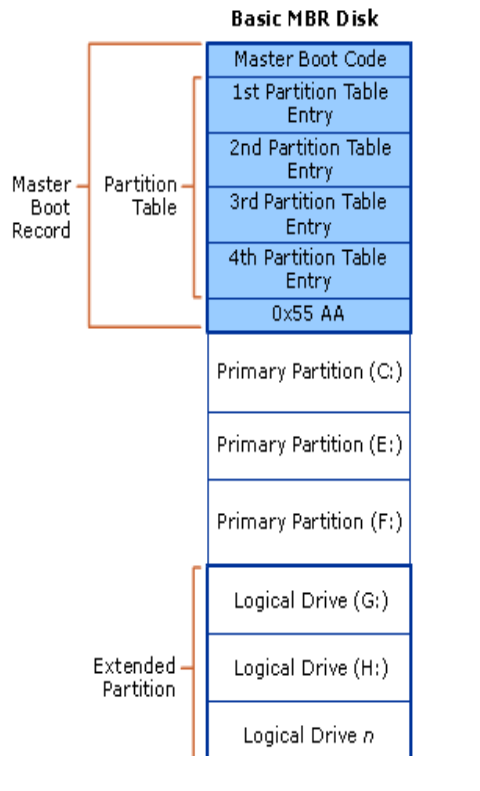
**Контрольные вопросы**

1. Место для скольких записей предусмотрено в стандартной таблице разделов главной загрузочной записи (mbr)?



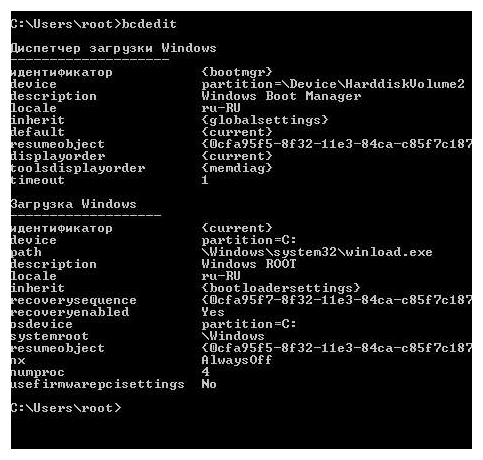
**4 записи**

1. Как можно определить тип установленной на разделе диска файловой системы программно?

Windows: утилита diskpart, PowerShell-getVolume, diskmgmt.msc

Linux: командs fsck, file, df -T, mount

1. Каким образом определяется раздел с операционной системой для загрузки?

cmd: bcdedit (поле device)

1. Перечислите основные объекты файловой системы FAT, где и в каком порядке они расположены?

В файловой системе FAT смежные секторы диска объединяются в единицы, называемые кластерами. Количество секторов в кластере равно степени двойки (см. далее). Для хранения данных файла отводится целое число кластеров (минимум один), так что, например, если размер файла составляет 40 байт, а размер кластера 4 Кбайт, реально занят информацией файла будет лишь 1 % отведённого для него места. Во избежание подобных ситуаций целесообразно уменьшать размер кластеров, а для сокращения объёма адресной информации и повышения скорости файловых операций — наоборот. На практике выбирают некоторый компромисс. Так как ёмкость диска вполне может и не выражаться целым числом кластеров, обычно в конце тома присутствуют так называемые surplus sectors — «остаток» размером менее кластера, который не может отводиться ОС для хранения информации.

Пространство тома FAT32 логически разделено на три смежные области:

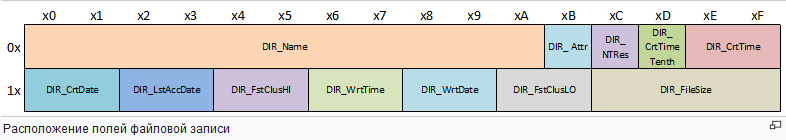
* Зарезервированная область. Содержит служебные структуры, которые принадлежат загрузочной записи раздела (Partition Boot Record — PBR, для отличия от Master Boot Record — главной загрузочной записи диска; также PBR часто некорректно называется загрузочным сектором) и используются при инициализации тома;
* Область таблицы FAT, содержащая массив индексных указателей («ячеек»), соответствующих кластерам области данных. Для повышения надёжности на диске обычно представлено две копии таблицы FAT;
* Область данных, где записано собственно содержимое файлов — то есть текст текстовых файлов, кодированное изображение для файлов рисунков, оцифрованный звук для аудиофайлов и т. д.

В FAT12 и FAT16 также специально выделяется область корневого каталога. Она имеет фиксированное положение (непосредственно после последнего элемента таблицы FAT) и фиксированный размер в 32-байтных элементах, то есть при описании в Partition Boot Record указывается именно количество 32-байтных элементов, каждый из которых описывает какой-либо элемент корневого каталога (будь то файл или другой вложенный каталог).

Первая структура тома FAT называется BPB (англ. BIOS parameter block) и расположена в зарезервированной области, в нулевом секторе. Эта структура содержит информацию, идентифицирующую тип файловой системы и физические характеристики носителя (дискеты или раздела на жёстком диске).

Следующая важная структура тома FAT — сама таблица FAT, занимающая отдельную логическую область. Она определяет список (цепочку) кластеров, в которых размещаются файлы и папки тома. Между кластерами и индексными указателями таблицы имеется взаимно однозначное соответствие — N-й указатель соответствует кластеру с тем же номером. Первому кластеру области данных присваивается номер 2. Значение индексного указателя соответствует состоянию соответствующего кластера.

Непосредственно после окончания последней таблицы FAT следует область данных, содержащая файлы и папки. Каталог FAT является обычным файлом, помеченным специальным атрибутом. Данными (содержимым) такого файла в любой версии FAT является цепочка 32-байтовых файловых записей (записей каталога). Каталог не может штатно содержать два файла с одинаковым именем. Если программа проверки диска обнаруживает искусственно созданную пару файлов с идентичным именем в одном каталоге, один из них переименовывается.



1. Какой максимальный размер раздела поддерживает каждый из вариантов FAT (FAT12, FAT16, FAT32, exFAT)?

FAT12: 32 MB (256 MB for 64 KB clusters)

FAT16: 4095 Мб

FAT32: 127 Гб

exFAT: 2­64 байт (16Экзабайт)

1. Как в FAT задаются каталоги?

Файловый атрибут 0x10 — свидетельствует, что файл является каталогом.

При создании каталога для него «пожизненно» выставляется DIR\_FileSize = 0. Размер содержимого каталога определяется простым следованием по цепочкам кластеров до метки End Of Chain. Размер самого каталога лимитируется файловой системой в 65 535 32-байтовых записей (то есть записи каталога в таблице FAT не могут занимать более 2 Мбайт). Это ограничение призвано ускорить операции с файлами и позволить различным служебным программам использовать 16-битное целое (WORD) для подсчёта количества записей в каталоге (как следствие, возникает теоретическое ограничение на количество файлов в каталоге — 65 535 при условии, что все имена файлов следуют стандарту 8.3). Каталогу отводится один кластер области данных (за исключением случая, если это корневой каталог FAT12/FAT16), и полям DIR\_FstClusHI / DIR\_FstClusLO присваивается значение номера этого кластера. В таблицу FAT для записи, соответствующей этому кластеру, помещается метка EOC, а сам кластер забивается нулями. Далее создаются два специальных файла, без которых каталог FAT считается повреждённым (первые две 32-байтовых записи в области данных кластера) — файлы нулевого размера с именами «.» (одна точка, идентификатор каталога) и «..» (две точки, указатель на родительский каталог). Отметки даты и времени этих файлов приравниваются значениям для самого каталога на момент создания и не обновляются при изменениях каталога. Поля DIR\_FstClusHI / DIR\_FstClusLO файла «.» содержат значение номера содержащего его кластера, а файла «..» — номера первого кластера каталога, содержащего данный. Таким образом, файл «.» отсылает к самому каталогу, а файл «..» — к начальному кластеру родительского каталога; если родительский каталог — корневой, начальным кластером считается нулевой.

1. За счет чего происходит потеря объема диска при большом размере кластеров? Поясните на примере.

Для хранения данных файла отводится целое число кластеров (минимум один), так что, например, если размер файла составляет 40 байт, а размер кластера 4 Кбайт, реально занят информацией файла будет лишь 1 % отведённого для него места.

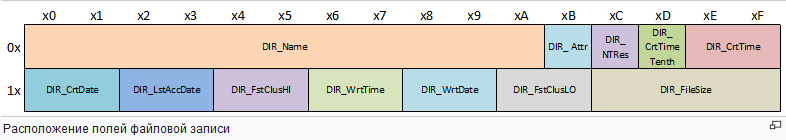
1. Каким образом в FAT помечается удаленный файл?

При удалении файла первый знак имени заменяется специальным кодом 0xE5, и цепочка кластеров файла в таблице размещения обнуляется.

1. Что такое длинное имя и как оно задается в FAT?

Имя большее чем 8.3 (8символов имени, точка, 3 символа расширения). Файлы и каталоги с длинным именем (свыше 8.3) обрабатываются файловой системой FAT особым образом. Структура 32-байтовой записи для файла с LFN (Long File Name) отличается от обычной (SFN-записи) - 255 символов при использовании LFN:

* LDIR\_Ord. Первый байт записи служит для нумерации записей в наборе.
* LDIR\_Name1. 10-байтовое поле по адресу 0х01 содержит первые пять символов имени файла (вернее, той части его имени, которая отражена в данной LFN-записи).
* LDIR\_Attr. Байт атрибута по адресу 0х0B, равен 0х0F (ATTR\_LONG\_NAME).
* LDIR\_Type. Байт по адресу 0х0C, обнулён и дополнительно свидетельствует, что данная запись таблицы FAT относится к файлу с длинным именем.
* LDIR\_Chksum. Байт по адресу 0х0D, содержит контрольную сумму SFN псевдонима файла, соответствующего набору LFN-записей.
* LDIR\_Name2. 12-байтовое поле по адресу 0х0E, содержащее 6-й и 11-й символы имени файла.
* LDIR\_FstClusLO. 2-байтовое поле по адресу 0х1A, в контексте LFN-записи лишено смысла и обнуляется.
* LDIR\_Name3. 4-байтовое поле по адресу 0х1C, содержащее 12-й и 13-й символы имени файла.



1. Каким образом можно восстановить удаленный файл (файловая система FAT)?

При удалении файла первый знак имени заменяется специальным кодом 0xE5, и цепочка кластеров файла в таблице размещения обнуляется. Поскольку информация о размере файла (которая располагается в каталоге рядом с именем файла) при этом остаётся нетронутой, в случае, если кластеры файла располагались на диске последовательно и не были перезаписаны новой информацией, удалённый файл можно восстановить.

1. Почему до восстановления нежелательно записывать информацию на раздел с удаленным файлом?

*Создание файла или каталога* командой «Создать» контекстного меню — создаётся файловая запись для нового «пустого» файла с именем по умолчанию (например «Новая папка») и размером, определяемым типом файла; сам файл, если имеет ненулевой размер (что верно для практически всех «пустых» файлов, кроме каталогов и текстовых документов) записывается в области данных в выделенные ему кластеры; в таблице FAT создаётся соответствующая кластерная цепочка. После присвоения файлу действительного имени (не по умолчанию) первоначально созданная файловая запись помечается как удалённая и создаётся новая.

1. Есть ли разница между удаленными файлом и каталогом?

Windows не анализирует остаток каталога после обнулённой записи.

1. Почему NTFS считается более надежной файловой системой, чем FAT?
2. Какие еще дополнительные возможности есть у файловой системы NTFS по сравнению с файловой системой FAT?

NTFS способна адресовывать до 264 кластеров. NTFS — журналируемая, основанная на транзакциях ФС, обладающая функциями самовосстановления. По сравнению с FAT обладает такими свойствами как:

Быстрая скорость доступа к файлам малого размера.

Разграничение прав доступа к файлам и шифрование.

Автоматическое восстановление системы при любых сбоях.

Эффективное хранение данных.

Высокая производительность при работе с крупными массивами данных и большими каталогами.

Размеры кластера могут быть очень маленькими (512 байт).

**Существенные функции:**

• **Разреженные файлы**. Это файлы, очень большие логически, но занимающие на диске только необходимый объем (например, благодаря этой технологии на разделе в 2Гб может быть сохранен разреженный файл объемом 32Гб). Эта технология используется самой NT.

• **Поддержка коротких имен** для совместимости с MS-DOS-программами. Каждый раз при создании файла NTFS делает дополнительную запись в MFT, содержащую короткий эквивалент имени в стандарте 8.3.

• **Компрессия файлов и каталогов**. NTFS обеспечивает динамическое, прозрачное для приложений сжатие файлов и каталогов на манер MS-DOS-утилит DriveSpace и Stack. Атрибут Сжатый можно установить как для всего тома, так и для отдельных файлов и каталогов. Сжатие возможно на разделах с кластером, не превышающим 4 Кб. Степень сжатия варьируется в зависимости от типа данных и максимальна для текстовых документов и файлов, созданных с помощью MS Office.

• **Многопоточные файлы**. Один и тот же файл может содержать несколько именованных потоков, содержащих разную информацию, причем размер файла высчитывается согласно содержимого главного, безымянного потока. Писать в потоки можно, например, с помощью перенаправления ввода-вывода:

Echo «какой-то текст» > File.txt:First.

Аналогично читаем:

More < File.txt:First.

В этом примере у файла File.txt создается дополнительный поток с именем «First» в который записывается, и затем считывается текстовая строка.

Внимание! Данная функция поддерживается только в NTFS, и при копировании на тома с другой ФС информация в именованных потоках пропадет.

**• Жесткие связи (hardlink)**. Для одного и того же файла можно создать несколько имен внутри тома. При этом мы не увеличиваем количество файлов, а лишь делаем своеобразный ярлык - запись в файловой системе, ссылающуюся на исходный файл. Файл остается на диске до тех пор, пока не удалят последнюю жесткую связь на него. Эта и 2 последующие технологии давно используются в UNIX-системах.

• **Точки переопределения (reparse point)**. Любой файл или каталог может быть точкой переопределения, т.е. указывать на другой каталог или файл. Это способ представления имен системой ввода/вывода. Простейшим примером может быть монтирование диска D: в каталог C:\Disks\D\. В итоге, входя в этот каталог, мы попадем на диск D:, хотя путь не изменится. Создание точки переопределения можно выполнить, например, с помощью утилиты junction. Доступны только на NTFS 5.0, 5.1 и старше.

• **Квотирование дискового пространства**. Можно создать квоту на используемое пространство диска. В итоге пользователь не сможет занять пространнство диска большее, чем ему позволено. Данная функция появилась в Windows 2000.

В дополнение ко всему вышесказанному: существующий том с FAT можно преобразовать в NTFS без потери данных с помощью команды CONVERT, однако эффективность такого решения не очень высокая из-за особенностей процесса преобразования.

1. Что такое точка переопределения? Как она может быть создана?

• **Точки переопределения (reparse point)**. Любой файл или каталог может быть точкой переопределения, т.е. указывать на другой каталог или файл. Это способ представления имен системой ввода/вывода. Простейшим примером может быть монтирование диска D: в каталог C:\Disks\D\. В итоге, входя в этот каталог, мы попадем на диск D:, хотя путь не изменится. Создание точки переопределения можно выполнить, например, с помощью утилиты junction. Доступны только на NTFS 5.0, 5.1 и старше.

1. Что означает понятие «разреженный файл» в NTFS?

**Разреженные файлы**. Это файлы, очень большие логически, но занимающие на диске только необходимый объем (например, благодаря этой технологии на разделе в 2Гб может быть сохранен разреженный файл объемом 32Гб). Эта технология используется самой NT.

1. Каким образом можно создать число разделов диска, большее, чем количество букв в английском алфавите? И как получить к ним доступ?
2. Что такое потоки (streams) в файловой системе NTFS?

• **Многопоточные файлы**. Один и тот же файл может содержать несколько именованных потоков, содержащих разную информацию, причем размер файла высчитывается согласно содержимого главного, безымянного потока. Писать в потоки можно, например, с помощью перенаправления ввода-вывода:

Echo «какой-то текст» > File.txt:First.

Аналогично читаем:

More < File.txt:First.

В этом примере у файла File.txt создается дополнительный поток с именем «First» в который записывается, и затем считывается текстовая строка.

Альтернативные потоки данных (англ. Alternate Data Streams, ADS) — метаданные, связанные с объектом файловой системы NTFS. В файловой системе NTFS файл, кроме основных данных, может также быть связан с одним или несколькими дополнительными потоками данных. При этом дополнительный поток может быть произвольного размера, в том числе может превышать размер основного файла.

1. Что обозначают термином "монтирование файловой системы" в Linux?

Для обозначения добавления устройства к файловой системе исплользуется термин *монтирование (mounting)*. Linux автоматически монтирует корневую (/) файловую систему. Отдельно может присутствовать файловая система /boot, в которой расположены загрузочные файлы ядра. Linux также монтирует некоторые особые файловые системы. Область свопинга не показывается как часть файловой системы, но управляется ядром. Однако, другие особые файловые системы, такие как proc, видны как нормальная часть файловой системы, а ее содержимое можно обрабатывать, как обычные файлы.

1. Каким образом для файловой системы в Linux можно запретить обновлять время доступа к файлам и каталогам (access time)?

Флаги mountfags системного вызова mount(): MS\_NOATIME для файлов, MS\_NODIRTIME для каталогов.

1. Как можно отформатировать USB диск в FAT или NTFS из Linux?
2. Определить имя USB (sudo fdisk -l, найти USB по размеру)
3. Размонтировать USB (sudo umount /dev/sdb1)

Дальше отформатировать флешку очень просто. Для этого используется утилита **mkfs**. Синтаксис команды очень прост:

**$ sudo mkfs -t файловая\_система устройство**

В качестве файловой системы может использоваться любая поддерживаемая системой, например, ext4, ntfs, xfs, btrfs, vfat. Но для нормальной работы флешки в Windows рекомендуется использовать файловую систему vfat. С помощью опции **-L** можно задать метку раздела. Тогда команда будет выглядеть вот так:

sudo mkfs -t ext4 -L FLASH /dev/sdb1

Для файловой системы vfat опция **-L** не поддерживается, вместо неё нужно использовать **-n**:

sudo mkfs -t vfat -n FLASH /dev/sdb1

После того, как форматирование флешки в Linux завершится, флешка будет готова к использованию.

1. Как в Linux узнать количество свободного места в определенной файловой системе?

…:~$ df –h, …:~$ df -m -h – дисковое пространство в Гб, -m – в Мб

…:~$ du -h

1. Каким образом в Linux можно узнать сколько места на диске занимает определенный каталог (файловая система)?

…:~$ du path path – путь к каталогу