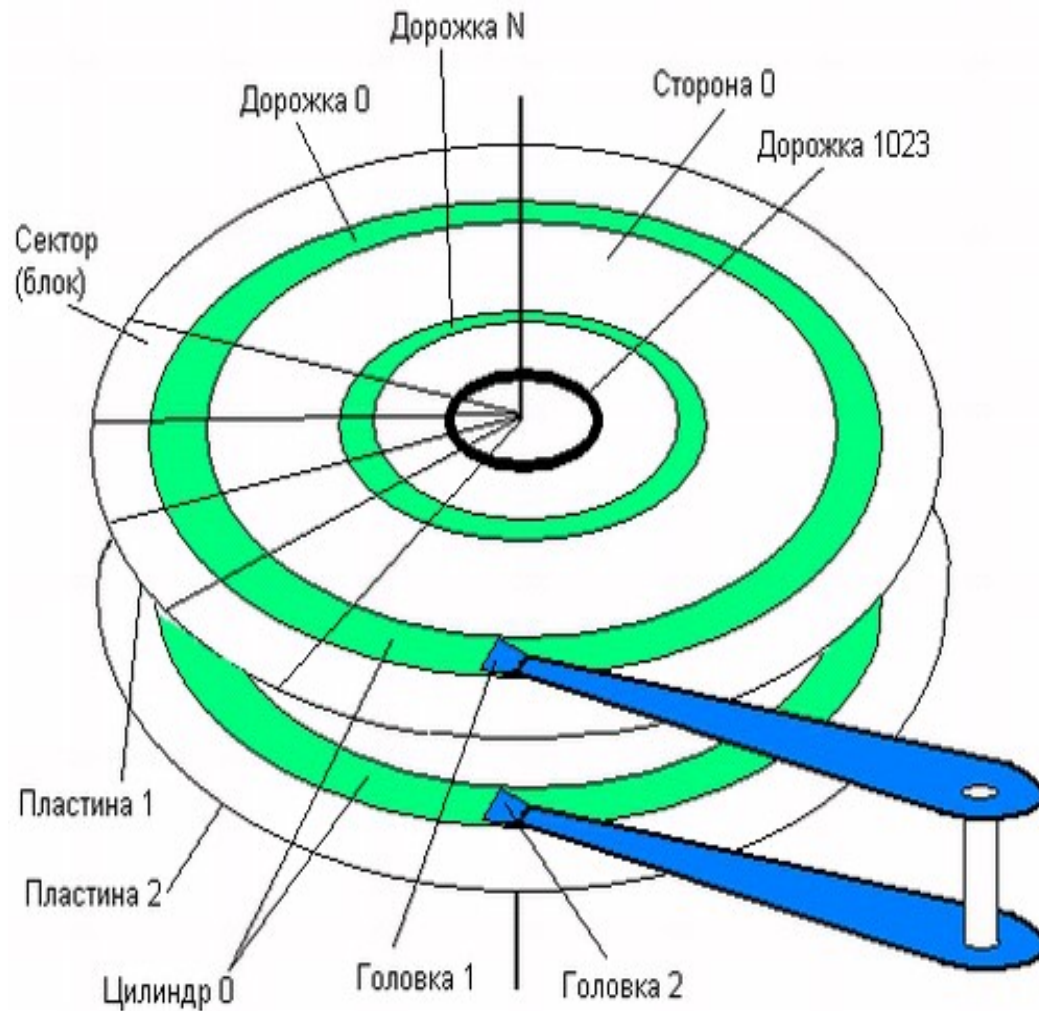


# Лекция №4

Файловые системы

# Носители информации



Структура диска:  
(А) дорожка  
(В) геометрический сектор  
(С) сектор дорожки  
(D) кластер

# Способы адресации

## **CHS (*cylinder-head-sector*)**

При этом способе сектор адресуется по его физическому положению на диске 3 координатами — *номером цилиндра, номером головки и номером сектора*. В дисках, объёмом больше 528 482 304 байт (504 Мб), со встроенными контроллерами эти координаты уже не соответствуют физическому положению сектора на диске и являются «логическими координатами»

## **LBA (*linear block addressing*)**

При этом способе адрес блоков данных на носителе задаётся с помощью логического линейного адреса.

$$\text{LBA} = [ (\text{Cylinder} * \text{no of heads} + \text{heads}) * \text{sectors/track} ] + (\text{Sector}-1)$$

# Задачи ОС

Подсистема ввода-вывода (Input-Output Subsystem) мультипрограммной ОС при обмене данными с внешними устройствами компьютера должна решать ряд общих задач, из которых наиболее важными являются следующие:

- организация параллельной работы устройств ввода-вывода и процессора;
- согласование скоростей обмена и кэширование данных;
- разделение устройств и данных между процессами;
- обеспечение удобного логического интерфейса между устройствами и остальной частью системы;
- поддержка широкого спектра драйверов с возможностью простого включения в систему нового драйвера;
- динамическая загрузка и выгрузка драйверов;
- поддержка нескольких файловых систем;
- поддержка синхронных и асинхронных операций ввода-вывода.

# Носители информации

Диск (Disk)

Том (Volume)

Каталог (Folder)

Файл (File)

Кластер (Cluster)

Байты, биты

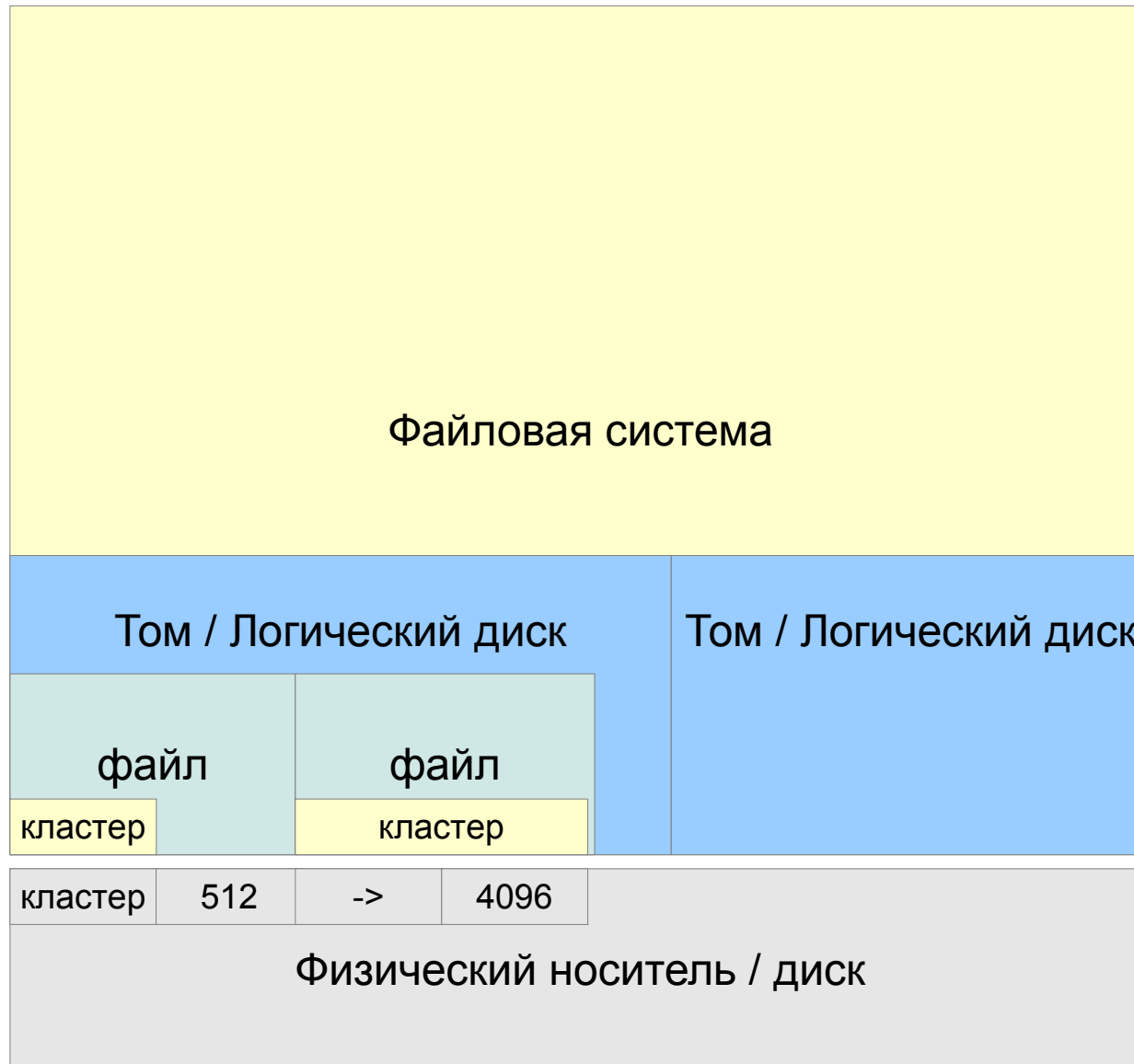
Таблица разделов

MBR

Boot Sector

Partition Table

# Носители информации



# Носители информации

Первый сектор жесткого диска (а точнее нулевой) носит название MBR (Master Boot Record) или главной загрузочной записи. В начале этого сектора находится код, куда передает управление базовая система ввода вывода компьютера при его загрузке. В дальнейшем этот код передает управление загрузчику операционной системы. Так же в 0 секторе находится таблица разделов жёсткого диска. Раздел представляет собой определённый диапазон секторов. В таблицу заносится запись о разделе, с номером его начального сектора и размером. Всего в таблице разделов может находиться четыре таких записи.

# Носители информации

Раздел, запись о котором находится в таблице разделов нулевого сектора, носит названия **первичного (primary)**. Из-за упомянутых ограничений таких разделов на одном диске может быть максимум четыре. Некоторые операционные системы устанавливаются только на первичные тома. При необходимости использования большего числа разделов в таблицу заносится запись о **расширенном (extended)** разделе. Данный тип раздела представляет собой контейнер, в котором создаются **логические (logical)** разделы. Логических томов может быть неограниченное количество, однако в ОС семейства Windows число одновременно подключенных томов ограничено количеством букв латинского алфавита.



# Файловая система

**Файловая система (ФС)** - это часть операционной системы, назначение которой состоит в том, чтобы организовать эффективную работу с данными, хранящимися во *внешней памяти*, и обеспечить пользователю удобный интерфейс при работе с такими данными.

**Файл** — это именованная область внешней памяти, в которую можно записывать и из которой можно считывать данные.

# Файловая система

Файловая система (ФС) — это часть операционной системы, включающая:

- совокупность всех файлов на диске;
- наборы структур данных, используемых для управления файлами, такие, например, как каталоги файлов, дескрипторы файлов, таблицы распределения свободного и занятого пространства на диске;
- комплекс системных программных средств, реализующих различные операции над файлами, такие как создание, уничтожение, чтение, запись, именованение и поиск файлов.

# Файловая система

Основные функции файловой системы

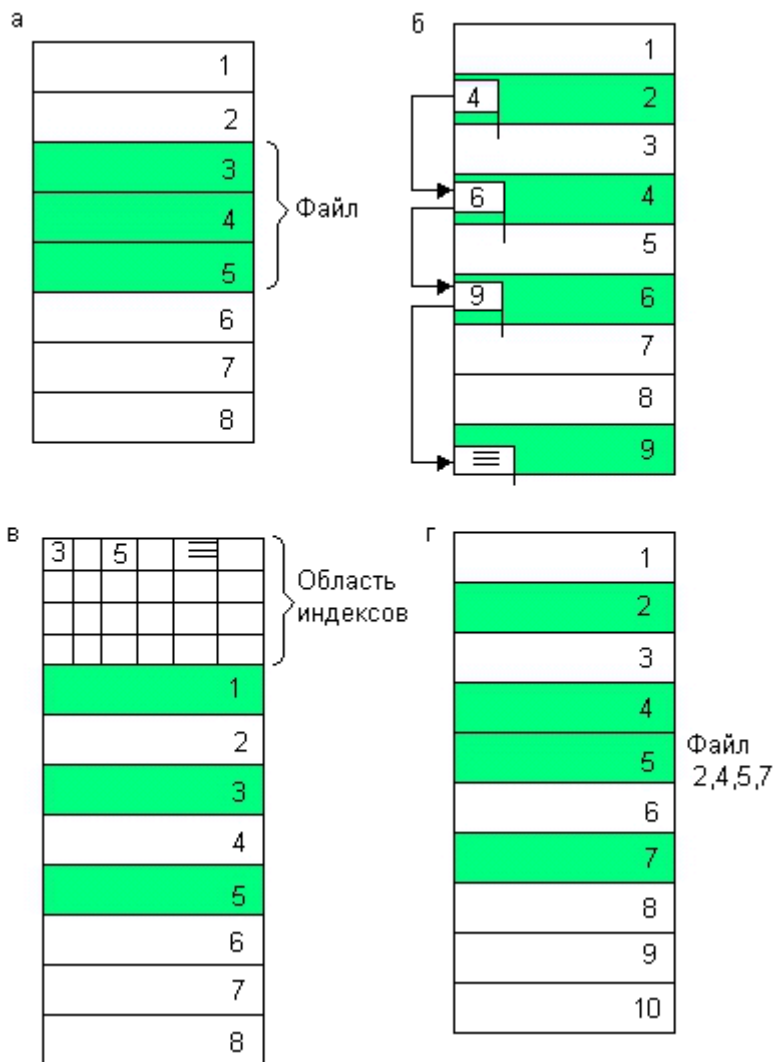
1. Идентификация *файлов*. Связывание имени *файла* с выделенным ему пространством *внешней памяти*.
2. Распределение *внешней памяти* между *файлами*.
3. Обеспечение надежности и отказоустойчивости.
4. Обеспечение защиты от несанкционированного доступа.
5. Обеспечение совместного *доступа к файлам*, так чтобы пользователю не приходилось прилагать специальных усилий по обеспечению синхронизации доступа.
6. Обеспечение высокой производительности.

# Файл

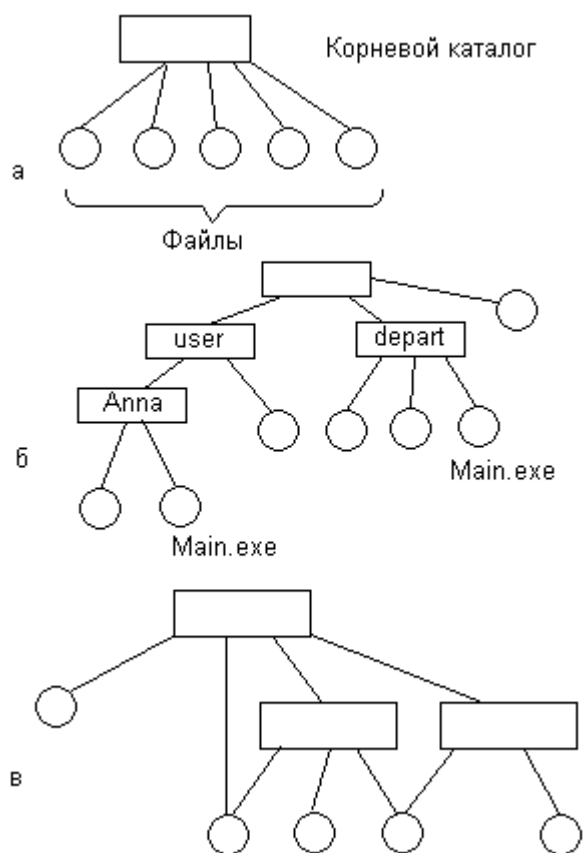
В некоторых ОС (MS-DOS) адреса блоков, содержащих данные *файла*, могут быть организованы в связный список и вынесены в отдельную таблицу в памяти.

В других ОС (Unix) адреса блоков данных *файла* хранятся в отдельном блоке *внешней памяти* (так называемом индексе или индексном узле). Этот прием, называемый ***индексацией***, является наиболее распространенным для приложений, требующих произвольного доступа к записям *файлов*.

# Файл



# Структура файловой системы



Граф, описывающий иерархию каталогов, может быть деревом или сетью. Каталоги образуют дерево, если файлу разрешено входить только в один каталог (б), и сеть — если файл может входить сразу в несколько каталогов (в). Например, в MS-DOS и Windows каталоги образуют древовидную структуру, а в UNIX — сетевую. В древовидной структуре каждый файл является листом. Каталог самого верхнего уровня называется корневым каталогом, или корнем (root).

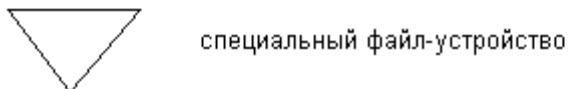
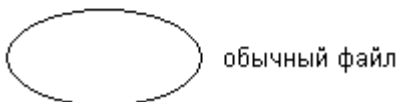
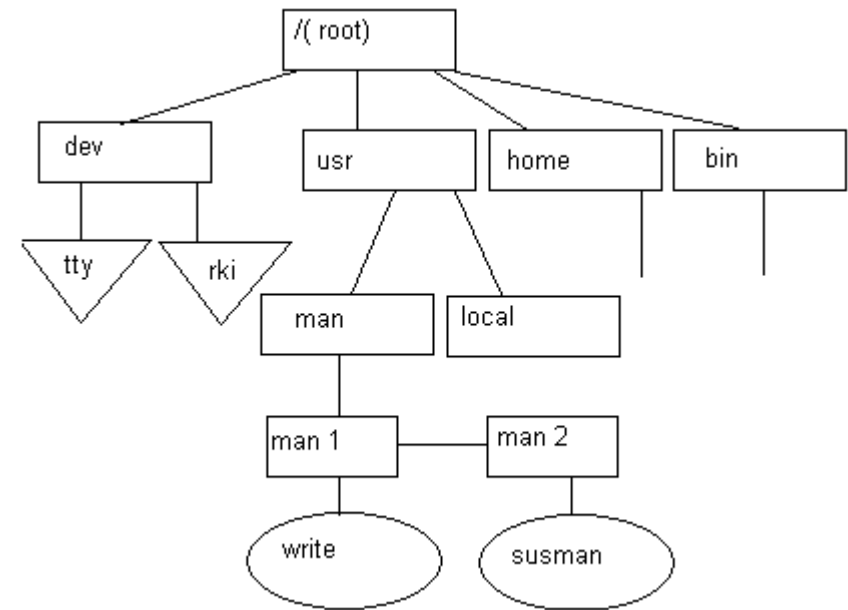
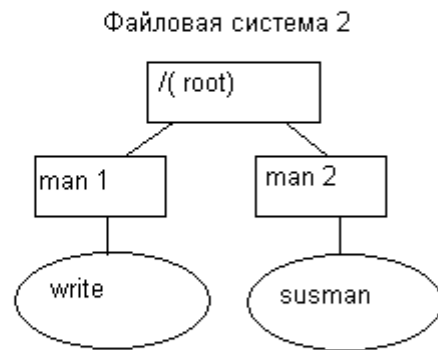
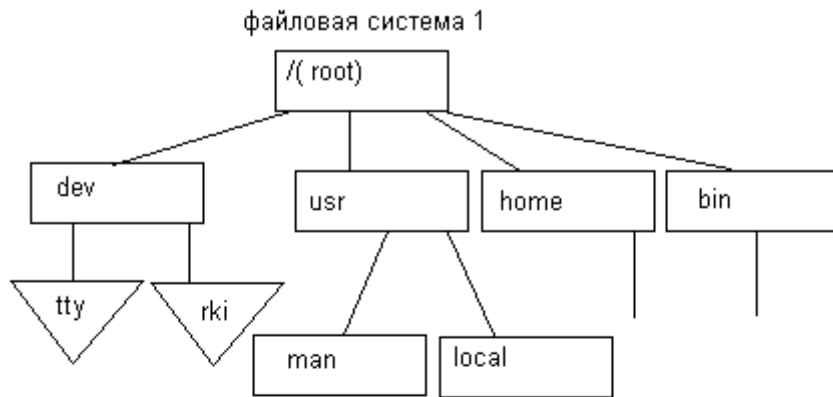
# Имена файлов

В иерархически организованных файловых системах обычно используются три типа имен -файлов: простые, составные и относительные.

8.3	autoexec.bat
14	automatedfile
255	«Мой отчет по лабораторной работе.doc»

Локальное имя	my.ini
Полное имя	/etc/my.ini
	C:\Windows\config.ini

# Монтирование





# Атрибуты файлов

- **тип файла (обычный файл, каталог, специальный файл и т. п.);**
- *владелец файла;*
- *создатель файла;*
- **пароль для доступа к файлу;**
- *информация о разрешенных операциях доступа к файлу;*
- *времена создания, последнего доступа и последнего изменения;*
- **текущий размер файла;**
- **максимальный размер файла;**
- **признак «только для чтения»;**
- **признак «скрытый файл»;**
- **признак «системный файл»;**
- **признак «архивный файл»;**
- **признак «двоичный/символьный»;**
- **признак «временный» (удалить после завершения процесса);**
- *признак блокировки;*
- **длина записи в файле;**
- **указатель на ключевое поле в записи;**
- **длина ключа.**

# Пример для FAT, Unix

8		3			1		4			
Имя файла		Расширение			R	A	H	S	Резервные	
Резервные	Время	Дата		№ первого кластера			Размер			

а

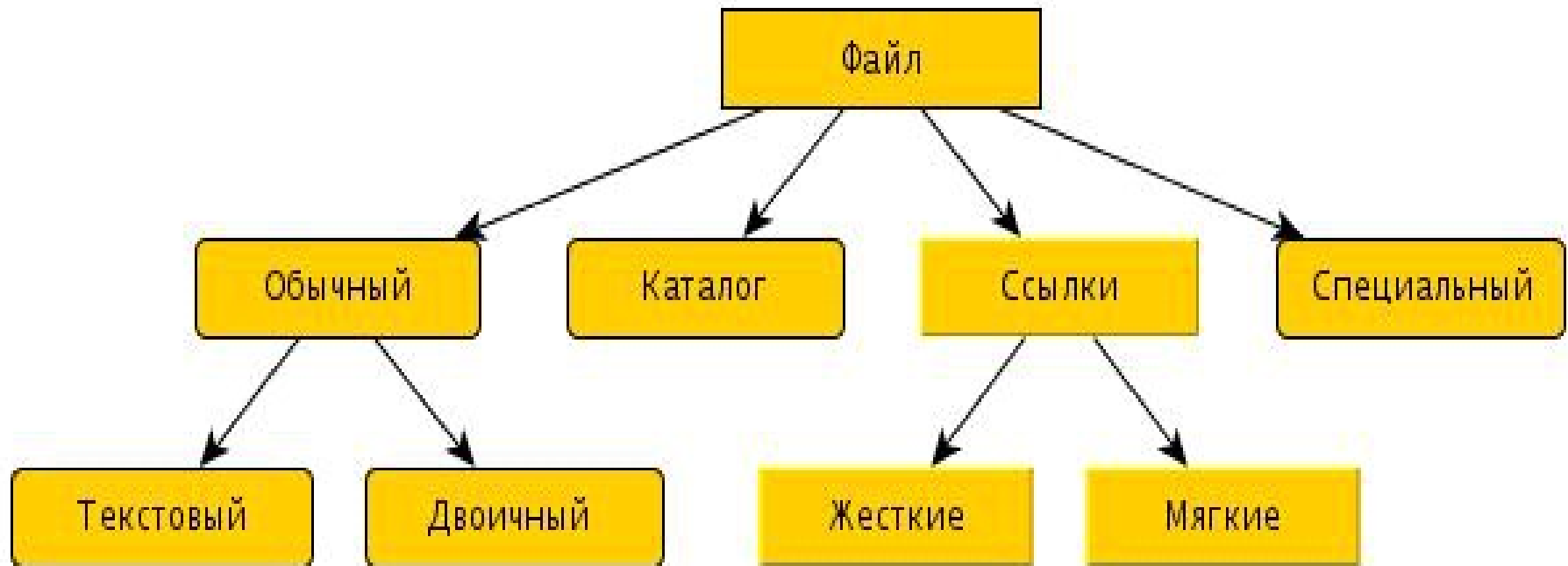
2

14

б

№ индексного дескриптора		Имя файла	
--------------------------	--	-----------	--

# Типы файлов



# Способы доступа

Последовательный

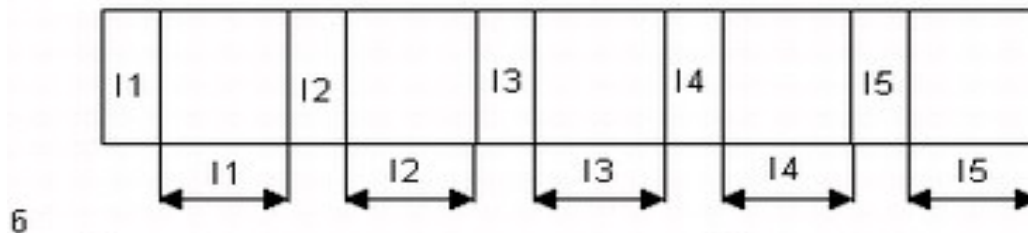
Прямой (random)

*Функция seek*

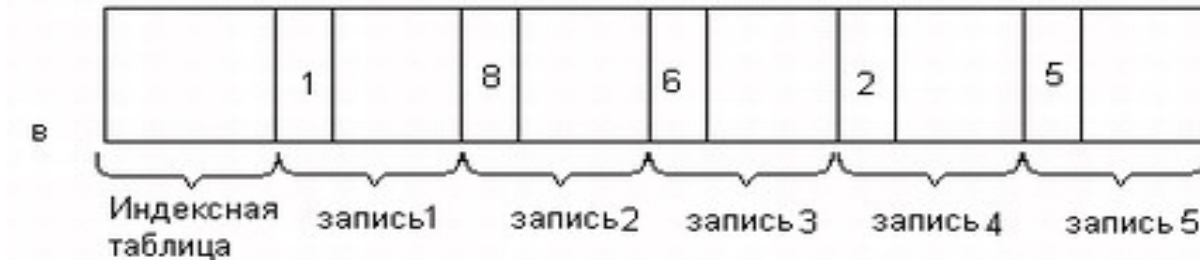
# Логическая структура файлов



Последовательность логических записей фиксированной длины



Последовательность логических записей переменной длины



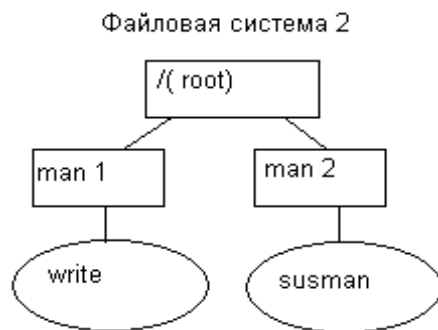
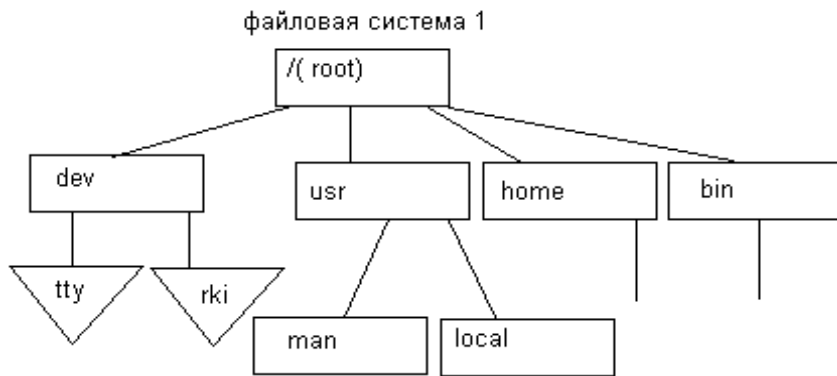
Индексная логическая организация

Индекс	1	2	3	4	5	6	Индекс $\equiv$ ключ
Адрес	21	201	315	661	670	715	


# Операции над файлами

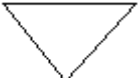
- Создание файла
- Удаление файла
- Открытие файла
- Закрытие файла
- Позиционирование файла
- Чтение из файла
- Запись в файл
- Переименование файла
- Изменение атрибутов
- ...

# Каталоги



 каталог

 обычный файл

 специальный файл-устройство

**Каталог** — специальный файл, содержащий структуру, позволяющую описывать вложенные в него файлы (имена, физическое месторасположение, атрибуты и т. д.)

pathname

path

# Операции над каталогами

- Создать каталог
- Удалить каталог
- Открыть каталог
- Заккрыть каталог
- Поиск
- Получение списка файлов
- Переименовать



# Файловые системы

- Для носителей с произвольным доступом (например, жёсткий диск): FAT32, HPFS, ext2 и др. Поскольку доступ к дискам в разы медленнее, чем доступ к оперативной памяти, для прироста производительности во многих файловых системах применяется асинхронная запись изменений на диск. Для этого применяется либо журналирование, например в ext3, ReiserFS, JFS, NTFS, XFS, либо механизм soft updates и др.
- Для носителей с последовательным доступом (например, магнитные ленты): QIC и др.
- Для оптических носителей — CD и DVD: ISO9660, HFS, UDF и др.
- Сетевые файловые системы: NFS, CIFS, SSHFS, GmailFS и др.
- Для флэш-памяти: YAFFS, ExtremeFFS, exFAT.

# FAT

## **File Allocation Table**

Разработана Биллом Гейтсом и Марком МакДональдом (англ.) в 1976—1977 годах. Использовалась в качестве основной файловой системы в операционных системах семейств DOS и Windows (до версии Windows 2000)

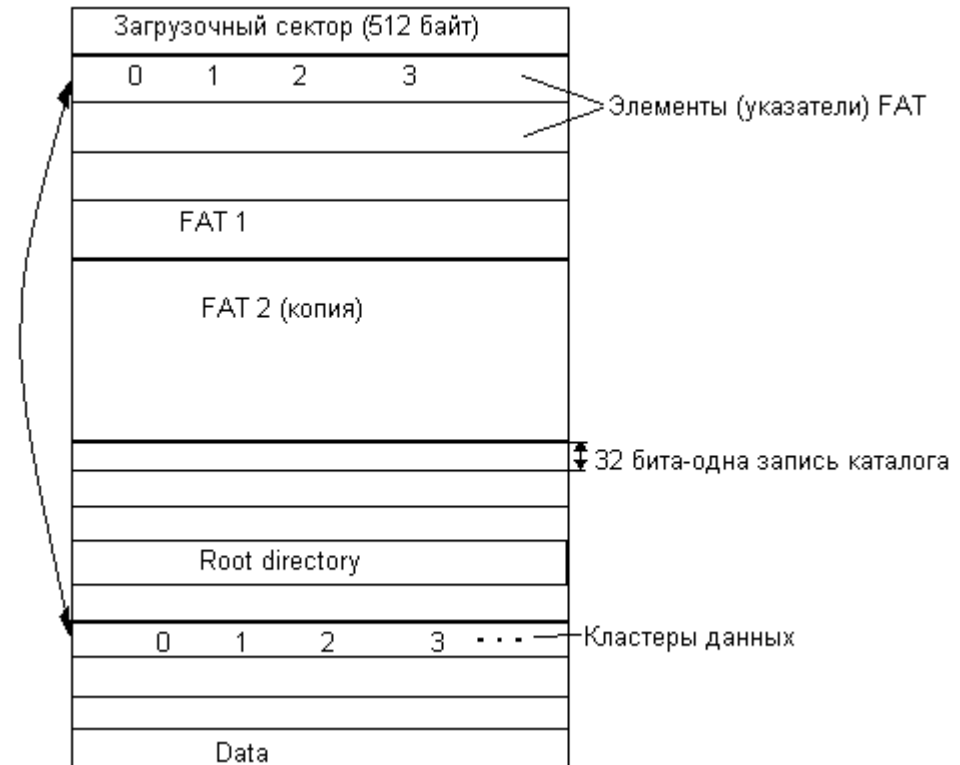
# FAT

Существует три версии FAT — FAT12, FAT16 и FAT32. Они отличаются разрядностью записей в дисковой структуре, т.е. количеством бит, отведённых для хранения номера кластера. FAT12 применяется в основном для дискет, FAT16 — для дисков малого объёма. На основе FAT была разработана новая файловая система exFAT (extended FAT), используемая преимущественно для флеш-накопителей.

VFAT — это расширение FAT, появившееся в Windows 95. В FAT имена файлов имеют формат 8.3 и состоят только из символов кодировки ASCII. В VFAT была добавлена поддержка длинных (до 255 символов) имён файлов (англ. Long File Name, LFN) в кодировке UTF-16LE, при этом LFN хранятся одновременно с именами в формате 8.3, ретроспективно называемыми SFN (англ. Short File Name). LFN нечувствительны к регистру при поиске, однако, в отличие от SFN, которые хранятся в верхнем регистре, LFN сохраняют регистр символов, указанный при создании файла.

# FAT

- PBR, Загрузочный сектор содержит программу начальной загрузки операционной системы. Вид этой программы зависит от типа операционной системы, которая будет загружаться из этого раздела.
- Основная копия FAT содержит информацию о размещении файлов и каталогов на диске.
- Резервная копия FAT.
- Корневой каталог занимает фиксированную область размером в 32 сектора (16 Кбайт), что позволяет хранить 512 записей о файлах и каталогах, так как каждая запись каталога состоит из 32 байт.
- Область данных предназначена для размещения всех файлов и всех каталогов, кроме корневого каталога.



# Таблица FAT

Важная структура тома FAT – это сама таблица FAT, занимающая отдельную логическую область. Она определяет список (цепочку) кластеров, в которых размещаются файлы и папки тома.

Возможны следующие состояния:

- **кластер свободен** – указатель обнулен;
- **кластер занят файлом** и не является последним кластером файла – значение указателя — это номер следующего кластера файла;
- **кластер является последним кластером файла** – указатель содержит метку EOC (End Of Clusterchain), значение которой зависит от версии FAT;
- **кластер поврежден** – указатель содержит специальную метку. Поврежденный кластер не может использоваться файловой системой для хранения данных; соответствующие указатели не затрагиваются при форматировании тома, когда все остальные указатели обнуляются.

# Достоинства и недостатки FAT

- Низкая устойчивость с мягким сбоям;
- "Потерянные кластеры";
- Поддерживается большинством устройств;
- Отсутствуют механизмы разграничения доступа;
- Максимальный размер файла = 32Мб / 2Гб / 4Гб.
- Максимальный размер тома = 32Мб / 2Гб / 2Тб (4Гб/8Тб)

# Имена файлов FAT

Имя и расширение файла могут содержать любую комбинацию букв, цифр или символов с ASCII-кодами свыше 127; специальные символы распределяются на три группы:

**Разрешенные:** ! # \$ % & ( ) - @ ^ \_ ` { } ~ ' "

**Запрещенные:** + , . ; = [ ]

**Служебные:** \* ? < : > / \ | " "

Имя файла не может начинаться или заканчиваться пробелом; ни в каком байте поля имени недопустимы служебные символы ASCII, предшествующие пробелу, т.е. 0x00-0x1F (за исключением 0x05 / 0xE5)

# NTFS

## New Technology File System

NTFS заменила FAT. Впервые представлена в 1993 в Windows 3.1. Разработана на основе HPFS (High Performance File System, которая создавалась IBM совместно с Microsoft).

OS/2 — HPFS

NTFS — Windows NT, Windows XP ...

В других ОС представлена в виде дополнительных драйверов.

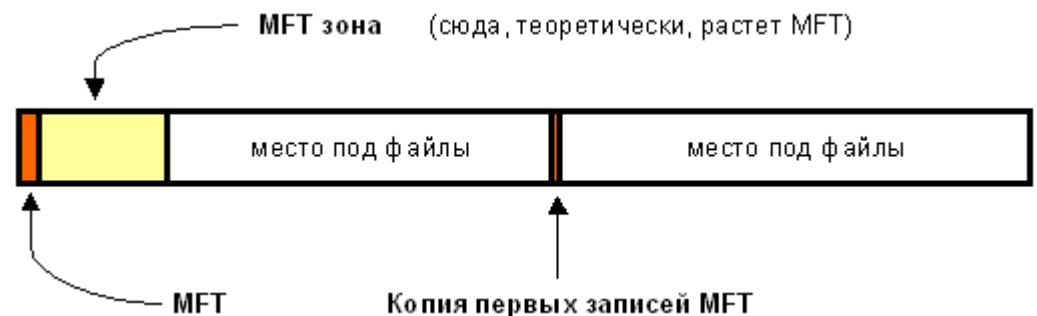
Linux(чтение, чтение/запись), MacOS (чтение, чтение/запись в последних версиях)



# NTFS

Как и любая другая система, NTFS делит все полезное место на кластеры - блоки данных, используемые единовременно. NTFS поддерживает почти любые размеры кластеров - от 512 байт до 64 Кбайт, неким стандартом же считается кластер размером 4 Кбайт.

Диск NTFS условно делится на две части. Первые 12% диска отводятся под так называемую MFT зону - пространство, в которое растет метафайл MFT. Запись каких-либо данных в эту область невозможна. MFT-зона всегда держится пустой - это делается для того, чтобы самый главный, служебный файл (MFT) не фрагментировался при своем росте. Остальные 88% диска представляют собой обычное пространство для хранения файлов.



# NTFS. MFT

Каждый элемент системы представляет собой файл - даже служебная информация. Самый главный файл на NTFS называется MFT (Master File Table) - общая таблица файлов. Именно он размещается в MFT зоне и представляет собой централизованный каталог всех остальных файлов диска, и, как не парадоксально, себя самого. MFT поделен на записи фиксированного размера (обычно 1 Кбайт), и каждая запись соответствует какому либо файлу (в общем смысле этого слова). Первые 16 файлов несут служебный характер и недоступны операционной системе - они называются метафайлами, причем самый первый метафайл - сам MFT. Эти первые 16 элементов MFT - единственная часть диска, имеющая фиксированное положение. Вторая копия первых трех записей, для надежности - хранится ровно посередине диска. Остальной MFT-файл может располагаться, как и любой другой файл, в произвольных местах диска - восстановить его положение можно с помощью его самого, "зацепившись" за самую основу - за первый элемент MFT.

# NTFS. Файл

- Запись в MFT. В этом месте хранится вся информация о файле, за исключением собственно данных. Имя файла, размер, положение на диске отдельных фрагментов, и т.д. Если для информации не хватает одной записи MFT, то используются несколько, причем не обязательно подряд.
- Опциональный элемент - потоки данных файла. Во-первых, файл может не иметь данных - в таком случае на него не расходуется свободное место самого диска. Во-вторых, файл может иметь не очень большой размер. Тогда идет в ход довольно удачное решение: данные файла хранятся прямо в MFT, в оставшемся от основных данных месте в пределах одной записи MFT. Файлы, занимающие сотни байт, обычно не имеют своего "физического" воплощения в основной файловой области - все данные такого файла хранятся в одном месте - в MFT.

# NTFS. Данные файла

Довольно интересно обстоит дело и с данными файла. Каждый файл на NTFS, в общем-то, имеет несколько абстрактное строение - у него нет как таковых данных, а есть потоки (streams). Один из потоков и носит привычный нам смысл - данные файла. Но большинство атрибутов файла - тоже потоки! Таким образом, получается, что базовая сущность у файла только одна - номер в MFT, а всё остальное опционально. Данная абстракция может использоваться для создания довольно удобных вещей - например, файлу можно "прилепить" еще один поток, записав в него любые данные - например, информацию об авторе и содержании файла. Эти дополнительные потоки не видны стандартными средствами: наблюдаемый размер файла - это лишь размер основного потока. Можно, к примеру, иметь файл нулевой длины, при стирании которого освободится 1 Гбайт свободного места.

# NTFS. Каталог

Каталог на NTFS представляет собой специфический файл, хранящий ссылки на другие файлы и каталоги, создавая иерархическое строение данных на диске. Файл каталога поделен на блоки, каждый из которых содержит имя файла, базовые атрибуты и ссылку на элемент MFT, который уже предоставляет полную информацию об элементе каталога. Внутренняя структура каталога представляет собой бинарное дерево.

# Достоинства и недостатки NTFS

- Высокая устойчивость с мягким сбоям;
- Поддерживается не всеми ОС, устройствами;
- Максимальный размер файла =  $2^{44}$ (практика),  $2^{64}$ (теория);
- Максимальный размер тома = 9.4 зетабайт;
- Максимальный размер диска =  $2^{64}$  (16 эксабайт, ~16тыс.Тб);
- Поддерживает жёсткие ссылки и символьные ссылки(?);
- Есть средства разграничения доступа / шифрования;
- Журналирование;
- Сжатие.

# Ext3

**Third Extended File System** (третья версия расширенной файловой системы), сокращённо `ext3` или `ext3fs` — журналируемая файловая система, используемая в операционных системах на ядре Linux, является файловой системой по умолчанию во многих дистрибутивах.

# Ext3

Граф, описывающий иерархию каталогов файловой системы ext, представляет собой сеть. Причиной такой организации является то, что один файл может входить сразу в несколько каталогов.

Все типы файлов имеют символьные имена. В иерархически организованных файловых системах обычно используются три типа имен: простые, составные и относительные. Не является исключением и ext. Ограничения на простое имя состоят в том что, его длина не должна превышать 255 символов, а также в имени не должны присутствовать символ NULL и слеш.



# Ext3

Атрибутами файловой системы ext2 являются:

- тип и права доступа к файлу,
- владелец, группа доступа,
- информация о разрешённых операциях,
- время создания, дата последнего доступа, дата последнего изменения и время последнего удаления,
- текущий размер файла,
- спецификация файла:
  1. обычный файл,
  2. каталог,
  3. файл байт-ориентированного устройства,
  4. файл блочно-ориентированного устройства,
  5. именованный канал,
  6. символическая ссылка,
- число занимаемых блоков,
- ...

# Ext3

Как и в любой файловой системе UNIX, в составе ext можно выделить следующие составляющие:

- блоки и группы блоков;
- индексный дескриптор;
- суперблок;

Всё пространство раздела диска разбивается на блоки фиксированного размера, кратные размеру сектора: 1024, 2048, 4096 или 8192 байт. Размер блока указывается при создании файловой системы на разделе диска. Меньший размер блока позволяет сэкономить место на жёстком диске, но также ограничивает максимальный размер файловой системы. Все блоки имеют порядковые номера. С целью уменьшения фрагментации и количества перемещений головок жёсткого диска при чтении больших массивов данных блоки объединяются в группы блоков.

# Ext3

Базовым понятием файловой системы является индексный дескриптор, или inode (англ. information node). Это специальная структура, которая содержит информацию об атрибутах и физическом расположении файла. Индексные дескрипторы объединены в таблицу, которая содержится в начале каждой группы блоков.

Суперблок (Superblock)
Описани группы блоков (Group Descriptors)
Битовая карта блоков (Block Bitmap)
Битовая карта индексных дескрипторов (Inode Bitmap)
Таблица индексных дескрипторов (Inode Table)
Данные (Data)

# Достоинства и недостатки Ext3

- Журналируема;
- Максимальный размер файла = 16Гб / 2Тб;
- Максимальный размер тома = 2Тб / 32Тб;
- Основное использование в Linux.

# Ext4

Основной особенностью стало увеличение максимального объёма одного раздела диска до 1 эксбибайта (2<sup>60</sup> байт) при размере блока 4Kb, и увеличение размера одного файла до 16 тебибайт. Кроме того, в ext4 представлен механизм пространственной (extent) записи файлов (новая информация добавляется в конец заранее выделенной по соседству области файла), уменьшающий фрагментацию и повышающий производительность.

# Ext4. Особенности

- Снижение фрагментации (несколько технологий, обеспечивающих эту особенность);
- Повышение производительности;
- Дефрагментация без размонтирования;
- Максимальный размер файла = 16 Тб;
- Максимальный размер диска = 1 эксабайт.

# ISO9660

ISO 9660 — выпущенный Международной организацией по стандартизации стандарт, описывающий файловую систему для дисков CD-ROM. Целью стандарта является обеспечить совместимость носителей под разными операционными системами, такими как Unix, Mac OS, Windows.

Расширение стандарта, называемое Joliet, добавляет поддержку длинных имён файлов и не-ASCII символов в именах.

DVD также может использовать ISO 9660, но файловая система UDF является более подходящей для них, так как имеет поддержку больших носителей и лучше подходит для современных операционных систем.

# ISO9660

Ограничения на имена файлов и директорий

- Можно использовать только латинские буквы в верхнем регистре, цифры и символ подчеркивания.
- Имена файлов не должны включать пробелов.
- Имена файлов не должны начинаться или заканчиваться точкой.
- Имена файлов не должны иметь более одной точки.
- Имена каталогов не должны содержать точки.

Ограничение на глубину вложенности директорий = 8

Ограничение на размер файла = 2Гб



# ISO9660. Juliet

Joliet — расширение файловой системы ISO 9660. Спецификация была разработана фирмой Microsoft и поддерживается всеми версиями ОС Microsoft Windows со времён Windows 95 и Windows NT 4.0. Главной целью было ослабление ограничений на имя файла, накладываемых ISO 9660.

Joliet достигает этой цели введением дополнительного набора имён файлов (до 64 символов Unicode длиной), в кодировке UCS-2. Эти имена хранятся в специальном дополнительном заголовке (Supplementary Volume Descriptor, SVD), который безусловно игнорируется ISO 9660-совместимыми программами, обеспечивая этим обратную совместимость.

# ISO9660. Rock Ridge

Rock Ridge Interchange Protocol (RRIP, IEEE P1282) — расширение файловой системы ISO 9660, разработанное для хранения файловых атрибутов, используемых в операционных системах POSIX. Расширения Rock Ridge записываются поверх файловой системы ISO 9660 так, что оптический диск с Rock Ridge может быть прочитан программным обеспечением, рассчитанным на работу с ISO 9660.

- длинные имена файлов (до 255 символов);
- структуру каталогов произвольной вложенности;
- поддерживаются специальные файлы.

# UDF

UDF (англ. Universal Disk Format, универсальный дисковый формат) — спецификация формата файловой системы, независимой от операционной системы для хранения файлов на оптических носителях. UDF является реализацией стандарта ISO/IEC 13346 (известного также как ECMA-167). Формат UDF призван заменить ISO 9660.

UDF позволяет дозаписывать файлы на CD-R или CD-RW дисках. Нет ограничения в 2 и 4 ГБ на размер файла.

Существуют версии: 1.02, 1.50, 2.0, 2.01, 2.50, 2.60, ...