

Лекция 2.

Файлы и файловые системы

“Операционные системы”-2019, ИВК, УлГТУ.

Введение

Система управления файлами является основной в абсолютном большинстве современных операционных систем.

Через файловую систему связываются по данным многие системные обрабатывающие программы.

С помощью этой системы решаются проблемы централизованного распределения дискового пространства и управления данными.

Наконец, пользователи получают более простые способы доступа к своим данным, которые они размещают на устройствах внешней памяти.

Файл и файловая система

Под **файлом** понимают именованный набор данных, организованных в виде совокупности записей одинаковой структуры.

Для управления этими данными создаются соответствующие файловые системы.

Файловая система предоставляет возможность иметь дело с логическим уровнем структуры данных и операций, выполняемых над данными в процессе их обработки.

Именно файловая система определяет способ организации данных на диске или на каком-нибудь ином носителе.

Функции и назначение системы управления файлами

Системы управления файлами отвечают за:

- создание, уничтожение, организацию, чтение, запись, модификацию и перемещение файловой информации;
- за управление доступом к файлам и за управление ресурсами, которые используются файлами.

Назначение системы управления файлами — предоставление более удобного доступа к данным, организованным как файлы, то есть вместо низкоуровневого доступа к данным используется логический доступ с указанием имени файла и записи в нем.

Файловая система vs Система управления файлами

Файловая система определяет, прежде всего, принципы доступа к данным, организованным в файлы.

А термин **система управления файлами** следует употреблять по отношению к конкретной реализации файловой системы, то есть это — комплекс программных модулей, обеспечивающих работу с файлами в конкретной операционной системе.

В одной ОС может быть реализовано несколько систем управления файлами.

Иерархия данных

При работе с файлами необходимо ввести механизмы структурирования.

Самый простой такой механизм – построение иерархии.

Для этого достаточно ввести понятия **каталога** (*directory*).

Каталог содержит информацию о данных, организованных в виде файлов.

И.. сам файл тоже может являться каталогом. Его зачастую называют **подкаталогом** (*subdirectory*).

Корневой каталог – специальная системная структура данных.

Иерархия данных

Microsoft Windows:

- C:
 - \Users
 - \Windows
 - \System32
 - \explorer.exe
- D:
 - \music
 - \games
 - \test.txt

Иерархия данных

Linux:

- /
 - /home
 - /user
 - /file.txt
 - /root
 - /etc
 - /drivers
 - /fstab
 - /usr

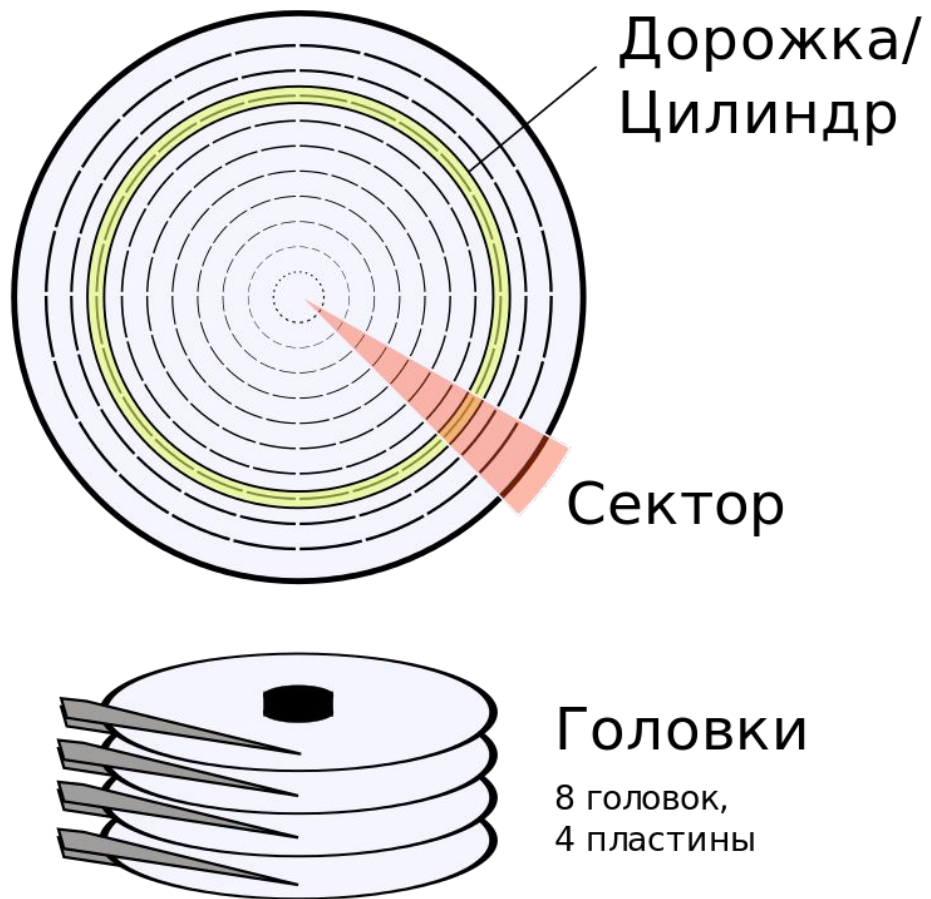
CHS

– система адресации сектора как минимальной единицы хранения данных.

Сектор адресуется кортежем из трёх координат: цилиндр-головка-сектор, именно так, как он физически расположен на диске.

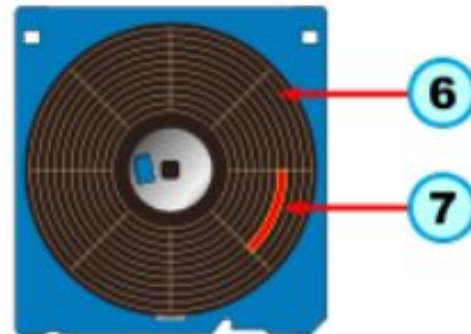
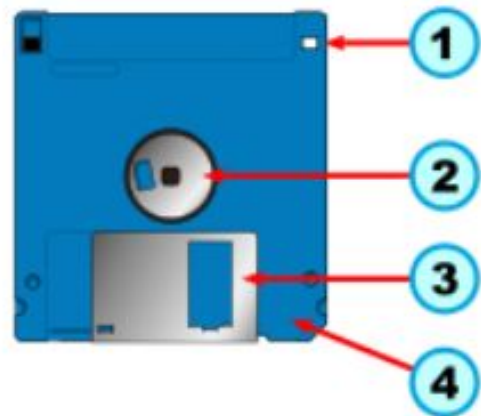
Цилиндр и головка считаются от нуля, а сектор - от единицы.

То есть, первый сектор диска в формате CHS будет иметь адрес (0, 0, 1).



Дискета 3.5"

Устройство дискеты 3½"



1 — окошко, определяющее плотность записи (на другой стороне — переключатель защиты от записи); 2 — основа диска с отверстиями для приводящего механизма; 3 — защитная шторка открытой области корпуса; 4 — пластиковый корпус дискеты; 5 — антифрикционная прокладка; 6 — магнитный диск; 7 — область записи (красным условно выделен один сектор одной дорожки).

Дискета 3.5"

Самый распространённый объём – 1440 Кбайт (высокая плотность).

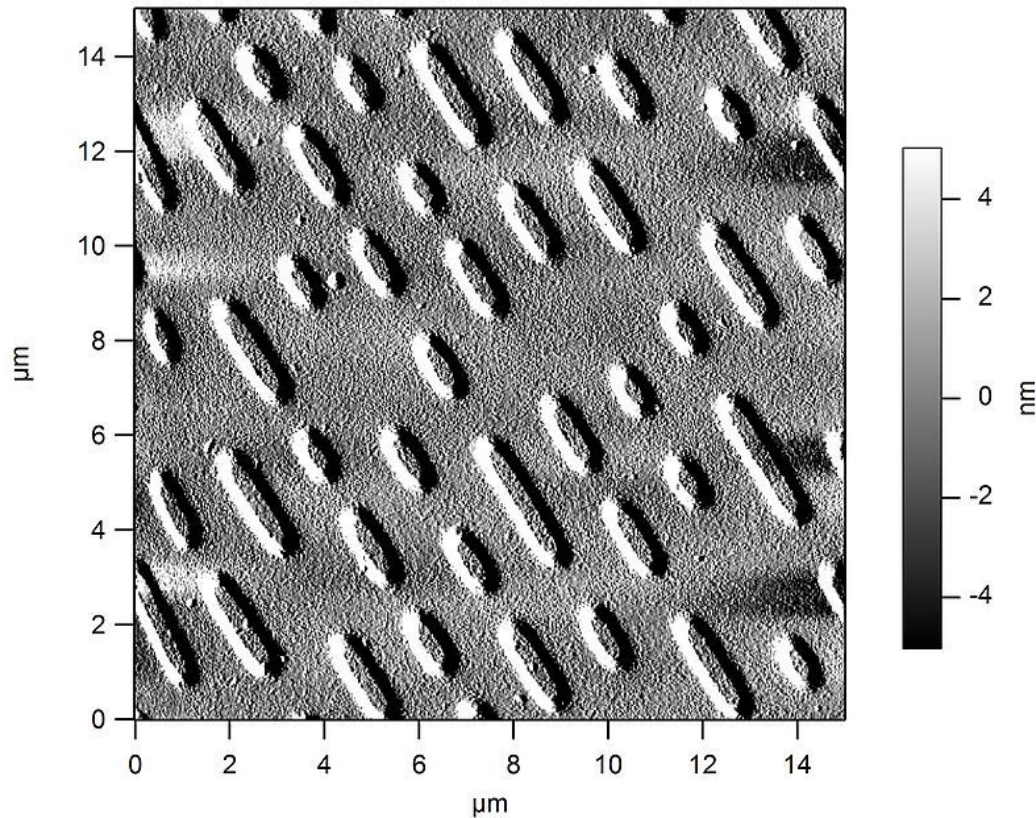
2 головки, 80 дорожек на каждой стороне, 18 секторов на дорожке, 1 сектор в кластере, 512 байт на сектор (стандарт для IBM PC).

Получается $2 * 80 * 18 * 512 = 1\,474\,560$ байт, или 1 440 Кб, или 1,44 МиБ.

Компакт-диск

Информация записывается в виде спиральной дорожки.

333 000 секторов для 650 Мб,
360 000 для 700 Мб.



LBA

LBA (англ. Logical block addressing) — механизм адресации и доступа к блоку данных на жёстком или оптическом диске, при котором системному контроллеру нет необходимости учитывать геометрию самого жесткого диска (количество цилиндров, сторон (головок), секторов на дорожке).

Кортежи CHS можно преобразовать в адреса LBA и обратно по следующим формулам:

$$LBA(c, h, s) = (c \cdot H + h) \cdot S + s - 1$$

$$s = (LBA \bmod S) + 1$$

$$h = \frac{LBA - (s - 1)}{S} \bmod H$$

$$c = \frac{LBA - (s - 1) - h \cdot S}{H \cdot S}$$

где c — номер цилиндра, h - номер головки, s - номер сектора, H — число головок, S — число секторов на дорожке, mod — операция взятия остатка от деления.

Файловая система FAT

FAT (англ. File Allocation Table — «таблица размещения файлов») — классическая архитектура файловой системы.

Разработана Б. Гейтсом и М. МакДональдом в 1977 году, однако из-за своей простоты её модификации всё ещё широко используются для флеш-накопителей.

Существует 3 модификации стандарта: FAT12, FAT16, FAT32.

Номер стандарта: ECMA-107. Всего 55 листов.

Кластеры и размер кластеров в FAT

В файловой системе FAT смежные секторы диска объединяются в единицы, называемые **кластерами**.

Для хранения данных файла отводится целое число кластеров.

Может случиться, что размер файла намного меньше размера кластера. Тогда место на диске будет потрачено впустую.

Для избежания подобных ситуаций целесообразно уменьшать размер кластеров, а для сокращения объёма адресной информации и повышения скорости файловых операций — наоборот.

Логические области ФС FAT

Пространство тома FAT32 логически разделено на три смежные области:

1. Зарезервированная область. Содержит служебные структуры, которые принадлежат загрузочной записи раздела (Partition Boot Record — PBR) и используются при инициализации тома;
2. Область таблицы FAT, содержащая массив индексных указателей ("ячеек" по 32 бита), соответствующих кластерам области данных. Обычно на диске представлено две копии таблицы FAT в целях надёжности;
3. Область данных, где записано собственно содержимое файлов.

Возможные значения в таблице FAT

Если кластер принадлежит файлу, то соответствующая ему ячейка в таблице FAT содержит номер следующего кластера этого же файла.

Если ячейка соответствует последнему кластеру файла, то она содержит специальное значение (0xFFFF для FAT16). Таким образом выстраивается цепочка кластеров файла. EOC (*End of Clusterchain*). Больше 0xFFFFFFFF8 для FAT32.

Неиспользуемым кластерам в таблице соответствуют нули.

«Плохим» кластерам (которые исключаются из обработки, например, по причине нечитаемости соответствующей области устройства) также соответствует специальный код (0xFFF7 для FAT16).

Partition Boot Record

Первая структура тома FAT называется BPB (*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).

В FAT32 задействованы также секторы 1 и 2.

В секторе 1 находится структура FSInfo.

Смысл введения FSInfo в оптимизации работы системы, так как в FAT32 таблица индексных указателей может иметь значительные размеры и её побайтовый просмотр может занять значительное время.

Таблица FAT

Следующая важная структура тома FAT — это 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.

Первые 2 указателя в таблице FAT

Формально $FAT[0] = EOC$ (на самом деле туда копируется часть структуры из нулевого сектора).

$FAT[1] = EOC$ на момент форматирования тома.

Однако, в $FAT[1]$ могут быть записаны служебные флаги о необходимости проверки тома (“грязный бит”). На самом деле это верхние 2 бита индексного указателя.

“Грязный бит” формируется при некорректном отключении тома или при аппаратной ошибке носителя и соответственно принимает два возможных значения.

Область данных FAT. Каталог

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-байтных файловых записей (записей каталога). Каталог не может штатно содержать два файла с одинаковым именем.

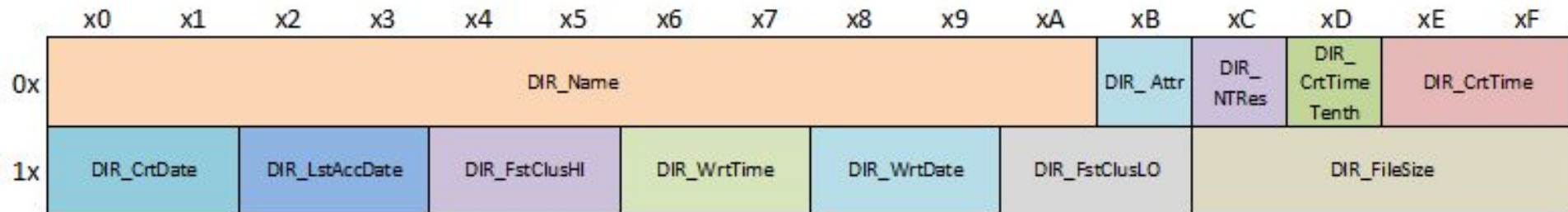
Корневой каталог FAT

Первая запись после таблицы FAT – корневой каталог. Он является единственным обязательным каталогом.

Однако, по сравнению с другими каталогами, в корневом каталоге нет:

- имени;
- меток даты и времени;
- файлов “.” и “..”.

Структура файловой записи FAT



DIR_Name. 11-байтное поле, содержит короткое имя файла.

DIR_Attr. Байт по адресу 0x0B, отвечающий за атрибуты файла.

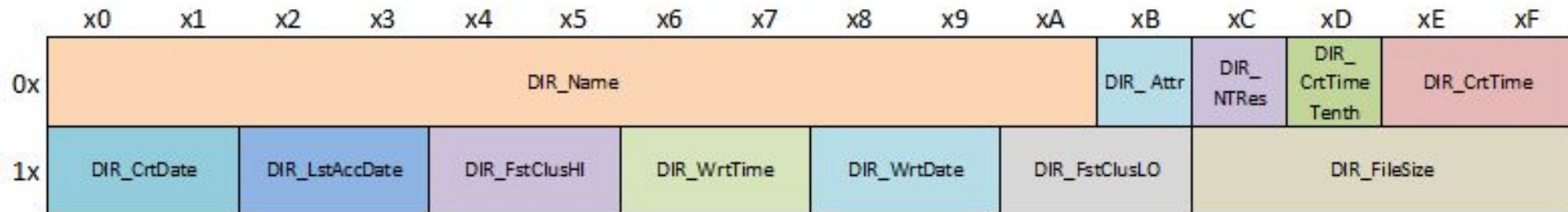
DIR_NTRes. Байт по адресу 0x0C, используется в Windows NT.

DIR_CrtTimeTenth. Байт по адресу 0x0D. Счётчик десятков миллисекунд времени создания файла, допустимы значения 0-199.

DIR_CrtTime. 2 байта по адресу 0x0E. Время создания файла с точностью до 2 секунд.

DIR_CrtDate. 2 байта по адресу 0x10. Дата создания файла.

Структура файловой записи FAT



DIR_LstAccDate. Дата последнего доступа к файлу. Аналогичное поле для времени не предусмотрено.

DIR_FstClusHI. Номер первого кластера файла (старшее слово).

DIR_WrtTime. Время последней записи (модификации) файла.

DIR_WrtDate. Дата последней записи файла, в том числе создания.

DIR_FstClusLO. Номер первого кластера файла (младшее слово).

DIR_FileSize. DWORD, содержащий значение размера файла в байтах.

Имена файлов FAT. Стандарт 8.3

C:\WINDOWS\system32\cmd.exe

Серийный номер тома: 4857-DE34

Содержимое папки c:\

11.06.2018	11:13	<DIR>	AMD
05.10.2018	21:27	<DIR>	cygwin64
15.09.2018	12:05	<DIR>	downloads
13.04.2017	19:48	<DIR>	ESD
02.03.2019	17:31		HaxLogs.txt
11.06.2018	11:54	<DIR>	inetpub
09.10.2015	21:02	<DIR>	Intel
16.01.2017	18:35	<DIR>	oracle
12.04.2018	03:38	<DIR>	PerfLogs
05.01.2019	10:23	<DIR>	Program Files
20.12.2018	18:31	<DIR>	Program Files (x86)
16.07.2016	23:37	<DIR>	Python34
02.09.2014	20:03	<DIR>	totalcmd
11.06.2018	11:14	<DIR>	Users
17.02.2019	19:03	<DIR>	Windows
30.07.2016	15:10		windows-version.txt
10.02.2015	21:03	<DIR>	WinSCP

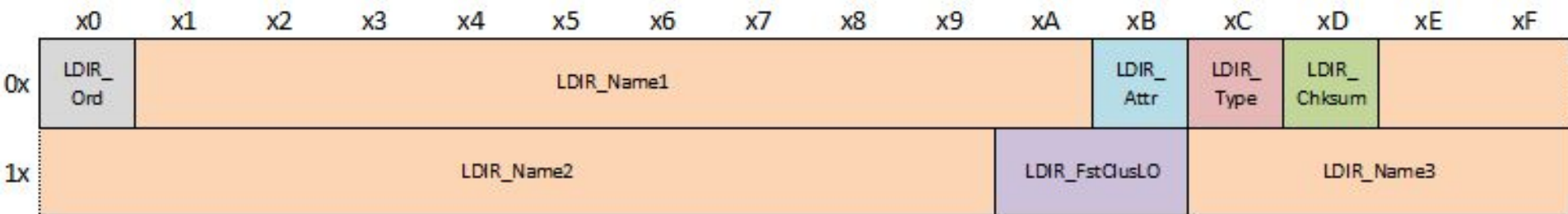
2 файлов 150 байт
15 папок 4 774 649 856 байт свободно

c:\>

LFN-запись

Файлы и каталоги с длинным именем (свыше 8.3) обрабатываются файловой системой FAT особым образом. Структура 32-байтной записи для файла с LFN (*Long File Name*) отличается от обычной (SFN-записи).

Набор LFN-записей каталога FAT всегда должен быть связан с обычной SFN-записью, которой физически предшествует на диске. Набор LFN-записей, обнаруженный без соответствующей обычной записи, называется орфаном и запись считается повреждённой.



Операции в файловой системе FAT32

Форматирование тома — таблица индексных указателей обнуляется, за исключением первых трёх.

Удаление файла — первый символ файловой записи и всех ассоциированных LFN-записей заменяется кодом 0xE5; занимаемые файлом кластеры помечаются в таблице FAT как свободные, а в области данных не затрагиваются.

Переименование файла — создаётся новая запись с обновлённым именем; старая запись помечается как удаленная.

Сохранение файла — создаётся запись, содержащая все поля, кроме размера и начального кластера файла; после завершения сохранения файла создаётся новая запись, содержащая все поля, а прежняя удаляется.

Отказоустойчивость ФС

Отказоустойчивость файловой системы FAT32 достигается тем, что операция создания файла выполняется в одно действие: нет разделения на “добавить запись о занятом кластере” и “убрать запись о свободном кластере”.

Таким образом, при потере питания файловая система останется целостной.

Однако, это не гарантирует целостность файлов.