каталогов, расположенных уровнем ниже корневого каталога
Логический дисковод удален с помощью программы Управление диском (Disk Management)	AdvancedRecovery	Да	Все файлы восстановлены	Сохранились все длинные имена файлов и каталогов
Отформатированный накопитель, на котором записаны новые данные	FormatRecovery	Частично	Восстановлены все файлы и каталоги, которые не были перезаписаны	Сохранились все длинные имена файлов и каталогов
После форматирования накопитель был разбит на разделы и отформатирован в системе FAT (диск 1, 117 Мбайт)	AdvancedRecovery	Нет	Файлы, которые могли быть восстановлены, не обнаружены	
	RawRecovery	Частично	Нефрагментированные файлы восстановлены	Исходная структура каталогов и файловые имена не сохранились; файлы каждого типа хранятся в отдельной папке, и все файлы последовательно пронумерованы

Характеристика утраченных данных	Метод восстановления данных	Данные восстановлены?	Подробности	Примечания
После форматирования накопитель был разбит на разделы и отформатирован в системе NTFS (диск 2, 18,8 Гбайт)	AdvancedRecovery	Нет	Файлы, которые могли быть восстановлены, не обнаружены	
	RawRecovery	Частично	Нефрагментированные файлы восстановлены	Исходная структура каталогов и файловые имена не сохранились; файлы каждого типа хранятся в отдельной папке, и все файлы последовательно пронумерованы

Как следует из табл. 25.31, до тех пор пока области данных на жестком диске не будут перезаписаны, полное восстановление данных возможно, причем даже в том случае, если дисковод был отформатирован или заново разбит на разделы. Таким образом, чем быстрее вы поймете, что отформатированный и разбитый на разделы жесткий диск содержит ценные данные, тем больше шансов. И чем больше пройдет времени, тем меньше данных может быть восстановлено. Кроме того, следует заметить, что “посекторный” поиск данных (процесс, получивший в Ontrack название RawRecovery) не позволит сохранить исходную структуру каталогов и длинные имена файлов. В этом случае придется заново создать необходимую структуру каталогов, а затем переименовать все восстановленные файлы, на что уйдет немало времени и сил.

Совет

При использовании программы EasyRecovery Professional или EasyRecovery DataRecovery для восстановления поврежденных Zip-файлов или файлов Microsoft Office воспользуйтесь параметрами меню Properties (Свойства) и укажите каталог, в котором будут храниться восстановленные файлы (исходное расположение или другой диск или каталог). По умолчанию восстановленные файлы Outlook будут скопированы в другую папку, тогда как файлы других типов останутся на своих прежних местах (по крайней мере до тех пор, пока не будет указан другой каталог).

Как следует из этого примера, специализированные программы восстановления данных, к которым относится EasyRecovery Professional от компании Ontrack, весьма эффективны. Но при этом они имеют довольно высокую стоимость. Поэтому, если есть пакеты Norton Utilities или Norton SystemWorks, а также свободное время, которое можно потратить на знакомство со структурой жесткого диска, попробуйте восстановить утраченные данные с помощью Norton Disk Editor.

Использование Norton Disk Editor

На своих семинарах, посвященных аппаратным средствам ПК (модернизации и ремонту) и восстановлению данных, при исследовании накопителей я часто использую Norton Disk Editor — незаслуженно игнорируемую программу, которая является одним из компонентов пакетов Norton Utilities и Norton SystemWorks. Я также использую эту программу для восстановления утраченных данных. Поскольку Disk Editor требует ручного управления, то иногда она оказывается более полезной, чем автоматические программы, которые далеко не всегда работают корректно. Например, в режиме работы с физическими секторами Disk Editor может работать с любым дисководом, независимо от используемой файловой системы. Это связано с функционированием на более низком уровне, чем операционная система. Кроме того, Disk Editor отображает структуру накопителя так, как не может ни одна программа, а потому является совершенным инструментом как для получения дополнительной информации о структуре диска, так и для

восстановления утраченных данных. В этом разделе речь идет о двух процедурах, которые можно выполнить с помощью программы Disk Editor.

- Восстановление файлов на дискете.
- Копирование файла, удаленного с жесткого диска, на другой накопитель.

В пакеты Norton SystemWorks, SystemWorks Professional или Norton Utilities для Window обязательно входит и программа Norton Disk Editor. Откройте папку *Norton Utilities*, которая находится в каталоге *Program Files*, и посмотрите, есть ли там файлы DISKEDIT.EXE и DISKEDIT.HLP.

Если файлы не найдены на жестком диске, запустите их с установочного компакт-диска Norton Utilities. Для SystemWorks или SystemWorks Professional найдите на компакт-диске папку \NU, где расположены нужные файлы.

Программа Disk Editor, управление которой осуществляется с помощью командной строки, предназначена для обращения к файловым системам, созданным на основе таблицы размещения файлов (FAT), к которым относятся FAT 12 (дискеты), FAT 16 (жесткие диски MS-DOS и первых версий Windows 95) и FAT 32 (жесткие диски Windows 95/98/Me). Если жесткие диски отформатированы в файловой системе FAT 16 или FAT 32, то Disk Editor можно использовать также с Windows NT/2000/XP; программа Disk Editor будет также работать в разделах NTFS, но в этом случае ее можно будет использовать только в режиме работы с физическими секторами.

Прежде чем приступить к восстановлению важных файлов или данных, хранящихся на жестком диске, рекомендуется опробовать программу Disk Editor на гибких дисках. Disk Editor требует исключительно ручного управления, поэтому возможность ошибки довольно высока.

Файлы Disk Editor поместятся на обычной дискете, но, если вы плохо знакомы с этой программой, лучше разместить их на запасном диске. Никогда не копируйте файлы Disk Editor (или любой другой программы, используемой для восстановления данных) на диск, содержащий данные, которые будут восстанавливаться, что может привести к перезаписи области данных и уничтожению исходных файлов. Например, если необходимо проверить или восстановить данные на гибком диске, создайте на жестком диске папку с именем Disk Editor и скопируйте туда все файлы.

Обычно в программе Disk Editor используются команды, вводимые с клавиатуры, но можно также работать и с мышью. Для этого необходимо подключить мышшь к последовательному порту или порту PS/2 (мышшь USB, как правило, не работает в режиме командной строки, но если она имеет соответствующий адаптер, ее можно подключить к порту PS/2), а затем загрузить драйвер мыши MS-DOS (как правило, MOUSE.COM). Все это необходимо сделать до начала работы с Disk Editor. Для работы с мышью Logitech имеет смысл загрузить драйвер MS-DOS с Web-узла компании Logitech. В свою очередь, компания Microsoft не обеспечивает пользователей драйверами MS-DOS, поэтому, чтобы получить необходимый драйвер, нужно посетить Web-узел <http://www.bootdisk.com/readme.htm#mouse>.

При использовании других типов мыши попробуйте установить драйвер Microsoft/Logitech или обратитесь к разработчику. Имейте в виду, что колесо прокрутки и дополнительные кнопки мыши не будут работать с драйверами MS-DOS.

Рекомендуется скопировать драйвер мыши в ту же папку, в которой находится программа Disk Editor.

Использование программы Disk Editor для проверки накопителя

Чтобы запустить программу, выполните ряд действий.

1. Загрузите компьютер в режиме командной строки (не Windows); программе Disk Editor требуется монопольный доступ к анализируемому накопителю. В Windows 9x откройте меню загрузки, нажав во время загрузки системы клавиши <F8> или <Ctrl>.

и выберите опцию Режим командной строки (Safe Mode Command Prompt). Можно также воспользоваться загрузочным диском Windows 9x/Me, который создается посредством приложения Установка и удаление программ (Add/Remove Programs). В Windows 2000/XP, вставьте пустую дискету в дисковод A:, откройте окно Мой компьютер (My Computer), щелкните правой кнопкой мыши на имени дисковода и выберите команду Форматировать (Format). Создайте загрузочную дискету, установив флажок Создание загрузочного диска MS-DOS (Create an MS-DOS Startup Disk), и перезагрузите компьютер.

2. Перейдите в каталог, в котором находится драйвер мыши и программа Disk Editor.
3. Введите команду MOUSE (если драйвер мыши называется MOUSE.COM или MOUSE.EXE; если файл называется как-нибудь иначе, введите в строку ввода соответствующее имя). Загрузите драйвер мыши, нажав клавишу <Enter>.
4. Введите команду DISKEDIT и запустите программу, нажав клавишу <Enter>. Если дисковод не будет определен, программа Disk Editor выполнит сканирование того диска, на котором она установлена. Чтобы использовать Disk Editor для проверки дискет, следует ввести команду DISKEDIT A:, определив тем самым накопитель на гибких дисках. При выполнении этой команды Disk Editor просмотрит содержимое дискеты, определяя местоположение файлов и папок.
5. При первом запуске Disk Editor на экране появится сообщение, что эта программа выполняется в режиме “только для чтения” (по крайней мере до тех пор, пока ее конфигурация не будет изменена с помощью меню Tools). Для продолжения щелкните на кнопке ОК.

Запустив программу Disk Editor, можно перейти к анализируемому диску или диску, содержащему утраченные данные. Выполните перечисленные ниже действия.

1. Откройте меню Object, нажав комбинацию клавиш <Alt+O>.
2. Выберите опцию Drive.
3. В меню Logical Disks выберите проверяемый дисковод.
4. Структура диска будет отображена в основном окне Disk Editor.

Программа Disk Editor обычно начинает работу в режиме каталога (Directory), но можно выбрать и другие режимы, воспользовавшись меню View. При просмотре диска в режиме Directory на экране появится перечень файлов, который будет выглядеть примерно так, как на рис. 25.2.

В столбце Name перечислены названия записей каталога, а в столбце .EXT указаны расширения файла/папки (если таковые существуют). В столбце ID указывается тип записи, в том числе:

- *Dir* — каталог (папка);
- *File* — файл данных;
- *LFN* — фрагмент длинного файлового имени Windows. В операционной системе Windows перед фактическим именем файла сохраняется начало LFN. Если LFN содержит более 13 символов, то оставшаяся часть файлового имени сохраняется в дополнительных записях каталога. В трех следующих столбцах указываются размер файла, дата и время.

В столбце Cluster указывается номер кластера, в котором размещается первая часть файла. При форматировании накопителей диски разбиваются на кластеры или ячейки размещения, которые представляют собой наименьший блок выделяемой памяти, используемый для хранения файла. Размеры кластера зависят от емкости накопителя и файловой системы, используемой при форматировании диска.

Disk Editor														
Object	Edit	Link	View	Info	Tools	Help	More>							
Name	.Ext	ID	Size	Date	Time	Cluster	76	A	R	S	H	D	U	
..		Dir	0	9-19-02	4:02 pm	0	0	-	-	-	-	D	-	
.wpd		LFN				0	0	-	R	S	H	-	U	
SSL_outline01		LFN				0	0	-	R	S	H	-	U	
SSLOUT~1	WPD	File	5080	1-04-00	10:40 am	230	0	A	-	-	-	-	-	
SSL_01.wpd		LFN				0	0	-	R	S	H	-	U	
SSL_01~1	WPD	File	13081	1-15-00	2:47 pm	240	0	A	-	-	-	-	-	
SSL_02.wpd		LFN				0	0	-	R	S	H	-	U	
SSL_02~1	WPD	File	13234	1-15-00	4:12 pm	295	0	A	-	-	-	-	-	
te_guide.html		LFN				0	0	-	R	S	H	-	U	
secure_web_si		LFN				0	0	-	R	S	H	-	U	
SECURE~1	HTM	File	48294	1-15-00	4:03 pm	321	0	A	-	-	-	-	-	
il_secure.gif		LFN				0	0	-	R	S	H	-	U	
IL_SEC~1	GIF	File	22999	1-15-00	4:04 pm	462	0	A	-	-	-	-	-	
LOCK	GIF	File	8389	1-15-00	4:04 pm	527	0	A	-	-	-	-	-	
rans.gif		LFN				0	0	-	R	S	H	-	U	
Cluster 631, Sector 662														
verisignsealt		LFN				0	0	-	R	S	H	-	U	
VERISI~1	GIF	File	6006	1-15-00	4:04 pm	632	0	A	-	-	-	-	-	
SSL_03.wpd		LFN				0	0	-	R	S	H	-	U	
SSL_03~1	WPD	File	18378	1-15-00	5:24 pm	681	0	A	-	-	-	-	-	
Sub-Directory														
A:\2000SS~1														
Cluster 631														
Offset 544, hex 220														

Рис. 25.2. Содержимое каталога типичного гибкого диска, отображенное в окне Norton Disk Editor

Буквы A, R, S, H, D и V используются в записях каталога в качестве атрибутов. Буква A (Archive) указывает на то, что резервная копия файла не создавалась с момента его последнего изменения. Буква R означает, что запись каталога имеет атрибут “только для чтения”, а буква S — атрибут “системный”. Скрытые файлы имеют атрибут H (Hidden), а для обозначения каталогов используется атрибут D (Directory). Записи LFN имеют атрибут V.

Файл VERISI~1.GIF (он выделен черным цветом и расположен в нижней части рис. 25.2) имеет несколько интересных особенностей. Тильда (~) и цифра в конце имени файла указывают на то, что этот файл был создан в 32-разрядной версии Windows (Windows 9x/Me/2000/XP), которая дает пользователю возможность сохранить файл с длинным (более восьми символов) файловым именем (плюс трехсимвольное расширение файла, например .EXE, .BMP или .GIF). Кроме того, длинные файловые имена могут включать в себя пробелы и другие символы, которые не поддерживаются ранними версиями Windows и MS-DOS. Создание записей LFN в различных версиях Windows описано выше в главе.

Просмотр содержимого накопителей в программе Проводник (Windows Explorer) или в окне Мой компьютер (My Computer) позволяет увидеть длинные имена файлов. Для того чтобы увидеть имя DOS в графическом интерфейсе Windows, щелкните правой кнопкой мыши на имени файла и выберите в контекстном меню опцию Свойства (Properties). При работе в режиме командной строки для этого можно воспользоваться командой DIR. Имя LFN сохраняется в виде одной или нескольких отдельных записей каталога, которые располагаются прямо перед псевдонимом DOS. Настоящее имя файла VERISI~1.GIF (Verisignsealt-trans.gif) состоит из 21 символа, поэтому для хранения длинного файлового имени требуется две дополнительные записи (каждая запись каталога может содержать до 13 символов LFN), как показано на рис. 25.2.

Определение количества кластеров, используемых файлом

Как отмечалось в начале главы, в области диска, которая называется таблицей размещения файлов, хранятся начальная ячейка файла и все дополнительные кластеры, используемые для хранения файла. Например, файл VERISI~1.GIF начинается с кластера 632. Кластеры представляют собой наименьшие структуры диска, используемые для хранения файла, размер которых зависит от файловой системы диска, где размещаются файлы, и емкости носителя. В данном случае файл хранится на диске емкостью 1,44 Мбайт, кластеры кото-

рого имеют размер 512 байт (один сектор). Размер кластера диска необходимо знать в том случае, если вы собираетесь восстанавливать данные с помощью программы Disk Editor.

Чтобы определить размер кластера, введите в командную строку команду CHKDSK C:, при выполнении которой отобразится размер ячейки размещения (кластера) и другие статистические данные, относящиеся к указанному диску.

Для того чтобы определить количество кластеров, необходимых для хранения файла, узнайте размер данного файла и сопоставьте с размером кластера диска, на котором он находится. Файл VERISI~1.GIF содержит 6 006 байтов. Поскольку этот файл хранится на дискете, имеющей кластеры с размером 512 байт, следовательно, он занимает несколько кластеров; но сколько именно? Чтобы это узнать, разделите размер файла на размер кластера и округлите полученную величину до следующего целого числа. Результаты вычислений приведены в табл. 25.32.

Таблица 25.32. Определение числа кластеров, используемых файлом VERISI~1.GIF

Размер файла (FS) VERISI~1.GIF	Размер кластера (CS)	Величина (CR), полученная при делении FS на CS	Полученная величина (CR), округленная до следующего целого числа
6 006	512	11,730 468 75	12

Как следует из табл. 25.32, файл VERISI~1.GIF занимает на дискете 12 кластеров; на жестком диске FAT 16 или FAT 32 этот файл занял бы меньше кластеров (точное значение зависит от используемой файловой системы и размера жесткого диска). Чем больше кластеров занимает данный файл, тем выше вероятность того, что при удалении файла некоторый фрагмент его области данных будет вскоре перезаписан другими, более новыми данными. Чем быстрее вы возьметесь за восстановления файла, отправленного “мимо корзины” либо удаленного с дискеты или сменного носителя (накопители на гибких дисках и сменных носителях не поддерживают приложение Корзина), тем больше шансов на успех.

В окне программы Norton Disk Editor показан начальный кластер (632) файла VERISI~1.GIF. Если файл хранится на диске, где много свободного пространства, то его остальные кластеры будут, скорее всего, находиться непосредственно за начальным кластером. На фрагментированном диске для хранения остальных фрагментов файла могут использоваться кластеры, находящиеся в разных местах диска. Восстанавливать данные, которые занимают последовательно расположенные кластеры, значительно проще, поэтому имеет смысл чаще проводить фрагментацию жестких дисков.

Чтобы увидеть остальные кластеры, занятые этим файлом, наведите на файл курсор, нажмите комбинацию клавиш <Alt+L> или откройте меню Link, после чего выберите опцию Cluster Chain (FAT); для непосредственного перехода к нужным данным можно также нажать комбинацию клавиш <Ctrl+T>. В результате на экране появится список кластеров, показанных в порядке их представления в таблице размещения файлов (рис. 25.3). Кластеры, которые используются этим файлом, выделены красным цветом. В нижней части экрана показано имя файла. Последний кластер файла обозначается символом <EOF> (End Of File — EOF).

Как операционная система отмечает удаленный файл

При удалении файла (в данном случае VERISI~1.GIF) с диском, на котором находился файл, происходят определенные изменения (рис. 25.4).

- Первый символ имени файла (V) был заменен символом “σ” (строчная сигма).
- В столбце ID появились записи новых типов, относящиеся к файлу и соответствующему имени LFN:
 - *Erased* — удаленный файл;
 - *Del LFN* — LFN, принадлежащий удаленному файлу.

Disk Editor									
Object	Edit	Link	View	Info	Tools	Help			
486	487	488	489	490	491	492	493		
494	495	496	497	498	499	500	501		
502	503	504	505	506	<EOF>	508	509		
510	511	512	513	514	515	516	517		
518	519	520	521	522	523	524	525		
526	<EOF>	528	529	530	531	532	533		
534	535	536	537	538	539	540	541		
542	543	<EOF>	545	546	547	548	549		
550	551	552	553	554	555	556	557		
558	559	560	561	562	563	564	565		
566	567	568	569	570	571	572	573		
574	575	576	577	578	579	580	581		
582	583	584	585	586	587	588	589		
590	591	592	593	594	595	596	597		
598	599	600	601	602	603	604	605		
606	607	608	609	610	611	612	613		
614	615	616	617	618	619	620	621		
622	623	624	625	626	627	628	629		
630	<EOF>	1015	633	634	635	636	637		
638	639	640	641	642	643	<EOF>	645		
646	647	648	649	650	651	652	653		
FAT (1st Copy)							Sector 2		
A:\2000SS~1\VERISI~1.GIF							Cluster 632, hex 278		

Рис. 25.3. FAT-представление файла VERISI~1.GIF. Все кластеры файла располагаются друг за другом

Disk Editor											
Object	Edit	Link	View	Info	Tools	Help	More>				
Name	Ext	ID	Size	Date	Time	Cluster	76	A	R	S	H D U
Cluster 144, Sector 175											
.		Dir	0	9-19-02	4:02 pm	144	-	-	-	D	-
..		Dir	0	9-19-02	4:02 pm	0	-	-	-	D	-
.wpd		LFN				0	-	R	S	H	- U
SSL outline01		LFN				0	-	R	S	H	- U
SSLOUT~1	WPD	File	5080	1-04-00	10:40 am	230	A	-	-	-	-
SSL_01.wpd		LFN				0	-	R	S	H	- U
SSL_01~1	WPD	File	13081	1-15-00	2:47 pm	240	A	-	-	-	-
SSL_02.wpd		LFN				0	-	R	S	H	- U
SSL_02~1	WPD	File	13234	1-15-00	4:12 pm	295	A	-	-	-	-
te_guide.html		LFN				0	-	R	S	H	- U
secure_web_si		LFN				0	-	R	S	H	- U
SECURE~1	HTM	File	48294	1-15-00	4:03 pm	321	A	-	-	-	-
il_secure.gif		LFN				0	-	R	S	H	- U
IL_SEC~1	GIF	File	22999	1-15-00	4:04 pm	462	A	-	-	-	-
LOCK	GIF	File	8389	1-15-00	4:04 pm	527	A	-	-	-	-
rans.gif		Del LFN				0	-	R	S	H	- U
Cluster 631, Sector 662											
verisignsealt		Del LFN				0	-	R	S	H	- U
VERISI~1	GIF	Erased	6006	1-15-00	4:04 pm	632	A	-	-	-	-
Sub-Directory							Cluster 631				
A:\2000SS~1							Offset 544, hex 220				

Рис. 25.4. Структура каталога после удаления файла VERISI~1.GIF

Также обратите внимание, что начальный кластер (632) все еще отображен в столбце Cluster.

Содержимое всех ячеек памяти, расположенных в таблице размещения файлов сразу после начального кластера, заменено нулями (обнулено), что указывает операционной системе на доступность этих кластеров для повторного использования. Таким образом, если сразу же не заняться восстановлением данных, некоторые из этих кластеров (или даже все) могут быть перезаписаны новыми данными. Файл, о котором идет речь, является графическим файлом формата GIF, поэтому потеря одного кластера может привести к уничтожению всего файла.

Проанализировав процесс удаления файла, можно предположить, что процесс восстановления должен включать в себя следующие этапы:

- восстановление исходного имени файла;
- локализация кластеров, используемых файлом;
- восстановление записей FAT, относящихся к этому файлу;
- повторное связывание записей LFN для каждого файла.

Наиболее важными операциями будет локализация используемых кластеров и восстановление записей FAT, относящихся к данному файлу. Тем не менее, если этот файл относится к той или иной программе, восстановление его первоначального имени является необходимым условием для работы последней (с учетом того, что программа не может быть перезагружена). При этом восстановленные записи LFN дают возможность пользователю Windows, привыкшему к длинным файловым именам, упростить работу с файлом.

Чтобы внести в исходный диск какие-либо изменения, переконфигурируйте программу Disk Editor для работы в режиме “Read-Write” (чтение/запись).

Чтобы перейти в этот режим, выполните ряд действий.

1. Откройте меню Tools, нажав клавиши <Alt+T>.
2. Откройте диалоговое окно Configuration, нажав клавишу <N>.
3. Сбросьте флажок Read Only, нажав клавишу <Пробел>.
4. Несколько раз нажмите клавишу <Tab>, пока не выделите поле Save.
5. Нажмите клавишу <Enter> для того, чтобы сохранить внесенные изменения и вернуться в исходное окно.

Внимание!

В качестве меры предосторожности рекомендуется, перед тем как начать восстановление данных, воспользоваться командой DISKCOPY и создать точную посекторную копию гибкого диска. При этом лучше работать не с оригиналом, а с созданной копией диска. Это позволит обезопасить оригинальную дискету от любых ошибок, которые могут быть допущены в процессе восстановления данных. Кроме того, при необходимости можно сделать другую копию.

Как только будут сохранены внесенные изменения, программа Disk Editor перейдет в режим “Read-Write” и будет использовать его до возвращения в предыдущий режим “Read-Only” (только для чтения). Чтобы это сделать, необходимо выполнить ранее описанные действия, но при этом установить флажок Read-Only. При работе с программой Disk Editor в режиме “Read-Write” во время сканирования накопителя на экране появится сообщение Drive x is Locked.

Восстановление удаленного файла

После того как программа Disk Editor была переведена в режим “Read-Write”, ее можно использовать для восстановления удаленного файла. Выполните перечисленные ниже действия.

1. Перейдите в каталог, в котором находится удаленный файл. Для этого выделите каталог, содержащий указанный файл, и нажмите клавишу <Enter>. В этом примере будет восстанавливаться удаленный файл VERISI~1.GIF.
2. Установите курсор под строчной буквой “сигма” (σ) и введите символ для переименования файла.
3. Если клавиатура находится в режиме *Insert*, то сигма переместится в правую сторону; чтобы ее удалить, нажмите клавишу <Delete>.

4. Эта операция позволит восстановить имя удаленного файла. Обратите внимание, что в столбце ID вместо записи Erased появится запись File. Однако это не означает, что процесс восстановления файла полностью завершен. Теперь необходимо найти остальные кластеры, которые используются данным файлом. Начальный кластер файла указан в правой части окна, рядом с именем файла.
5. Чтобы перейти к следующему кластеру, который занят этим файлом, нажмите комбинацию клавиш <Ctrl+T> для выбора команды Cluster Chain. Поскольку имя файла было изменено, для продолжения этой операции необходимо записать внесенные изменения. Достаточно нажать клавишу <W> или щелкнуть на кнопке Write.
6. Программа Disk Editor перейдет к первому кластеру, который используется удаленным файлом. Трехзначные номера кластеров (см. рис. 25.3) были заменены нулями (0). Поскольку этот файл занимает 12 кластеров, в окне программы должно быть 12 последовательно расположенных обнуленных кластеров (если, конечно, файл дефрагментирован).
7. Чтобы определить, соответствуют ли эти кластеры данному файлу, откройте меню Object, нажав комбинацию клавиш <Alt+O> или щелкнув мышью на опции Object. Затем откройте диалоговое окно Cluster (для этого введите в командной строке букву С или нажмите комбинацию клавиш <Alt+C>). Введите номер начального кластера (в данном случае — 632) и номер конечного кластера (644). Чтобы отобразить эти кластеры, щелкните на кнопке ОК.

Программа Disk Editor автоматически переключается на оптимальное представление тех или иных объектов. В данном случае используется представление данных в шестнадцатеричном формате (рис. 25.5). Обратите внимание, что первой записью в кластере 632 является выражение GIF89A (как показано в правом столбце). Поскольку удаленный файл был в формате GIF, это именно то, что нужно. Кроме того, файл в формате GIF является двоичным графическим файлом, поэтому остальная информация, хранящаяся в определенных секторах, будет представлена набором неудобочитаемых символов. Обратите внимание: конец файла обозначен рядом нулей (0), которые занимают несколько секторов диска, расположенных перед началом другого файла.

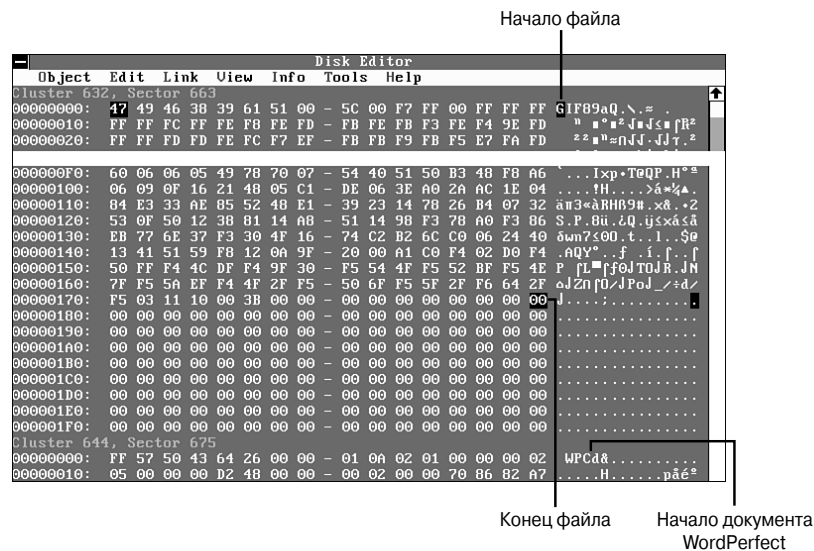


Рис. 25.5. Начало и конец файла VERISI~1.GIF

Поскольку область, занятая пустыми кластерами (632–644), содержит двоичные данные, начинающиеся выражением GIF89A, это гарантирует, что кластеры содержат необходимые данные.

8. Теперь необходимо вернуться к таблице размещения файлов (FAT) и ввести номера кластеров файла. Откройте меню Object и выберите опцию Directory. Затем укажите текущий каталог и щелкните на кнопке OK.
9. Укажите курсором на запись VERISI~1.GIF, откройте меню Link и выберите параметр Cluster Chain (FAT). При выполнении функции Cluster Chain создаются ссылки на кластеры, расположенные после начального кластера (632); введите в первое пустое поле число 633 и продолжайте так до тех пор, пока не введете число 643 и не разместите курсор на последнем пустом поле. Это поле должно содержать отметку <EOF>, которая обозначает конец файла. Откройте меню Edit, нажав комбинацию клавиш <Alt+E> и выберите опцию Mark (или нажмите комбинацию клавиш <Ctrl+B>). Снова откройте меню Edit и выберите команду Fill. Затем выберите в меню опцию End of File и щелкните на кнопке OK. Чтобы получить представление о том, как будет выглядеть таблица FAT после внесенных изменений, обратитесь к рис. 25.3.
10. Чтобы сохранить внесенные изменения, откройте меню Edit еще раз и выберите команду Write. При появлении сообщения о сохранении изменений щелкните на кнопке Write; затем щелкните на кнопке Rescan the Disk.
11. Теперь следует вернуться к представлению Directory; откройте меню Object и выберите опцию Directory. Затем щелкните на кнопке OK.
12. Записи LFN, расположенные непосредственно над файлом VERISI~1.GIF, все еще имеют отметку Del LFN. Чтобы восстановить их связь с файлом VERISI~1.GIF, выделите первую запись (verisignsealt), откройте меню Tools, нажав комбинацию клавиш <Alt+T>, и выберите опцию Attach LFN. При появлении запроса щелкните на кнопке Yes. Повторите эту операцию для записи rans.gif.
13. Чтобы убедиться, что файл успешно восстановлен, завершите работу программы Disk Editor и откройте этот файл посредством какого-либо приложения. Если кластеры были корректно размещены и соединены, файл будет открыт обычным образом.

Как видите, восстановление данных занимает довольно много времени, но, по существу, представляет собой точно такой же процесс, который Norton UnErase и подобные ей программы выполняют автоматически. Тем не менее Disk Editor позволяет выполнять такие операции на дисках различных типов, использующих файловые системы FAT. Эта программа является фаворитом среди “продвинутых” пользователей Linux.

Восстановление файлов на жестких дисках или картах флэш-памяти

Что нужно сделать для того, чтобы восстановить файл, удаленный с жесткого диска или карты флэш-памяти? Лучше всего переписать восстановленный файл на другой диск (желательно на дискету, если файл достаточно мал) или на другой раздел того же жесткого диска, имеющий другое буквенное обозначение. Эту задачу также можно выполнить с помощью программы Disk Editor.

Совет

Если необходимо восстановить данные, удаленные с жесткого диска, и скопировать их на другой носитель, переведите программу Disk Editor в режим “Read-Only” (только для чтения), чтобы предотвратить внесение случайных изменений в содержимое жесткого диска. При использовании Disk Editor в многозадачной среде, например в операционной системе Windows, программа по умолчанию работает в режиме “Read-Only”.

Процесс поиска файла ничем не отличается от ранее описанного.

1. Определите размер кластера (ячейки размещения) накопителя, на котором находится данный файл.
2. Запустите программу Disk Editor для того, чтобы узнать имя удаленного файла и определить соответствующие кластеры.

При этом нет необходимости восстанавливать имя файла, поскольку файл будет скопирован на другой диск (или дискету).

Кластеры копируются в другой файл, поэтому следует воспользоваться опциями меню Object для просмотра кластеров и убедиться, что они содержат необходимые данные. Откройте меню Object, выберите опцию Cluster и определите диапазон кластеров, которые, как указывает цепочка кластеров, должны содержать необходимые данные. В некоторых случаях в первом кластере указывается тип файла. Например, в начале графического файла GIF расположено выражение GIF89A, тогда как начальный кластер документа WordPerfect содержит выражение WPC.

Совет

Перед тем как начать восстановление файлов того или другого типа, постарайтесь просмотреть начальные и конечные кластеры этих файлов, используя программу Norton Disk Editor. Это особенно важно в тех случаях, когда будут восстанавливаться файлы с отформатированных носителей. Попробуйте также создать базу данных шестнадцатеричных символов, расположенных в начале и конце восстанавливаемых файлов.

При восстановлении файлов, содержащих текстовые фрагменты, например файлов Microsoft Word или WordPerfect, программу Disk Editor можно переключать в различные режимы просмотра. Для отображения текста нажмите клавишу <F3>, что позволит перейти в текстовый режим. Однако для определения начальных и конечных кластеров файла лучше воспользоваться режимом отображения в шестнадцатеричной форме (Hex). Чтобы перейти в этот режим, нажмите клавишу <F2>. На рис. 25.6 показаны начало файла Microsoft Word в текстовом формате (Text) и конечная область файла в шестнадцатеричном формате (Hex).

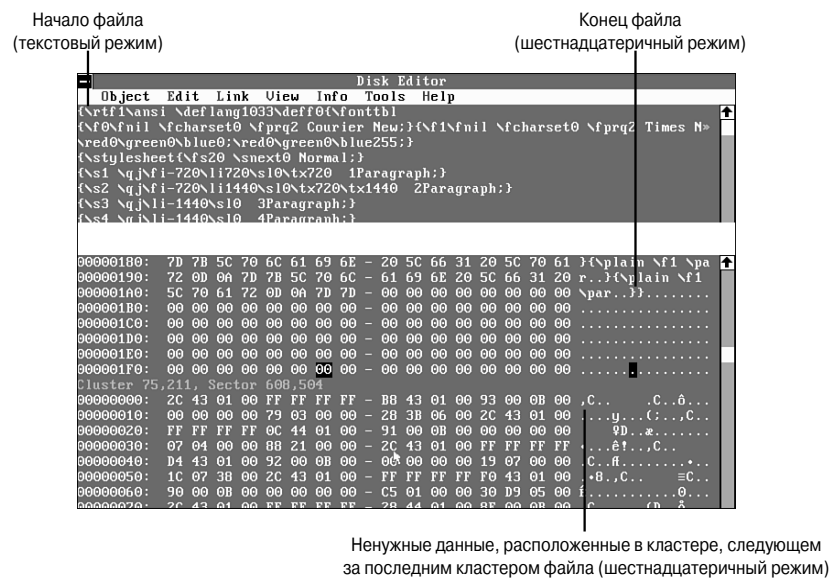


Рис. 25.6. Просмотр содержимого удаленного файла с помощью программы Disk Editor

Для того чтобы безошибочно скопировать содержимое кластеров в файл, необходимо определить секторы, непосредственно содержащие файл. В верхней области окна Disk Editor показаны номера секторов и кластеров. Например, файл, представленный на рис. 25.6, начинается с кластера 75 207, который также является сектором под номером 608 470. Конечная область файла размещена в секторе 608 503.

Для записи секторов в новый файл выполните ряд действий.

1. Откройте меню Object.
2. Выберите команду Sector.
3. Определите начальные и конечные секторы файла.
4. Щелкните на кнопке ОК.
5. Просмотрите секторы на предмет наличия необходимых данных.
6. Откройте меню Tools.
7. Выберите команду Write Object to.
8. Выберите параметр To a File.
9. Укажите накопитель, на который будут записаны данные.
10. Присвойте файлу имя DOS-типа (8 символов плюс 3-символьное расширение); этот файл можно будет переименовать по завершении работы с Disk Editor.
11. Для того чтобы записать файл, щелкните на кнопке ОК, а затем на кнопке Yes. При копировании секторов на экране появится строка текущего состояния.
12. Выйдите из Disk Editor и попробуйте открыть файл с помощью какой-либо совместимой с файлом программы. Если файл откроется, значит, восстановление данных успешно завершено. Если нет, то, скорее всего, были неправильно определены секторы или файл был фрагментирован.

Программа Norton Disk Editor — мощный инструмент, который используется для проверки накопителей и восстановления утраченных данных. Однако наилучший метод восстановления данных — предотвращение ситуаций, в которых пришлось бы этим заниматься. Поэтому, прежде чем удалить файлы или отформатировать диск, подумайте — а стоит ли игра свеч?

Восстановление данных, удаленных из флэш-памяти

Устройства с флэш-памятью, к которым относятся сменные и USB-накопители, используемые в цифровых фотоаппаратах и аудиопроигрывателях, представляют собой исключительно сложный объект для программ восстановления данных. С точки зрения пользователя, устройства похожи на стандартные накопители, имеют такие же таблицы размещения файлов, как и те, что используются на гибких дисках, и могут быть отформатированы с помощью программы Проводник (Windows Explorer). Тем не менее многие программы восстановления данных, которые отлично работают со стандартными дисковыми носителями, не справляются с восстановлением данных, удаленных из флэш-памяти, — особенно в тех случаях, когда устройство было отформатировано.

Бывают ситуации, которые могут привести к потере данных, хранящихся в устройствах с флэш-памятью. Например, форматирование носителей или удаление одной или нескольких фотографий/файлов часто происходит при подключении устройства к компьютеру через модуль считывания или при вставке карты флэш-памяти в цифровой фотоаппарат. При удалении фотографий расположение файлов и перечень имен в таблицах размещения файлов изменяется точно так же, как и при удалении файлов, хранившихся на магнитных носителях: первый символ файлового имени заменяется строчной сигмой (σ), обозначающей, что файл был удален. Для восстановления файлов, удаленных из флэш-памяти (как и для файлов на магнитных носителях), могут использоваться программы, поддерживающие накопители на сменных носителях, а также Norton Disk Editor. Обратите внимание: Disk Editor должна работать в режиме “Read-

Only”. Эта программа больше подходит для устройств с операционной системой Windows 9x/Me. К повреждению файлов данных может привести также извлечение платы флэш-памяти из устройства до завершения процесса записи данных.

Тем не менее восстанавливать данные из отформатированной флэш-памяти значительно труднее, независимо от того, форматировалась ли она с помощью соответствующей опции цифрового фотоаппарата или операционной системы Windows. Традиционные программы, к числу которых относится Norton Unformat, работающая в режиме командной строки (эта программа является одним из компонентов Norton Utilities и Norton SystemWorks), не могут использоваться для этих целей, так как к устройствам флэш-памяти можно обращаться только через графический интерфейс Windows, а программы с командной строкой предназначены для работы с BIOS-совместимыми устройствами, такими, как накопители на жестких или гибких дисках.

Программы, зависящие от определенной файловой системы, например Ontrack EasyRecovery Personal Edition Lite (включенная в пакет VCOM System's Suite) и Ontrack EasyRecovery Personal Edition, также не подходят, поскольку форматирование флэш-памяти приводит к разрушению предыдущей файловой системы.

Замечание

Цифровой фотоаппарат, выполняя форматирование карты флэш-памяти, обычно создает папку, в которой содержатся фотографии. В некоторых фотоаппаратах могут также создаваться каталоги для хранения драйверов и другой информации.

Если необходимо восстановить данные из отформатированного устройства флэш-памяти, обратите внимание на две программы.

- *Ontrack EasyRecovery Professional Edition*. С обзором и характеристиками программы можно познакомиться на Web-сайте <http://www.ontrack.com>.
- *PhotoRescue*. С обзором и техническими характеристиками программы можно познакомиться на Web-сайте <http://www.datarescue.com/photorescue/>.

Для восстановления данных также пригодится программа Norton Disk Editor, входящая в пакеты Norton SystemWorks и Norton SystemWorks Pro. При этом необходимо определить начальные и конечные кластеры данных, хранящихся во флэш-памяти.

Чтобы восстановить данные из отформатированной карты флэш-памяти с помощью программы EasyRecovery Professional Edition, необходимо воспользоваться функцией RawRecovery, которая позволяет восстанавливать данные посекторно. Эта функция игнорирует файловую систему и может использоваться на носителях любых поддерживаемых типов. Встроенный модуль просмотра файлов позволяет определить, считываются ли восстановленные данные.

Программа PhotoRescue, которая поддерживает только изображения стандартных типов, а именно файлы форматов JPG, BMP и TIFF, может обращаться к носителям в режиме логического диска (который довольно хорошо зарекомендовал себя в ранее описанных тестах) или в режиме физического диска. В последнем используется “посекторный” метод восстановления данных, немного напоминающий технологию, используемую в программе EasyRecovery Professional Edition. Программа PhotoRescue содержит также встроенный модуль для просмотра восстановленных фотографий.

Указанные программы позволяют восстанавливать не только последние файлы или фотографии, сделанные непосредственно перед форматированием, но и данные, оставшиеся во флэш-памяти после ее предыдущего использования. Пока область данных, используемая тем или другим файлом, не будет перезаписана, утраченные данные могут быть восстановлены, даже если устройство было отформатировано несколько раз.

В табл. 25.33 приведены результаты восстановления данных, удаленных из устройств флэш-памяти наиболее распространенных типов: с карты Compact Flash, используемой в цифровых фотоаппаратах, и сменного запоминающего устройства USB.

Таблица 25.33. Восстановление утраченных данных в устройствах с флэш-памятью

Результаты использования программы восстановления данных					
Устройство	Причина потери данных	Norton Utilities	Ontrack/Vcom DataRescue System Suite	DataRescue Photo Rescue	Ontrack EasyRecovery Professional
Compact Flash, 64 Мбайт	Удаление некоторых файлов в фотоаппарате	Запись восстановленных данных в исходный каталог, но только при использовании Windows 9x/Me ^{1,2}	Запись восстановленных данных в каталог, определяемый пользователем ^{1,3}	Восстановление данных после последнего форматирования и использования карты. Данные записываются в определенный каталог ^{3,4}	Восстановление файлов, удаленных при текущем и предыдущем использовании посредством функции RawRecovery; ограничения указаны в табл. 25.31 ^{3,4}
Compact Flash, 64 Мбайт	Удаление некоторых файлов с помощью программы Проводник (Windows Explorer)	Запись восстановленных данных в исходный каталог, но только при использовании Windows 9x/Me ^{1,2}	Запись восстановленных данных в указанный каталог, но только при использовании поддерживаемой версии Windows ³	Восстановление данных после последнего форматирования и использования карты. Данные записываются в определенный каталог, файлы и каталоги переименовываются	Восстановление файлов, удаленных при текущем использовании посредством функции DeletedRecovery (первый символ имени файла/каталога утрачен)
Compact Flash, 64 Мбайт	Форматирование карты в фотоаппарате	Формат накопителя восстановить не удалось. Disk Edit позволила восстановить данные, удаленные при текущем и предыдущем использовании, и записать их в каталог, определяемый пользователем ^{2,5,6}	Данные не были обнаружены и восстановлены	Восстановление данных после последнего форматирования и использования карты. Данные записываются в определенный каталог ^{3,4}	Функцией RawRecovery восстановлены все читабельные данные, в том числе и удаленные после предыдущего использования. Данные записаны в каталог, определенный пользователем ^{3,4}
Compact Flash, 64 Мбайт	Форматирование карты в устройстве для считывания	Формат накопителя восстановить не удалось. Disk Edit позволила восстановить данные, удаленные при текущем и предыдущем использовании ^{5,6}	Данные не были обнаружены и восстановлены	Восстановление данных после последнего форматирования и использования карты. Данные записываются в определенный каталог ^{3,4}	Функцией RawRecovery восстановлены все читабельные данные, в том числе и удаленные после предыдущего использования. Данные записаны в каталог, определенный пользователем ^{3,4}
USB-брелок, 128 Мбайт	Удаление каталога в окне Мой компьютер (My Computer)	Disk Edit позволила восстановить данные, удаленные при текущем использовании ^{3,5}	Частичный успех: восстановлены некоторые файлы ³	Восстановлены только файлы фотографий ^{3,4}	Функцией RawRecovery восстановлена большая часть файлов ^{3,4}
USB-брелок, 128 Мбайт	Форматирование накопителя с помощью программы Проводник (Windows Explorer)	Disk Edit позволила восстановить данные, удаленные при текущем использовании ^{3,5}	Частичный успех: восстановлены некоторые файлы (утрачены структура и имена каталогов) ³	Восстановлены только файлы фотографий ^{3,4}	Функцией RawRecovery восстановлена большая часть файлов ^{3,4}

¹ В процессе восстановления данных пользователь заменил первый символ в имени файла.

² Программа Norton UnErase не поддерживает накопители на сменных носителях в Windows NT/2000/XP.

³ На диске, содержащем потерянные данные, программа функционирует в режиме "только для чтения".

⁴ Исходные имена файлов и каталогов не сохранились; файлы последовательно пронумерованы и могут быть переименованы по завершении восстановления данных.

⁵ Для обращения к устройствам флэш-памяти должна использоваться операционная система Windows. Программа Norton Unformat, работающая в режиме командной строки, не может использоваться в многозадачной среде Windows.

⁶ При работе с программой Disk Edit пользователь должен вручную найти начальные и конечные секторы каждого файла и переписать все секторы на другой накопитель, присвоив файлу какое-нибудь имя.

Наиболее часто встречающиеся сообщения об ошибках

В этом разделе речь идет о наиболее часто встречающихся ошибках файловых систем и способах их устранения.

Missing Operating System

Такое сообщение об ошибке указывает на проблемы в главной загрузочной записи или записях таблицы раздела. Запись в таблице раздела может указывать на сектор, который не является началом раздела. Такая ошибка иногда появляется вследствие разряда батареи на системной плате, что приводит к удалению параметров BIOS.

Для решения проблемы прежде всего необходимо проверить правильность установки параметров в BIOS. Главную загрузочную запись можно восстановить с помощью команды FDISK/MBR. В других случаях решить возникшую проблему можно с помощью радикальных средств — разбивки диска на разделы и форматирования с последующей повторной установкой операционной системы и необходимых приложений.

NO ROM BASIC — SYSTEM HALTED

Эту ошибку генерирует AMI BIOS в случае повреждения или отсутствия загрузочного сектора либо главной загрузочной записи на загрузочном диске. Кроме того, такая ошибка может появиться при неправильной установке параметров жесткого диска в BIOS.

В IBM-системах в таком случае запускается версия BASIC для BIOS, однако большинство производителей других BIOS не лицензируют этот код у компании Microsoft, поэтому вместо перехода в режим BASIC отображается столь загадочное сообщение. Для решения этой проблемы необходимо проверить параметры диска в BIOS или же восстановить главную загрузочную запись с помощью команды FDISK/MBR.

Boot error Press F1 to retry

Эта ошибка генерируется Phoenix BIOS при отсутствии жесткого диска или загрузочных областей. Наиболее частая причина появления этой ошибки — отсутствие активного раздела.

Invalid drive specification

Такая ошибка появляется в том случае, если жесткий диск не разбит на разделы, записи таблицы разделов повреждены или содержат неверные данные. Для устранения подобных проблем воспользуйтесь программой FDISK либо программой Diskedit из пакета Norton Utilities.

Invalid Media Type

При появлении такого сообщения, скорее всего, поврежден (или не инициализирован) загрузочный сектор, каталог или таблица размещения файлов. Например, такая ошибка появляется, если диск разбит на разделы, но не отформатирован с помощью команды Format. Для устранения этой ошибки необходимо использовать одну из программ восстановления диска или же просто отформатировать диск.

Hard Disk Controller Failure

Такого типа ошибка появляется, если в BIOS установлены неправильные параметры накопителя либо плохо подключены кабели к накопителю или системной плате. Для устранения этой проблемы прежде всего проверьте подключение накопителя, а затем установите в BIOS его правильные параметры.

Решение общих проблем файловой системы в MS-DOS, Windows 9x/Me

Чтобы устранить проблемы при доступе к жесткому диску, выполните ряд действий.

1. Загрузите компьютер с загрузочной дискеты (ее иногда называют аварийным диском). Это может быть загрузочная дискета как DOS, так и Windows, главное, чтобы на ней были записаны следующие программы: `Fdisk.exe`, `Format.com`, `Sys.com` и `Scandisk.exe`. Лучше, если это будет аварийный диск с операционной системой Windows 95B или более поздней.
2. Если с загрузочной дискеты нельзя загрузить операционную систему, скорее всего существуют проблемы с аппаратным обеспечением. В некоторых системах гибкий диск не определен как загрузочное устройство. В этом случае необходимо повторно запустить BIOS, указать гибкий диск как первое загрузочное устройство и перезагрузить компьютер.
3. Запустите с загрузочной дискеты программу FDISK. В меню выберите вывод сведений об имеющихся разделах (четвертый пункт меню).
4. Если отображается список разделов, проверьте наличие активного раздела: в столбце состояния возле одного из разделов должна быть буква A.
5. Если в списке не отображается ни одного раздела и вы не желаете восстанавливать данные на диске, создайте новый раздел (или разделы), а затем отформатируйте его. При выполнении этих действий все данные на диске будут уничтожены.
6. Если вам необходимо восстановить данные, воспользуйтесь одной из программ восстановления данных.
7. Если список разделов отображается и один из них активный, очевидно, повреждены системные файлы. Для их восстановления введите команду `Sys C:`.
8. Теперь ваш жесткий диск содержит системные файлы той операционной системы, которая была на загрузочном диске.
9. Извлеките дискету из дисковода и перезагрузите компьютер. Если и сейчас при загрузке появляются ошибки, они, скорее всего, вызваны неверной конфигурацией жесткого диска в BIOS.
10. Запустите программу Scandisk с загрузочного диска и проверьте диск на наличие ошибок.
11. При проверке диска с помощью программы Scandisk не забудьте проверить поверхность диска. При появлении большого количества поврежденных секторов необходимо заменить накопитель на жестких дисках.

Решение общих проблем файловой системы в Windows 2000/XP

Способы решения проблем файловой системы в Windows 2000/XP практически те же, что и в Windows 9x. Их основным отличием является использование служебной программы Recovery Console, входящей в Windows 2000/XP.

- Если программа Recovery Console введена в загрузочное меню, запустите систему в обычном порядке, войдите в систему как администратор, если это необходимо, и выберите опцию Recovery Console.
- В том случае, если программа Recovery Console не была предварительно введена в загрузочное меню, загрузите систему с помощью установочного компакт-диска Windows или диска Windows Setup. Выберите опцию Repair в меню Welcome to Setup и нажмите клавишу <C> для запуска Recovery Console.

Если система не загружается с установочного компакт-диска или загрузочной дискеты, то возможны некоторые проблемы с аппаратным обеспечением. Проверьте жесткие диски, конфигурацию BIOS и установочные параметры системной платы. Определите гибкий диск как первое загрузочное устройство, а CD-ROM — как второе, а затем перезапустите систему.

После запуска Recovery Console выполните ряд действий.

1. Для получения справки и списка команд Recovery Console введите `HELP` в командной строке.
2. Запустите программу `DISKPART`, чтобы получить сведения о существующих разделах диска.
3. Когда отобразится перечень разделов, проверьте, определен ли загрузочный раздел как активный.
4. В том случае, если в списке не отображены имеющиеся разделы и у вас нет желания восстанавливать какие-либо данные, существующие в настоящее время на диске, создайте новый раздел (или разделы) с помощью команды `FDISK`. Для форматирования созданных разделов воспользуйтесь командой `FORMAT`. При выполнении этих действий все данные на диске будут уничтожены.
5. Если необходимо восстановить данные, воспользуйтесь для этого одной из существующих программ восстановления данных, например Norton Utilities от компании Symantec или Lost and Found от PowerQuest.
6. Если при выполнении программы `DISKPART` отображается список разделов и один из них определен как активный, значит, возможно повреждение системных файлов. Для их восстановления введите команду `FIXBOOT`.
7. Для того чтобы перезапустить систему, введите команду `EXIT`. Перед этим извлеките загрузочную дискету из дисководов A: или установочный компакт-диск Windows 2000 из дисководов CD-ROM.
8. Если и после перезагрузки компьютера появятся те же ошибки, то они, скорее всего, вызваны повреждением или неверной конфигурацией жесткого диска.
9. Перезапустите Recovery Console и проверьте диск на наличие ошибок, введя команду `CHKDSK` в командной строке.