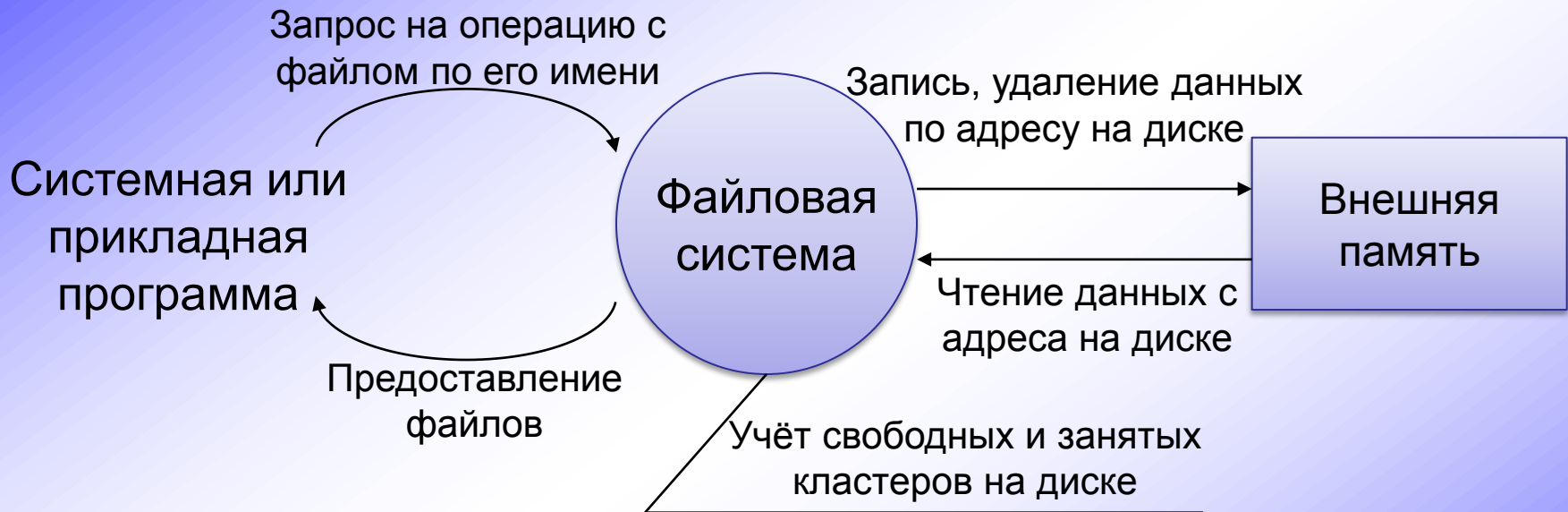


# Файловые системы

(часть 1)

# Файловая система



- Работа с данными на диске осуществляется на основе адресов секторов.
- Работа программы с данными осуществляется на основе имени файла.

# Цель использования файловой системы

Экранирование физической организации долговременного хранилища данных, создание более простой логической модели этого хранилища и предоставление набора удобных в использовании команд для манипулирования файлами.

# Состав файловой системы

- Совокупность всех файлов на диске.
- Наборы структур данных, используемых для управления файлами (каталоги, дескрипторы файлов).
- Комплекс системных программных средств, реализующих различные операции над файлами (создание, уничтожение, поиск и др.).

# Функции файловой системы

- Отображение логической модели файловой системы на физическую организацию хранилища данных.
- Обеспечение устойчивости файловой системы к сбоям питания, ошибкам аппаратных и программных средств.
- Программный интерфейс для приложений.
- Обеспечение совместного доступа к файлу нескольких процессов.
- Защита файлов от несанкционированного доступа.

# Примеры файловых систем

Для Windows:

- FAT (File Allocation Table);
- NTFS (New Technology File System).

Для UNIX:

- UFS (Unix File System, для BSD);
- S5 (для System V).

Для Linux:

- ext (Extended File System);
- Reiser.

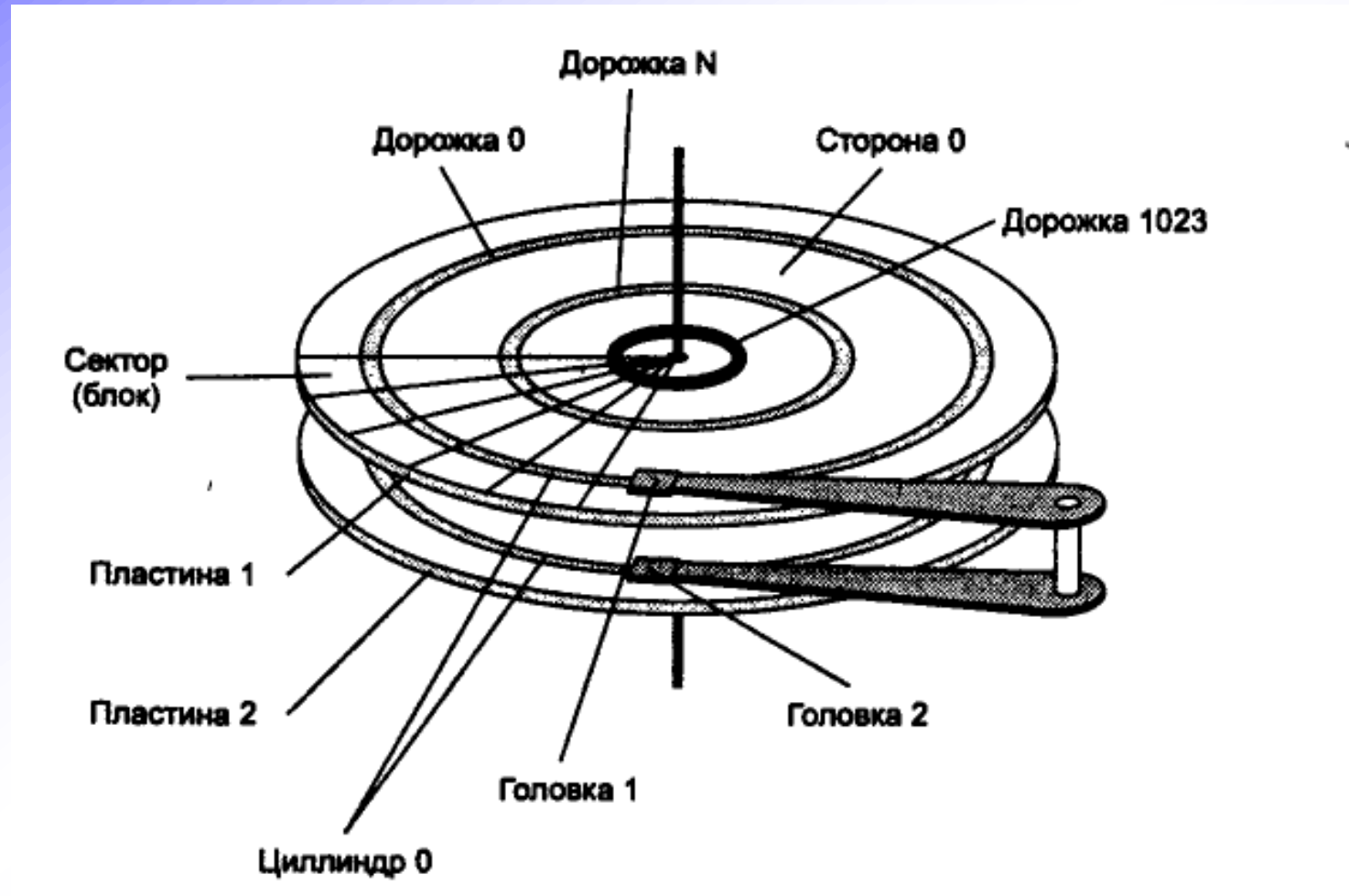
# Физическая организация файловых систем

# Низкоуровневое форматирование диска

- Дорожки – концентрические кольца для хранения данных, размеченные на диске; состоят из одинакового числа секторов.
- Сектор – наименьшая адресуемая единица дискового устройства для обмена данными. Размер сектора фиксирован.
- При низкоуровневом форматировании создаются дорожки и сектора, на диск записывается информация для определения границ секторов.



# Низкоуровневая организация физического диска



# Логические диски

- Логический диск (раздел) – непрерывная часть физического диска, которую операционная система представляет пользователю как логическое устройство.
- На каждом логическом диске может создаваться только одна файловая система.

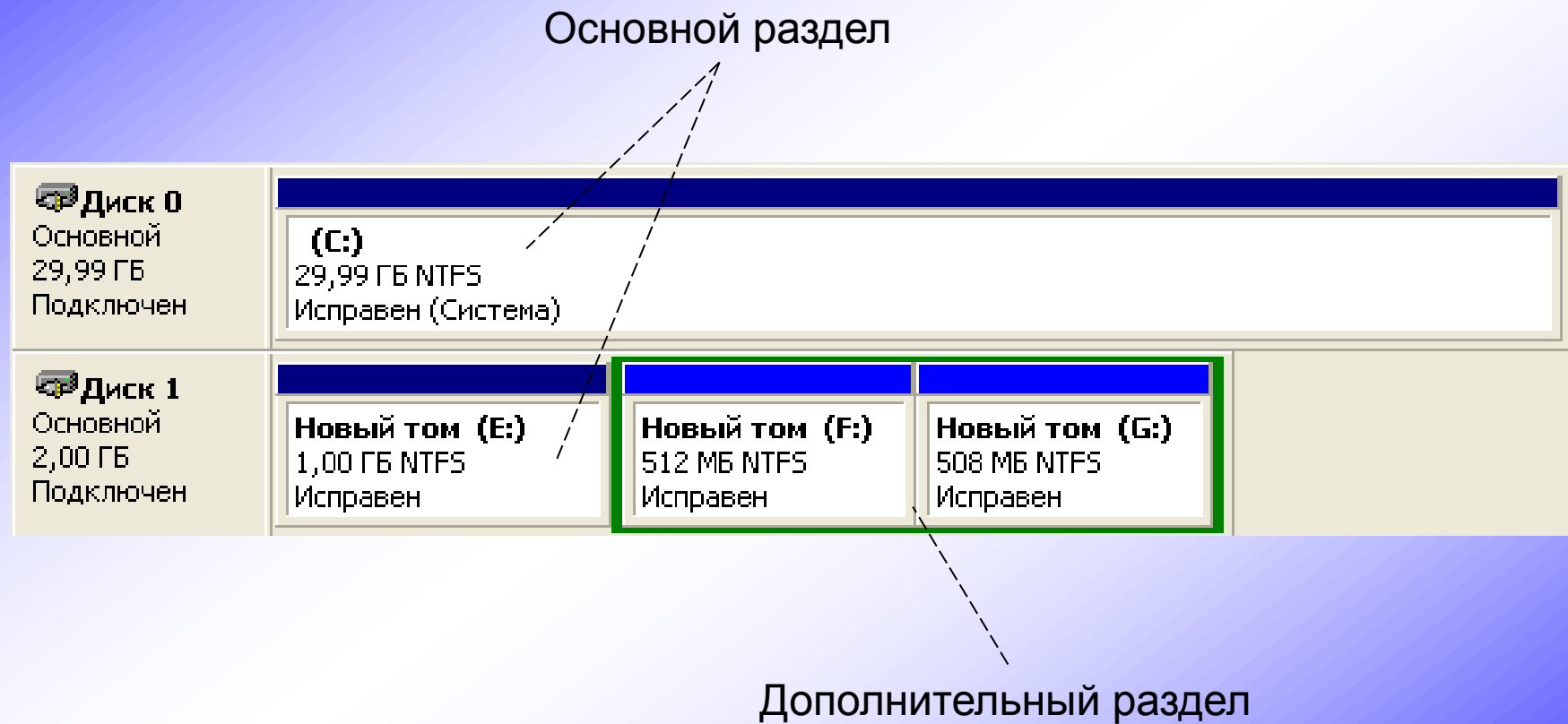
# Логические диски

Возможные варианты организации логических дисков:

- «один физический диск – один логический диск»;
- «один физический диск – несколько логических дисков»;
- «несколько физических дисков – один логический диск».

Один из логических дисков помечается как загружаемый и с него считывается загрузчик операционной системы.

# Логические диски



В 0-й сектор диска записывается главная загрузочная запись (MBR).

# Главная загрузочная запись

Структура MBR:

- код загрузчика (446 байт);
- таблица разделов (4 записи по 16 байт);
- сигнатура MBR (2 байта).

Структура описания раздела:

- признак активности раздела (1 байт);
- начало раздела (3 байта) – номер головки (1-й байт), номер сектора (биты 0-5 2-го байта), номер дорожки (биты 6,7 2-го байта и 3-й байт);
- код типа раздела (1 байт);
- конец раздела (3 байта) – номер головки (1-й байт), номер сектора (биты 0-5 2-го байта), номер дорожки (биты 6,7 2-го байта и 3-й байт);
- смещение первого сектора (4 байта);
- количество секторов раздела (4 байта).

# Основные и дополнительные разделы

- MBR вмещает 4 записи о разделах, значит максимальное количество разделов – 4 (4 основных раздела или 3 основных и 1 дополнительный).
- Операционная система может загружаться только из основного раздела.
- Если необходимо создать более 4 разделов, то нужно использовать дополнительный раздел. Дополнительный раздел может содержать несколько логических дисков.

# Высокоуровневое форматирование диска

- Высокоуровневое форматирование позволяет создать логический диск и файловую систему на нём.
- При форматировании пространство логического диска, предназначенное для хранения данных, разбивается на кластеры и определяется размер кластера.
- Кластер (блок) – минимальная единица хранения данных, используемая в файловой системе.

# Размещение файлов в кластерах

- Размер файла – 17 КБ.
- Размер кластера – 4 КБ.



Кластеры

- Размер файла – 17 КБ.
- Размер кластера – 16 КБ.



Кластеры



# Особенности кластеров большого размера

Достоинства:

- уменьшается размер массива, хранящего информацию о кластерах;
- уменьшается фрагментация больших файлов;
- ускоряется доступ к файлу, так как сокращается длина цепочек фрагментов дискового пространства, выделенных для него.

Недостаток:

- увеличивается свободное место в последнем кластере каждого файла.

# Максимальное количество кластеров и размер разделов

Максимальное количество кластеров:

- FAT12 – 4096.
- FAT16 – 65536.
- FAT32 – 268 435 446.

Максимальный размер раздела:

- FAT12 – 32 МБ (размер кластера – 8 КБ).
- FAT16 – 2 ГБ (размер кластера – 32 КБ).
- FAT32 – 8 ТБ (размер кластера – 32 КБ).

# Максимальный размер раздела в ext

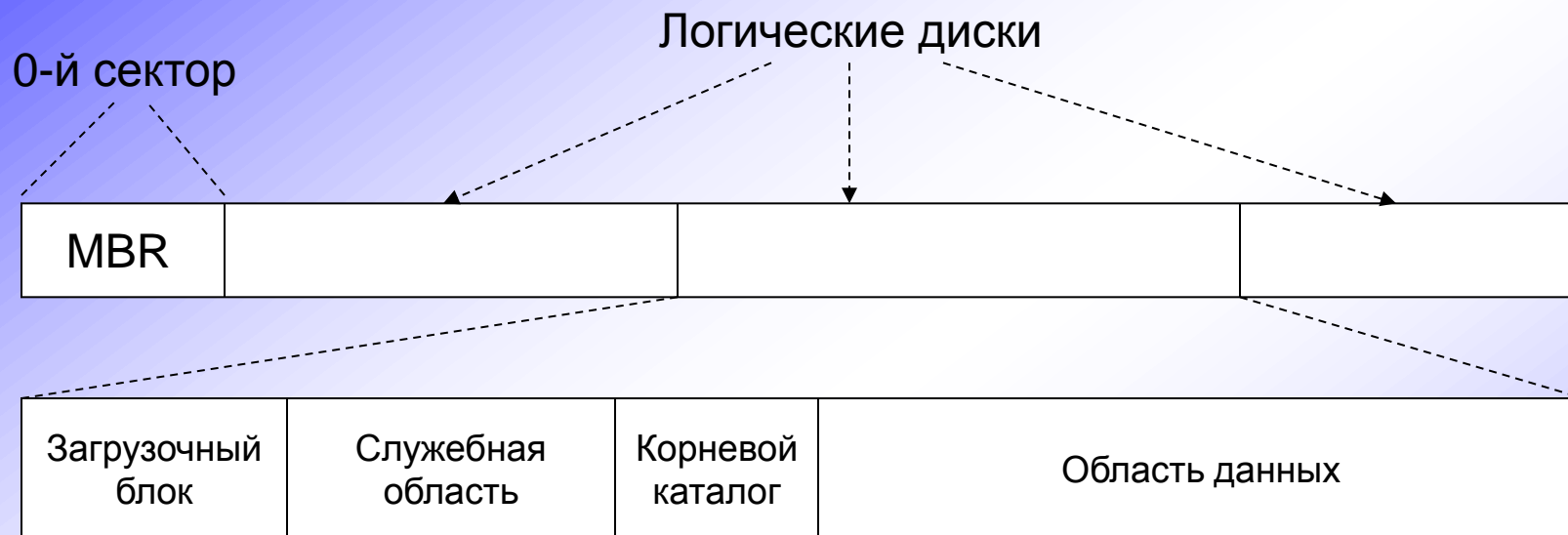
- Ext – 2 Гб.
- ext3 – 4 ТБ.
- ext4 – 48-битные номера блоков позволяют адресовать до 1 ЭБ (при размере блока 4КБ).

# Информация, записываемая на диск при форматировании

При форматировании на диск записывается:

- загрузчик операционной системы;
- информация о границах областей, отведённых под файлы и каталоги;
- информация о повреждённых областях;
- информация о доступном и неиспользуемом пространстве.

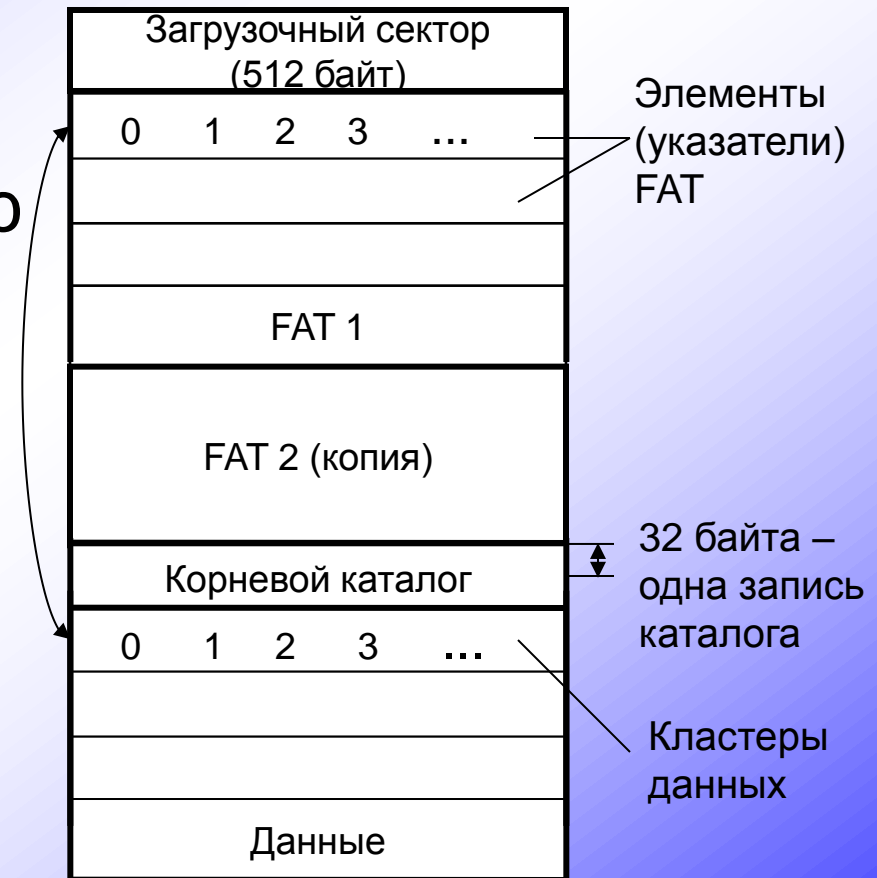
# Структура файловой системы



- Служебная область содержит общую информацию о файловой системе, информацию о свободных кластерах, информацию о размещении файлов в кластерах.
- Записи каталогов (о хранящихся в них файлах) находятся в области данных.

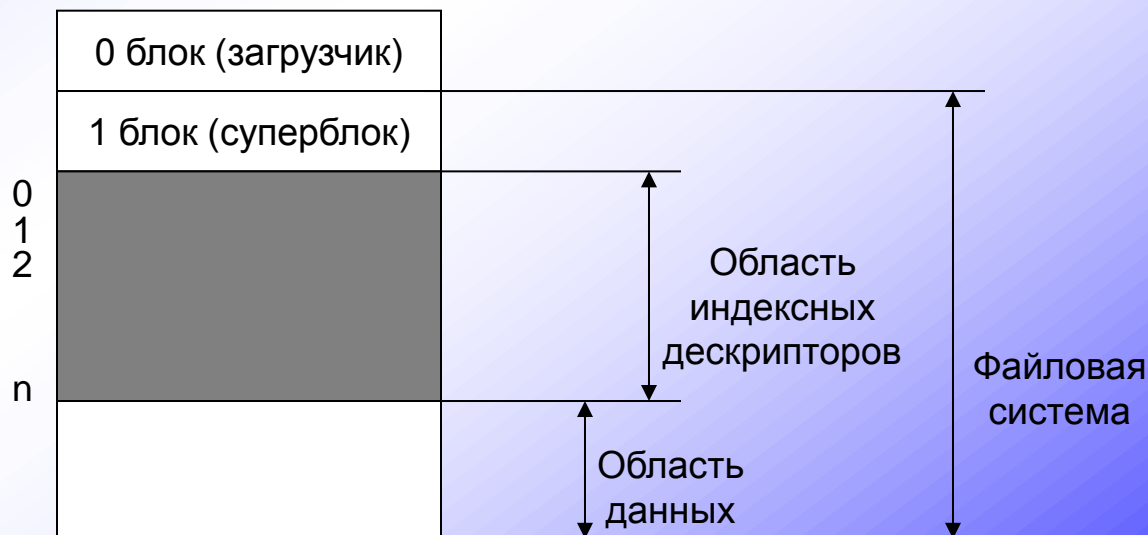
# Физическая организация FAT

- Загрузочный сектор.
- Основная копия FAT (содержит информацию о размещении файлов и каталогов).
- Резервная копия FAT.
- Корневой каталог (в FAT32 может находиться в любой части области данных).
- Область данных.



# Физическая организация S5

- Загрузочный блок.
- Суперблок (размер файловой системы, число индексных дескрипторов, размер области индексных дескрипторов и т.д.).
- Область индексных дескрипторов файлов (расположены в порядке своих номеров).
- Область данных.



# Физическая организация UFS

- Повторяющиеся последовательности областей созданы для ускорения доступа, т.к. блок группы цилиндров описывает количество индексных дескрипторов и блоков данных, расположенных на данной группе цилиндров диска.
- Области «суперблок» содержат одинаковые данные.





# Физическая организация NTFS

- Главная таблица файлов (MFT) – содержит как минимум одну запись для каждого файла и запись для себя (размер записи фиксирован).
- На кластеры делится весь раздел, а не только область данных.



# Логическая организация файловых систем

# Файл

Файл – неструктурированная последовательность байтов, в которую можно записывать и из которой можно считывать информацию.

Файл – логический объект, позволяющий по имени обращаться к информации.

# Задачи, решаемые за счёт использования файлов

- Возможность простого доступа пользователей и приложений к информации по имени.
- Наличие символического имени позволяет получать доступ к информации независимо от адресов кластеров, в которых располагается файл.
- Возможность определения прав доступа пользователя к файлу.

# Типы файлов

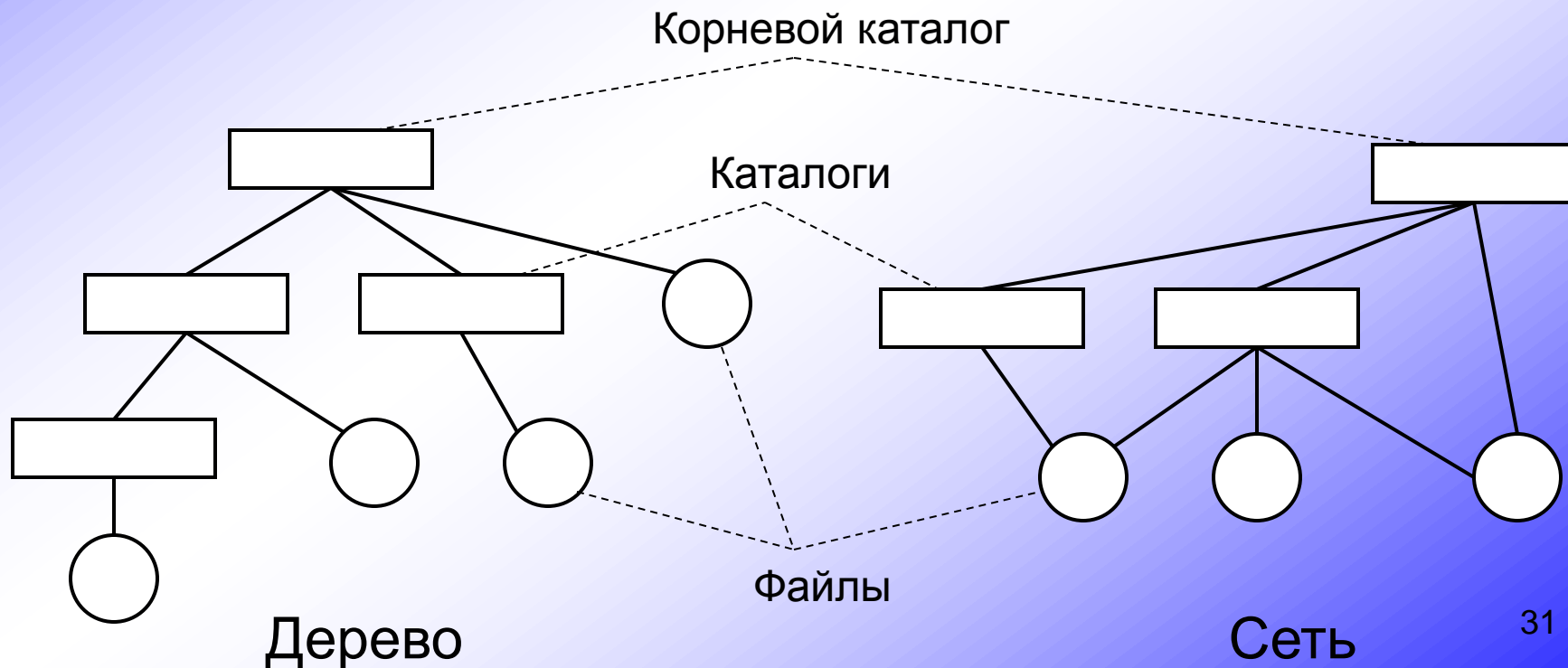
- Файлы – содержат информацию произвольного характера и имеют произвольную структуру; операционная система должна распознавать как минимум один тип файлов – собственные исполняемые файлы.
- Специальные файлы – фиктивные файлы, ассоциированные с устройствами ввода-вывода.

# Типы файлов

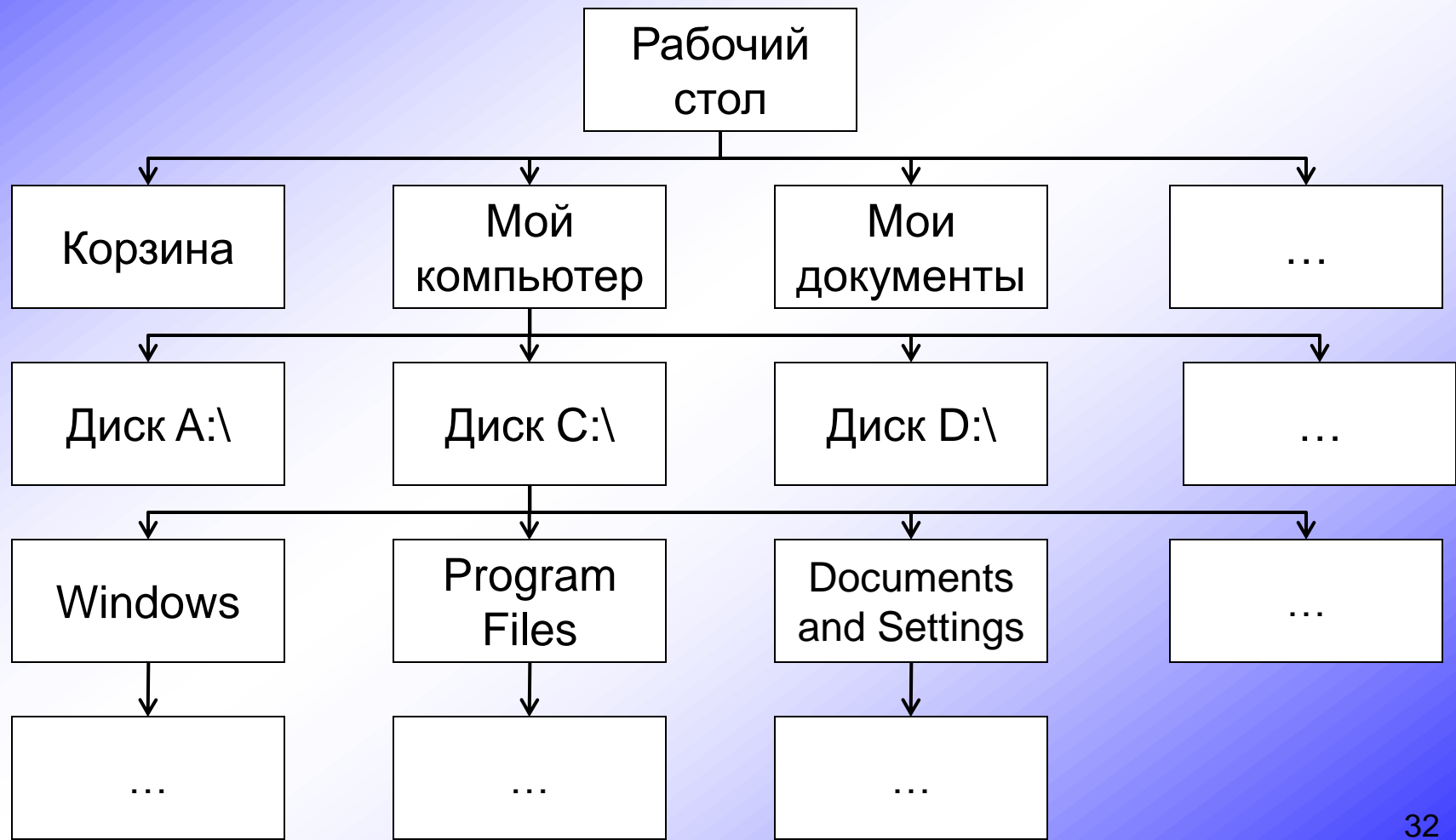
- Каталоги – содержат системную справочную информацию о наборе файлов, сгруппированных пользователем по какому-либо признаку; каталоги могут содержать файлы любых типов, включая каталоги.
- Другие типы файлов – символьные связи, именованные конвейеры, отображаемые в память файлы.

# Иерархическая структура файловой системы

- Дерево – файл может входить только в один каталог.
  - Сеть – файл может входить в несколько каталогов.
- Корневой каталог – каталог верхнего уровня.

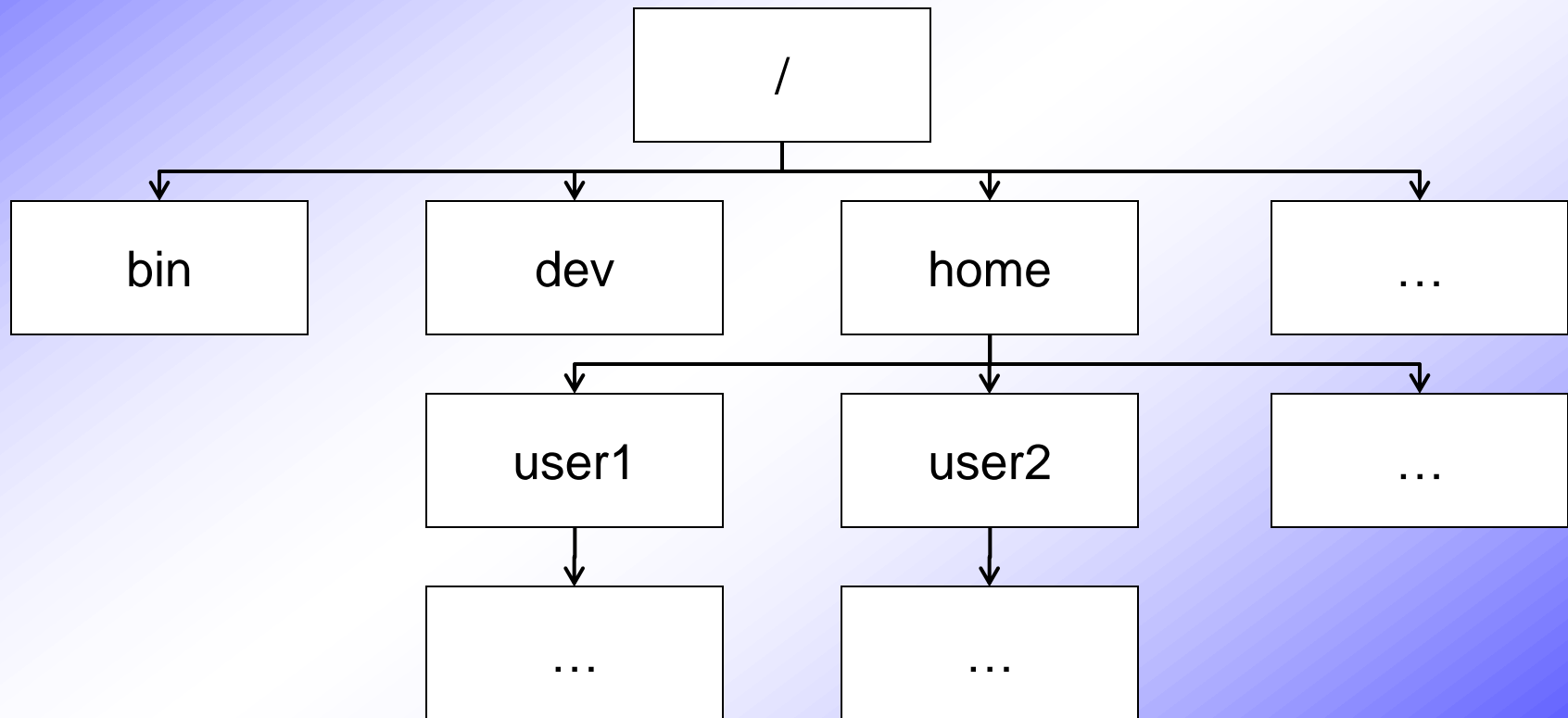


# Иерархическая структура файловой системы в Windows





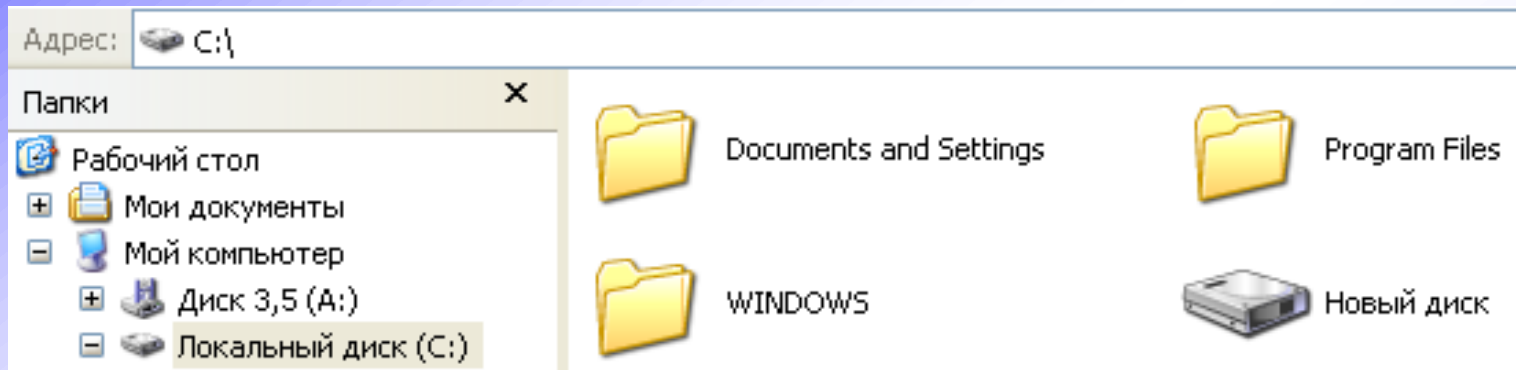
# Иерархическая структура файловой системы в UNIX



# Монтирование логических ДИСКОВ

- Монтирование – встраивание логического диска в иерархическую структуру файлов операционной системы.
- В качестве точки монтирования может выступать любой пустой каталог существующей файловой системы, становящийся корневым для файловой системы монтируемого диска.

# Монтирование логических ДИСКОВ



Возможные значения имени тома вместе с текущими точками подключения:

```
\\?\Volume{af6bd68d-72af-11dd-bcf7-806d6172696f}\  
C:\  
  
\\?\Volume{af6bd68a-72af-11dd-bcf7-806d6172696f}\  
A:\  
  
\\?\Volume{ce27d58e-7d8b-11e1-b6b6-806d6172696f}\  
D:\  
  
\\?\Volume{ce27d592-7d8b-11e1-b6b6-000c29604956}\  
C:\Новый диск\
```

# Физическая организация файлов

# Атрибуты файлов

Атрибуты – информация, описывающая свойства файла.

Примеры атрибутов:

- тип файла;
- владелец файла;
- создатель файла;
- пароль для доступа к файлу;
- информация о разрешённых операциях доступа к файлу;
- время создания, последнего доступа и изменения;
- текущий размер файла;
- признаки «скрытый», «только для чтения», «системный», «архивный», «временный», «двоичный/символьный» и т.д.

# Способы хранения атрибутов

- Хранение в каталогах (область данных) – (а). Структура записи о файле в каталоге: простое имя файла; заданные атрибуты файла.
- Хранение в специальных таблицах (служебная область диска) – (б). Структура записи о файле в каталоге: простое имя файла; указатель на индексный дескриптор (inode) файла, в котором хранятся атрибуты.

A	Атрибуты
B	Атрибуты
C	Атрибуты
D	Атрибуты

а

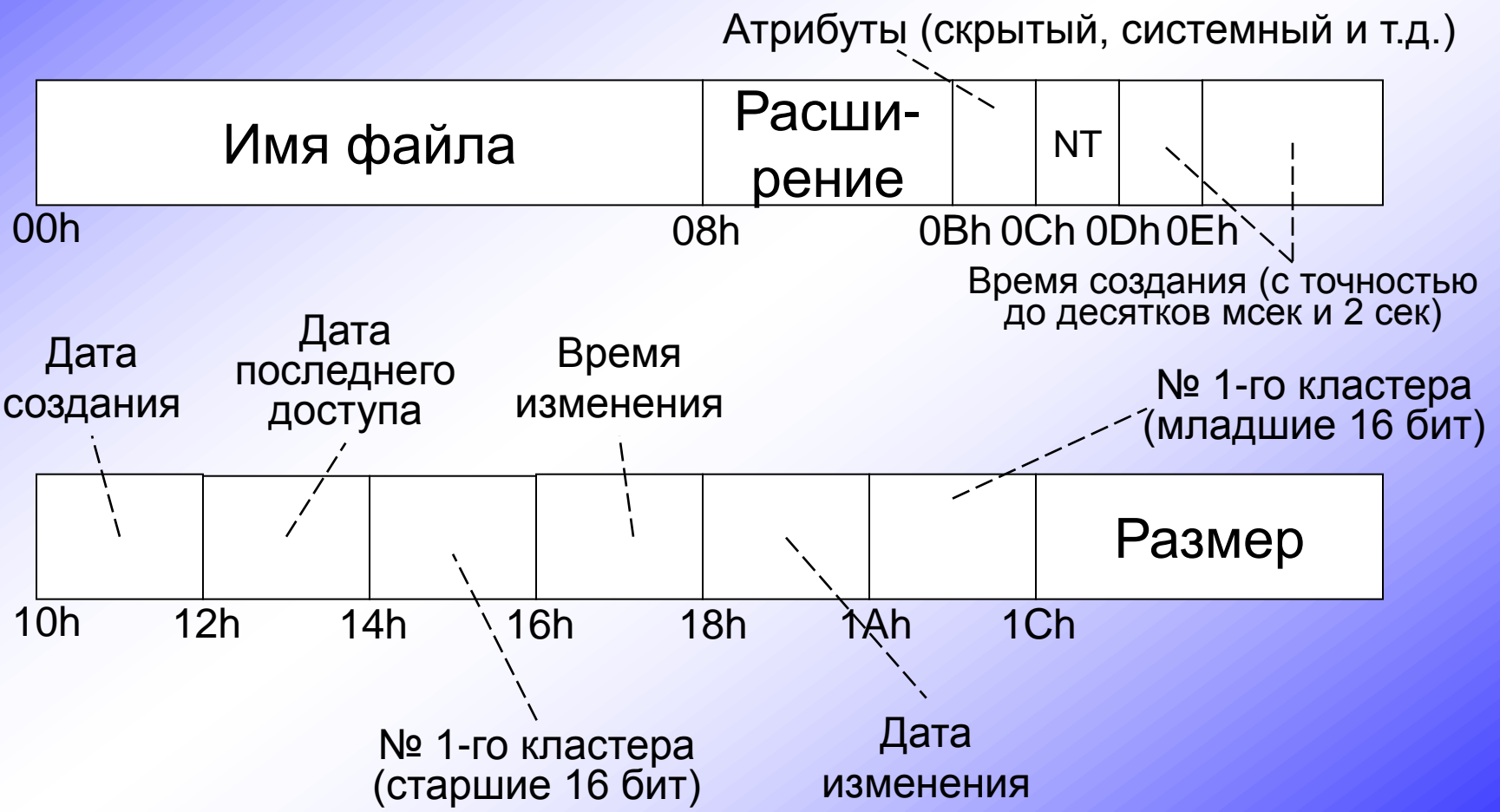
A	
B	
C	
D	

б

Индексные дескрипторы



# Запись каталога при хранении атрибутов в каталогах (FAT32)



# Атрибуты файла в FAT

- Первые 2 бита из байта обнуляются.

000001 – только чтение.

000010 – скрытый.

000100 – системный.

100000 – архивный.

010000 – каталог.

001000 – файл нулевого размера в корневом каталоге – метка тома.

001111 – не файл, а содержит часть длинного имени другого файла.



# Запись каталога при наличии индексных дескрипторов

- Символьное имя файла.
- Номер индексного дескриптора.

Доступ к файлу осуществляется путём последовательного просмотра всей цепочки каталогов, входящих в полное имя файла, и их индексных дескрипторов.

Имя	№ индексного дескриптора
Prog 1	23
firelights	126
doc_23.txt	51
gjazing.txt	17
lambda_good	34

# Индексный дескриптор в S5

Размер дескриптора – 64 байта, содержит:

- идентификатор владельца файла;
- тип файла (файл, каталог, специальный файл, символьная ссылка);
- права доступа к файлу;
- временные характеристики (создание, модификация, обращение);
- число ссылок на индексный дескриптор;
- адресная информация;
- размер файла в байтах.

# Формат поля, определяющего тип файла в UNIX-системах

- 4 бита определяют тип файла: 1000 – обычный файл; 0100 – каталог; 0110 – файл блочного устройства; 0010 – файл символьного устройства; 1010 – доменное гнездо (socket); 0001 – именованный канал (pipe); 1100 – символическая ссылка (link).
- 5-й бит – если равен 1, то исполняемый файл выполняется от имени владельца.
- 6-й бит – если равен 1, то исполняемый файл выполняется от имени группы владельца.
- 7-й бит – если равен 1, то удалить файл может только владелец файла, иначе любой имеющий доступ к файлу.

# Физическая организация файла

Физическая организация файла – способ размещения файлов на диске.

Критерии эффективности:

- скорость доступа к данным;
- объём адресной информации файла;
- степень фрагментированности дискового пространства;
- максимально возможный размер файла.

# Способы физической организации файла

- Непрерывное размещение.
- Связный список кластеров.
- Связный список индексов.
- Перечень номеров кластеров.

# Непрерывное размещение

- Файлу предоставляется последовательность кластеров, образующих непрерывный участок дисковой памяти.
- Скорость доступа – высокая.
- Объём адресной информации – минимальный (номер 1-го кластера и размер файла).
- Фрагментация свободного пространства – высокая.
- Возможность увеличения размера файла приводит к неопределённости при выборе выделяемого размера памяти.

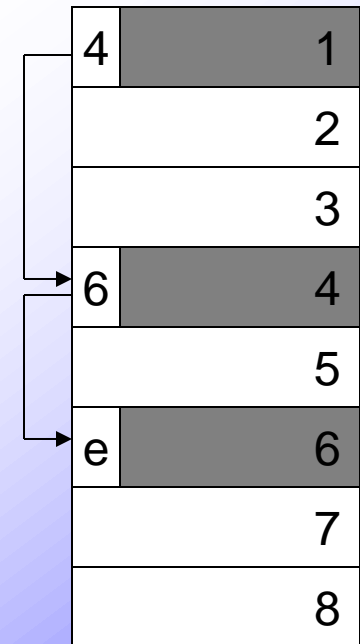


# Фрагментация файлов

- При непрерывном способе размещения файлов скорость считывания файлов высокая, т.к. данные расположены в смежных кластерах.
- Использование других способов приводит к фрагментации файлов и замедлению их считывания.
- Для увеличения скорости считывания файла применяется дефрагментация – копирование специальным сервисом файловой системы составных частей файла в смежные кластеры.

# Связный список кластеров

- В начале каждого кластера есть ссылка на следующий кластер.
- Скорость доступа – низкая из-за необходимости последовательного считывания всех кластеров от первого до заданного.
- Объём адресной информации – минимальный (номер 1-го кластера).
- Фрагментация свободного пространства отсутствует.
- Размер данных в кластере не кратен 2 из-за наличия ссылки на следующий кластер.





# СВЯЗНЫЙ СПИСОК ИНДЕКСОВ

- Файлу выделяется память в виде связанного списка кластеров. Номер первого кластера хранится в записи о файле в каталоге. С каждым кластером связан индекс, равный 0, если кластер свободен, или номеру следующего кластера, если он занят. Индексы хранятся в таблице.
- Отсутствует необходимость последовательного считывания кластеров, достаточно считать значения индексов в таблице.
- Объём адресной информации – минимальный (номер 1-го кластера).
- Фрагментация свободного пространства отсутствует.
- Размер данных в кластере кратен 2.

4			6	
е				
				1
				2
				3
				4
				5
				6
				7
				8

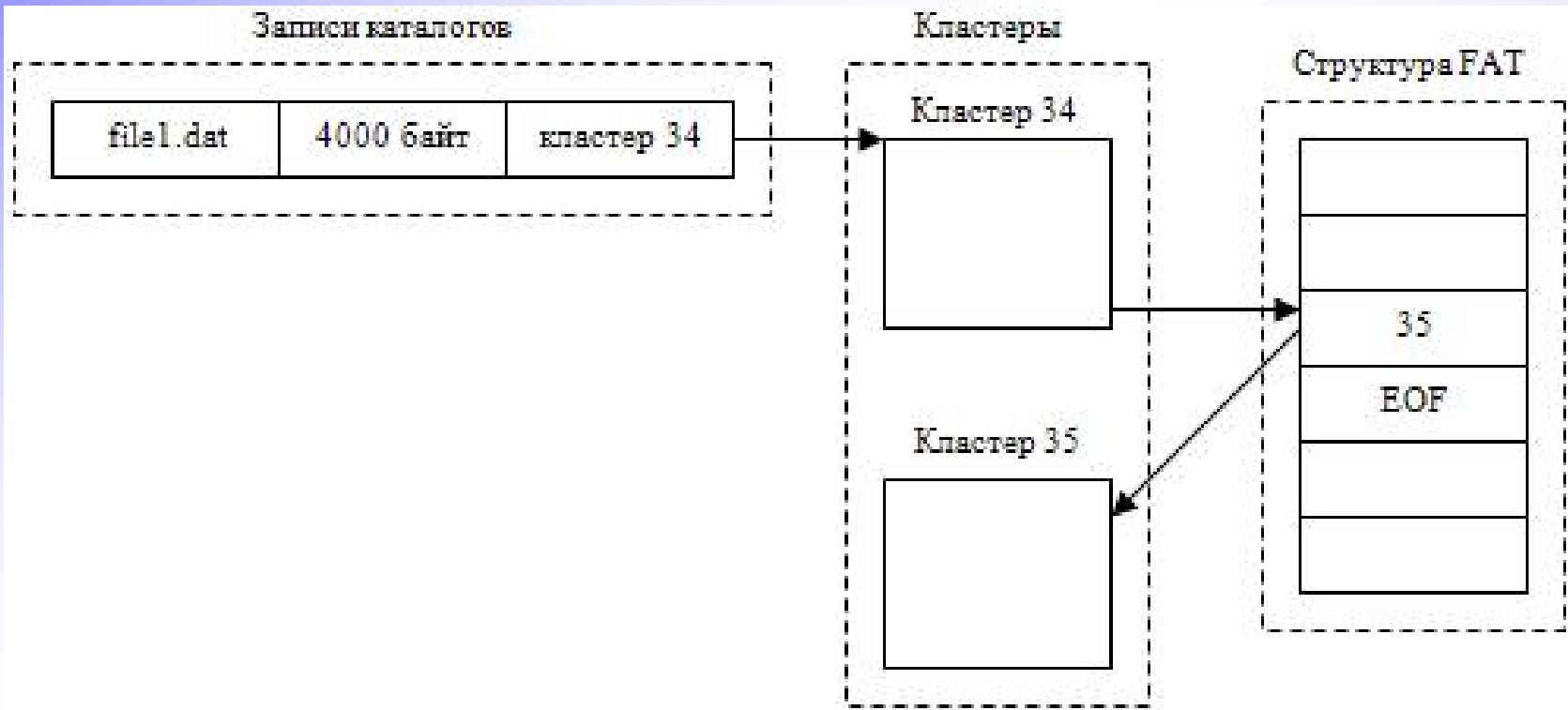
} Область индексов

# Таблица FAT

Состоит из массива индексных указателей, имеющих взаимно однозначное соответствие с кластерами. Значения, которые может принимать индексный указатель:

- кластер свободен;
- кластер используется файлом и не является последним (содержит номер следующего кластера файла);
- последний кластер файла;
- дефектный кластер;
- резервный кластер.

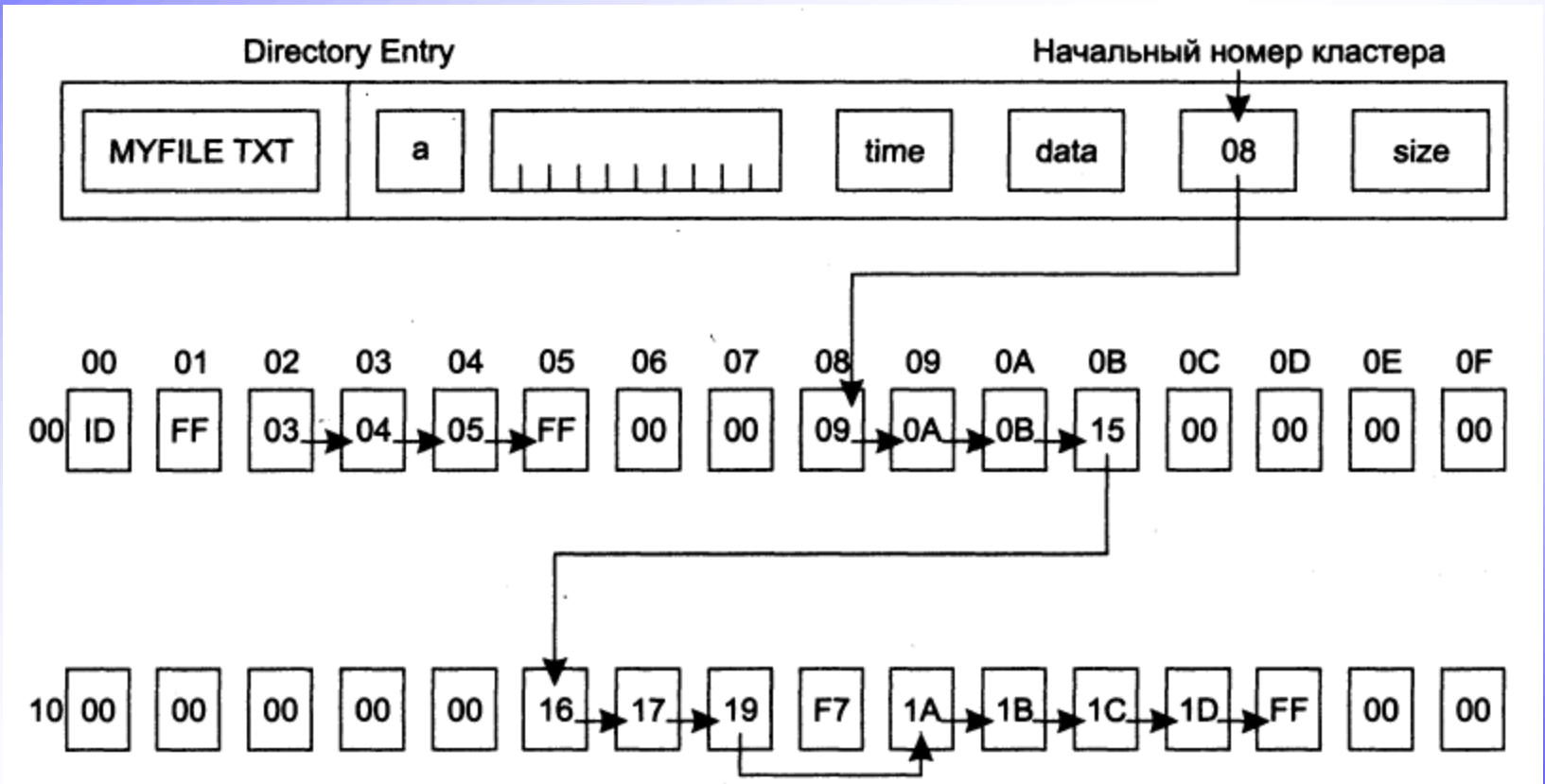
# Принцип считывания файла в FAT



# Принцип размещения файла

- По порядку просматривается таблица и ищется первый индексный указатель на свободный кластер.
- Номер этого указателя фиксируется как номер первого кластера файла в записи каталога.
- Записывается файл; если файл помещается в один кластер, то в индексный указатель заносится значение «последний кластер», а если нет – происходит поиск следующего указателя на свободный кластер и номер найденного кластера записывается в индексный указатель предыдущего.

# Размещение файлов в индексной таблице FAT



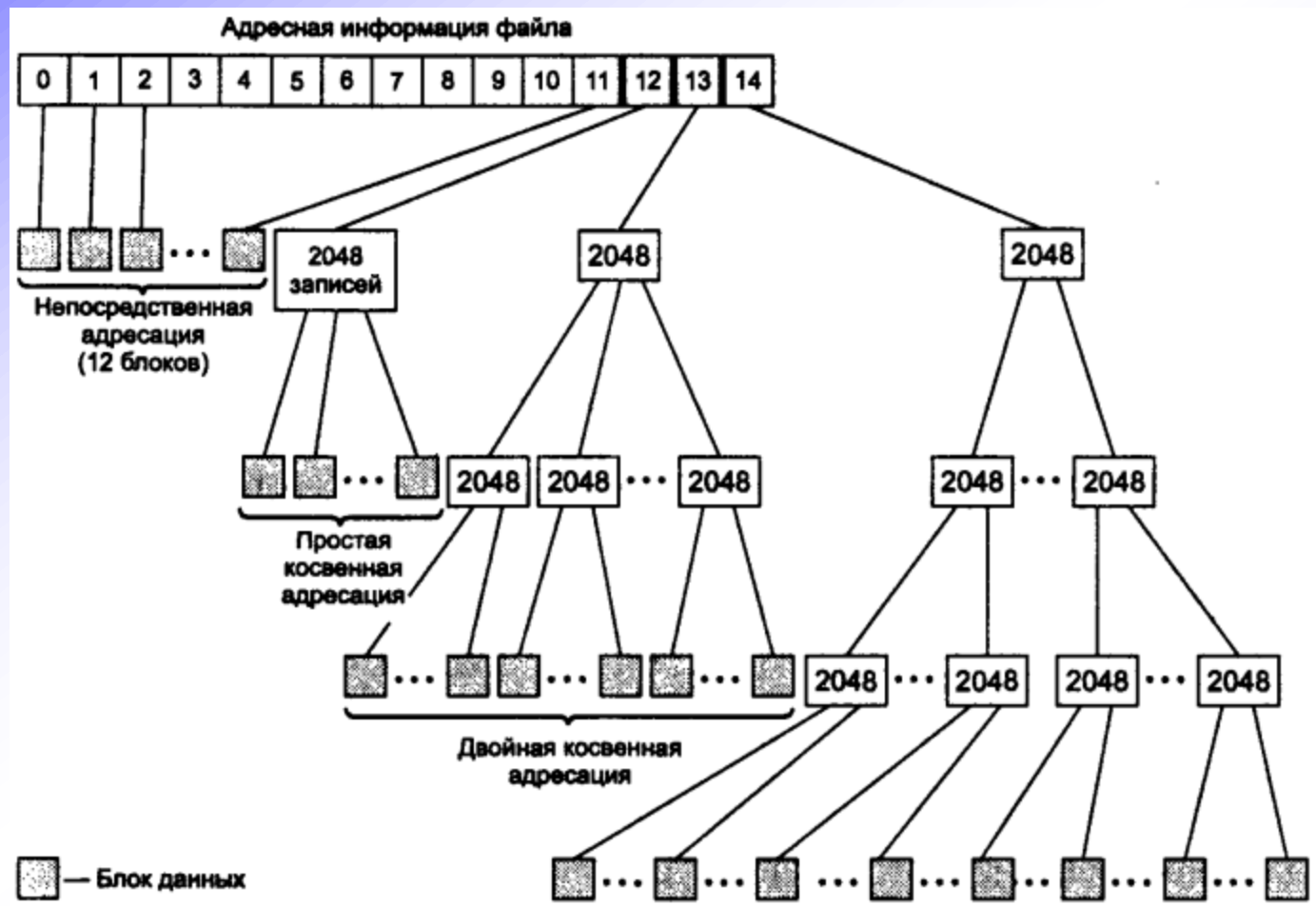
# Удаление файла в FAT

- В первый байт записи каталога (область данных) заносится значение, соответствующее свободной записи. При этом сами данные не удаляются.
- Во все индексные указатели таблицы заносится признак «кластер свободен».

# Перечень номеров кластеров

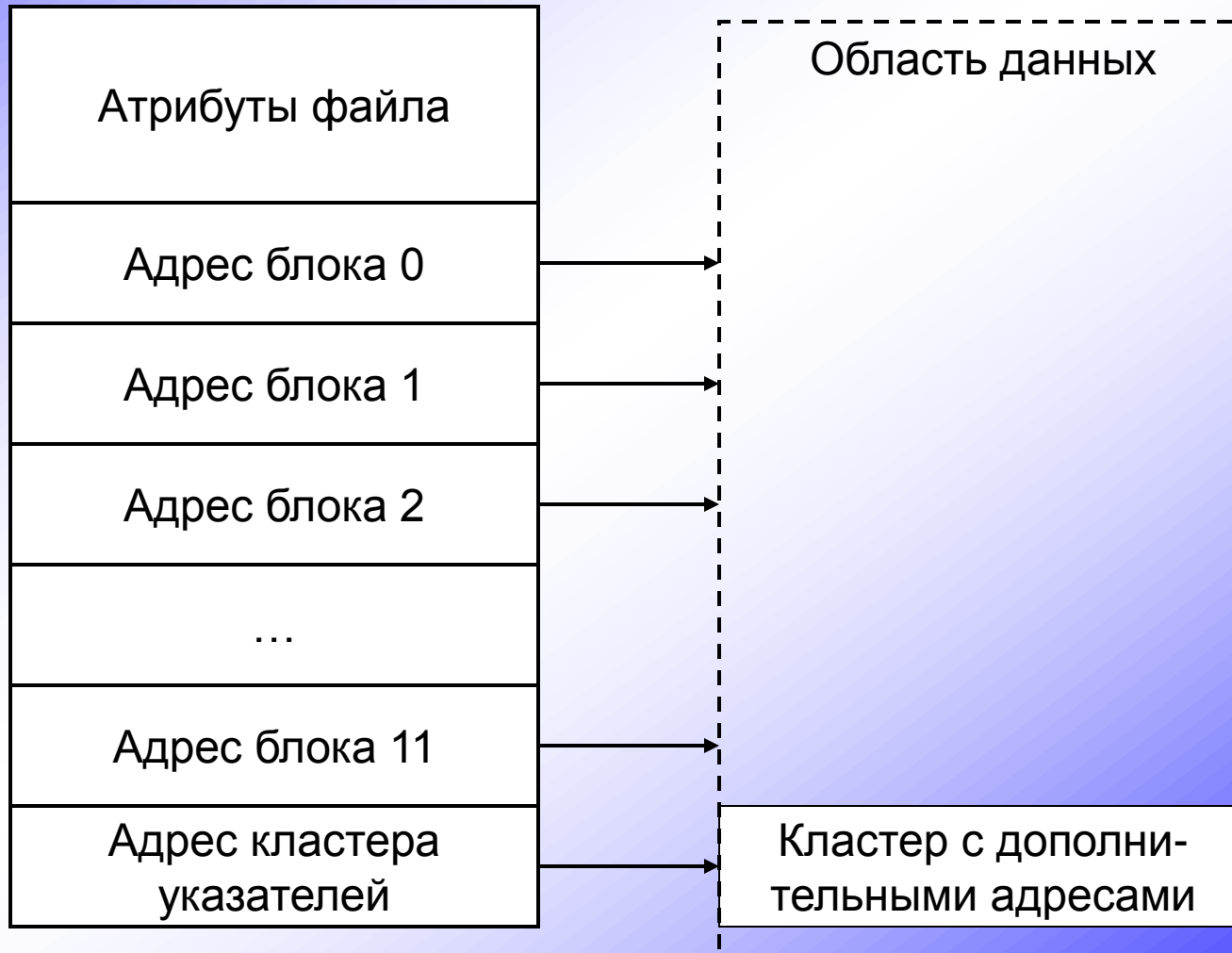
- Для хранения адреса файла выделено 15 полей, равных 4 байтам.
- 12 полей для непосредственной адресации, когда каждое поле – номер кластера.
- 13-е поле для косвенной адресации, когда поле – номер кластера, в котором указаны следующие кластеры.
- 14-е поле для двойной косвенной адресации.
- 15-е поле для тройной косвенной адресации.

# Перечень номеров кластеров





# Структура индексного дескриптора при косвенной адресации



# Перечень номеров кластеров

- Скорость доступа – высокая, так как осуществляется прямая адресация без просмотра цепочки указателей.
- Объем адресной информации – высокий (зависит от размера файла).
- Фрагментация свободного пространства отсутствует.
- При тройной косвенной адресации и размере кластера, равном 8 КБ, могут поддерживаться файлы размером 70 ТБ, хранящихся в 8 млрд кластеров. Объем адресной информации – 0,05% от объема адресуемых данных.

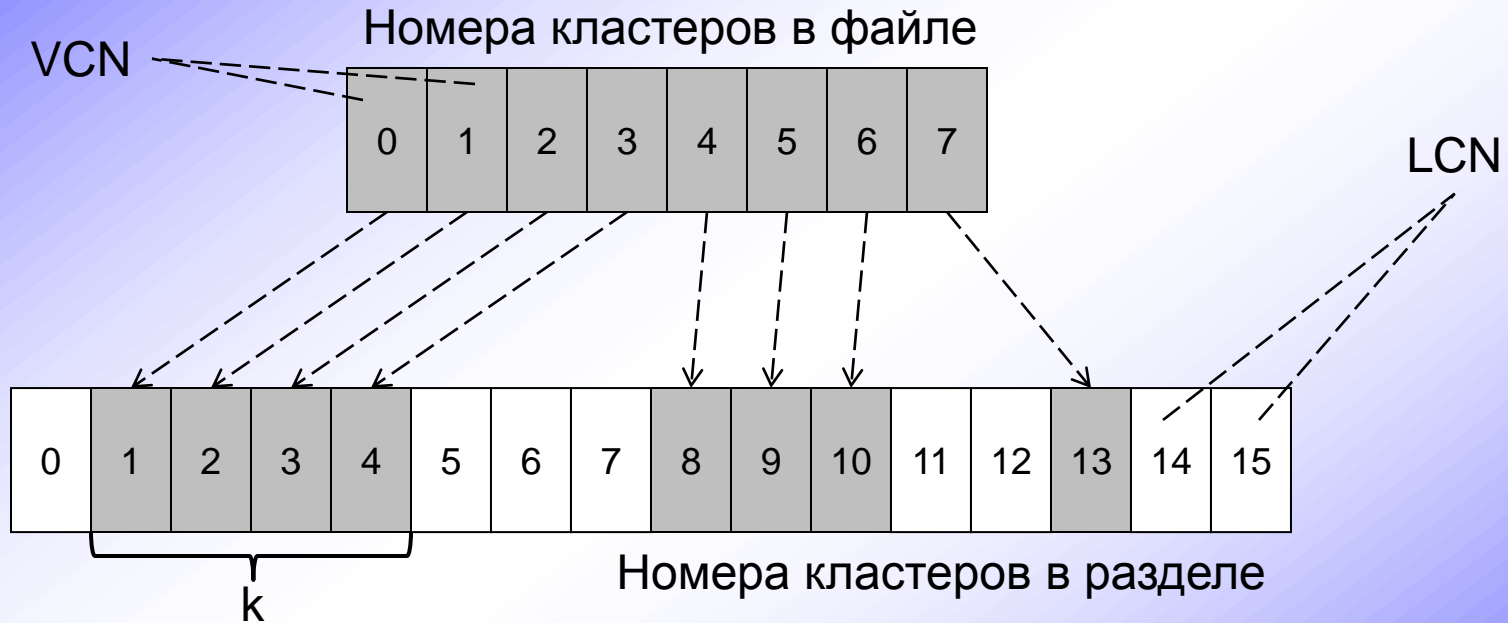
# Использование экстенентов при перечне номеров кластеров

- Адресуются не кластеры файла, а непрерывные области, состоящие из смежных кластеров.
- Для описания области нужны номер первого кластера и количество кластеров в области.
- Достигается сокращение объёма адресной информации.

# Использование экстенентов в NTFS

- LCN (logical cluster number) – порядковый номер кластера в разделе.
- VCN (virtual cluster number) – порядковый номер кластера внутри файла.
- Экстент – непрерывная последовательность кластеров (базовая единица распределения дискового пространства).
- $k$  – количество кластеров в отрезке.
- Адрес экстенента – логический адрес первого кластера отрезка и количество кластеров (LCN,  $k$ ).
- Часть файла, помещённая в отрезок и начинающаяся с адреса VCN – (VCN, LCN,  $k$ ).

# Использование экстенентов в NTFS



Адресация файла (VCN, LCN, k):

- экстенент1 – (0, 1, 4);
- экстенент2 – (4, 8, 3);
- экстенент3 – (7, 13, 1).

# Рассмотренные вопросы

- Функции файловой системы.
- Физическая организация файловых систем.
- Логическая организация файловых систем.
- Способы хранения атрибутов файлов.
- Способы физической организации файла.

**Всем спасибо –  
все свободны,  
если нет вопросов**