

Операционные системы



Операционная система — это комплекс системных управляющих и обрабатывающих программ, которые с одной стороны выступают как интерфейс между аппаратурой компьютера и пользователем, а с другой предназначены для эффективного распределения ресурсов вычислительной системы и организации надежных вычислений.



Операционная система управляет аппаратными ресурсами (распределяет между программами пользователя) и формирует среду исполнения, которая содержит все данные, необходимые для программы.

Такая среда называется *информационным окружением*.



Программа — набор инструкций процессора (исполняемый код), хранящийся на диске.

Для того чтобы программа могла быть запущена на выполнение, операционная система должна создает информационное окружение решаемой задачи.

После этого операционная система перемещает исполняемый код и данные программы в оперативную память и инициирует выполнение программы.



Выполняемая программа образует **процесс**.

Процесс представляет собой совокупность информационного окружения и области памяти, содержащей исполняемый код и данные программы.

Процесс, имеющий потенциальную возможность получить входные данные от пользователя с клавиатуры и вывести результаты своей работы на экран, называется *процессом переднего плана*;

Процесс, выполняемый без непосредственного взаимодействия с пользователем, — *фоновым процессом*.



В ходе своей работы процессы используют вычислительную мощность процессора, оперативную память, обращаются к внешним файлам, внутренним данным ядра операционной системы.

Все эти объекты входят в информационное окружение процесса и называются *ресурсами*.

Ресурсы

```
graph TD; A[Ресурсы] --> B[Неразделяемые]; A --> C[Разделяемые]; C --> D[Одновременного доступа]; C --> E[Доступа с разделением времени];
```

Неразделяемые

Разделяемые

Одновременного
доступа

Доступа с разделением
времени



Неразделяемые ресурсы могут быть использованы на заданном отрезке времени только одним процессом, при этом другие процессы не имеют доступа к такому ресурсу до полного освобождения ресурса занявшим его процессом.

Примером такого ресурса может служить файл, открытый на запись (даже на чтение).

Все попытки использовать этот файл другими процессами завершаются неудачей.



Разделяемые ресурсы могут использоваться несколькими процессами. При этом к таким ресурсам возможен одновременный доступ процессов (например, к часам, при помощи которых определяется текущее системное время).



Примером *ресурса с доступом с разделением времени* может служить процессорное время в многозадачных ОС.

В каждый квант времени выполняется определенное число инструкций одного процесса, после чего управление передается следующему процессу и начинается выполнение его инструкций.



Интервал времени, в течение которого пользователь решает одну или несколько последовательных задач, пользуясь при этом средствами ОС, называется *сеансом*.

Структура операционной системы

В составе операционной системы выделяют два уровня:

- *ядро системы;*
 - *вспомогательные системные программные средства*
(системные утилиты).
-



Ядро выполняет все функции по управлению ресурсами системы — как физическими, так и логическими — и разделяет доступ пользователей (программ пользователей) к этим ресурсам.

В ядро операционной системы входят следующие компоненты:

- система управления сеансами пользователей,
 - файловая система,
 - система управления процессами,
 - система ввода/вывода.
-

Прикладное ПО

Интерфейс системных вызовов

Файловая
Система

Система
управления
сеансами
пользователей

Система
управления
процессами

Система ввода/вывода

Драйверы

Аппаратное обеспечение

Ядро ОС

Интерфейс ядра ОС с прикладными программами осуществляется при помощи *программного интерфейса системных вызовов*, интерфейс с аппаратным обеспечением — при помощи *драйверов*.

Система управления сеансами пользователей

1. осуществляет регистрацию сеанса пользователя при начале его работы с ОС,
 2. хранит оперативную информацию, входящую в информационное окружение сеанса,
 3. при помощи системы ввода/вывода поддерживает соответствие пользовательского терминала реальным или виртуальным устройствам,
 4. корректно завершает сеанс при окончании работы пользователя с системой.
-



Файловая система выполняет преобразование данных, хранимых на внешних запоминающих устройствах в логические объекты — файлы и каталоги.

Система управления процессами

- распределяет ресурсы между выполняемыми задачами (процессами),
 - обеспечивает защиту памяти процессов от модификации ее другими процессами, реализует механизмы межпроцессного взаимодействия.
-



Система ввода/вывода обрабатывает запросы всех рассмотренных выше компонент ядра и преобразует их в вызовы логических устройств, поддерживаемых ОС.

Каждое такое устройство представляет собой логический объект, обращение к которому происходит стандартными для ОС средствами (например, как к адресу в оперативной памяти либо как к специальному файлу).

Драйверы устройств — это системные программы, которые преобразуют запросы системы ввода/вывода в последовательности управляющих команд для аппаратных устройств.

Драйвер каждого устройства скрывает особенности его аппаратной реализации и предоставляет системе ввода/вывода стандартизированный интерфейс доступа к аппаратному обеспечению системы.

Классификация операционных систем

Операционная
система

```
graph TD; A[Операционная система] --> B[однопользовательские]; A --> C[многopользовательские];
```

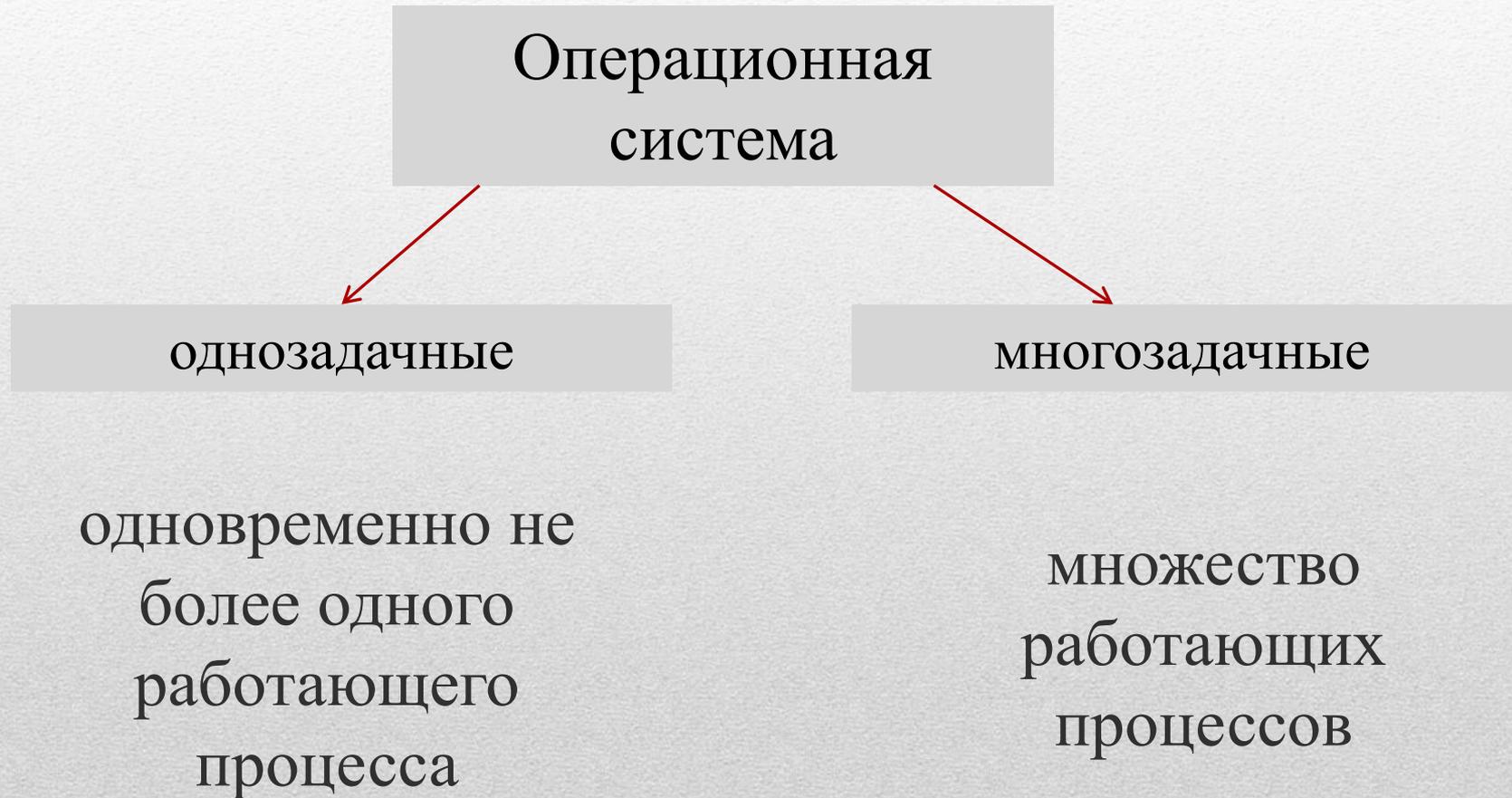
однопользовательские

одновременно
поддерживается не
более одного сеанса
пользователя
(MS DOS)

многopользовательские

одновременно
поддерживается
множество сеансов
пользователей
(Windows, Linux)

Классификация операционных систем



Функции операционных систем

- Прием от пользователя команд;
 - Загрузка в ОЗУ программ и их исполнение;
 - Инициация программы (передача ей управления);
 - Прием и исполнение программных запросов на запуск, приостановку, других программ;
 - Организация взаимодействия между задачами;
-

- Идентификация всех программ и данных;
 - Обеспечение режима мультипрограммирования (многозадачности);
 - Планирование и диспетчеризация задач;
 - Организация ввода-вывода;
 - Управление памятью, организация виртуальной памяти;
-

- Обеспечение сохранности данных;
 - Аутентификация, авторизация;
 - Предоставление услуг на случай частичного сбоя;
-

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Определение ОС.
 2. Информационным окружением называется ...
 3. Что называется программой?
 4. Что происходит при запуске программы на исполнение?
 5. Что мы понимаем под процессом?
 6. Фоновый процесс это ...
 7. К ресурсам ОС относятся ...
 8. Сеанс это ...
 9. Какие функции выполняет ядро ОС?
 10. Перечислите компоненты ядра ОС.
-



Логическая структура магнитного диска

Физический диск — деталь компьютера, вставляемая в системный блок, или монтируемая через внешние порты.

Жёсткий
магнитный
диск ("блин")

Шпиндель Головка

Коромысло

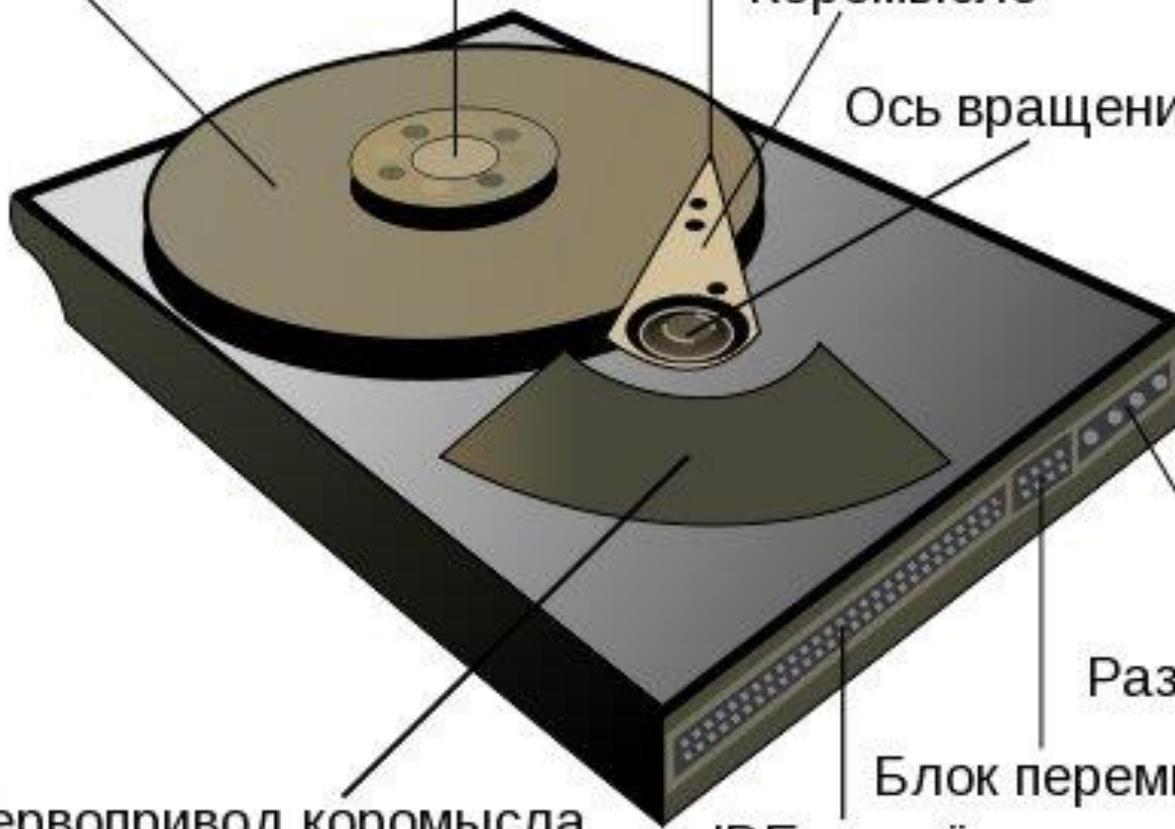
Ось вращения коромысла

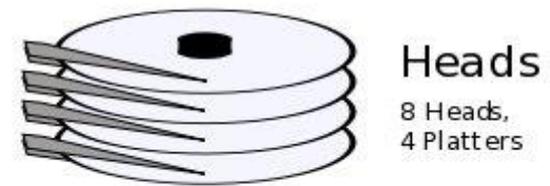
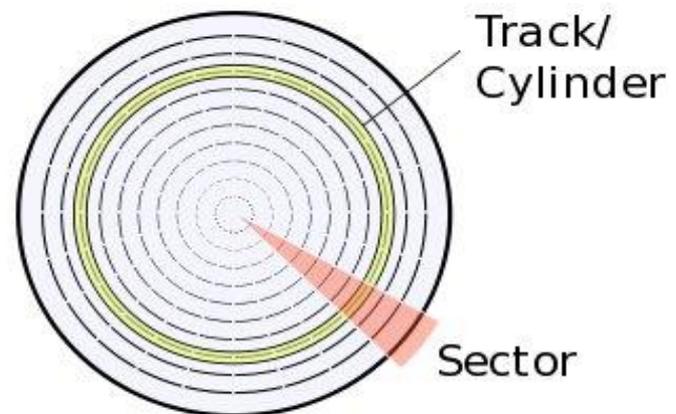
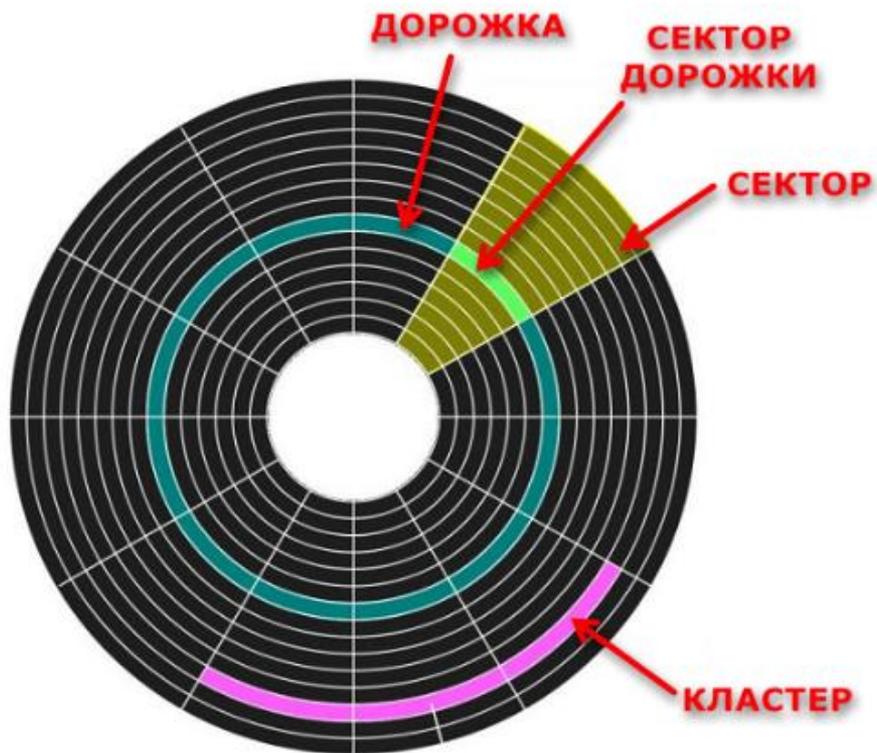
Сервопривод коромысла

IDE разъём

Блок перемычек

Разъём питания





Головка (Head) - электромагнит, скользящий над поверхностью диска, для каждой поверхности используется своя головка. Нумерация начинается с 0.

Дорожка (Track) - концентрическая окружность, которое может прочитать головка в одной позиции. Нумерация дорожек начинается с внешней (первая имеет номер - 0).

Цилиндр (Cylinder) - совокупность всех дорожек с одинаковым номером на всех дисках, т.к. дисков может быть много и на каждом диске запись может быть с двух сторон.

Сектор - на сектора разбивается каждая дорожка, сектор содержит минимальный блок информации. Нумерация секторов начинается от маркера.



Физический адрес сектора на диске определяется с помощью трех «координат» [c-h-s],

где c – номер цилиндра (cylinder),

h – номер рабочей поверхности диска (магнитной головки, head),

s – номер сектора на дорожке.

Например, триада [1-0-2] адресует сектор 2 на дорожке 0 (обычно верхняя рабочая поверхность) цилиндра 1.



Жесткий диск может быть разбит на несколько *разделов* (partition) которые могут использоваться либо одной ОС, либо различными ОС.

На каждом разделе может быть организована своя файловая система.



Разделы диска могут быть двух типов —
primary (первичный) и
extended (расширенный).



Максимальное число primary-разделов равно
четырем.

При этом на диске обязательно должен быть
один primary-раздел.

Согласно спецификациям на одном жестком диске может быть только один *extended*-раздел, который, в свою очередь, может быть разделен на большое количество подразделов – *логических дисков (logical)*.



Один из primary-разделов должен быть *активным*, именно с него должна загружаться программа загрузки операционной системы (*менеджер загрузки*), назначение которого – загрузить программу загрузки ОС из какого-нибудь другого раздела, и уже с её помощью загружать операционную систему.

Поскольку до загрузки ОС система управления файлами работать не может, то следует использовать для указания упомянутых загрузчиков исключительно абсолютные адреса в формате [c-h-s].



По физическому адресу [0-0-1] на винчестере располагается *главная загрузочная запись* (master boot record, MBR).

Master Boot Record — это первый сектор HDD объемом 512 байт, он зарезервирован и никогда не перезаписывается.

В MBR находятся три важных элемента:

- *программа начальной загрузки* (**non-system bootstrap**). Именно она запускается BIOS после успешной загрузки в память первого сектора с MBR. Она, очевидно, не превышает 512 байт и её хватает только на то, чтобы загрузить следующую, чуть более сложную программу, обычно – стартовый сектор операционной системы – и передать ему управление;
-

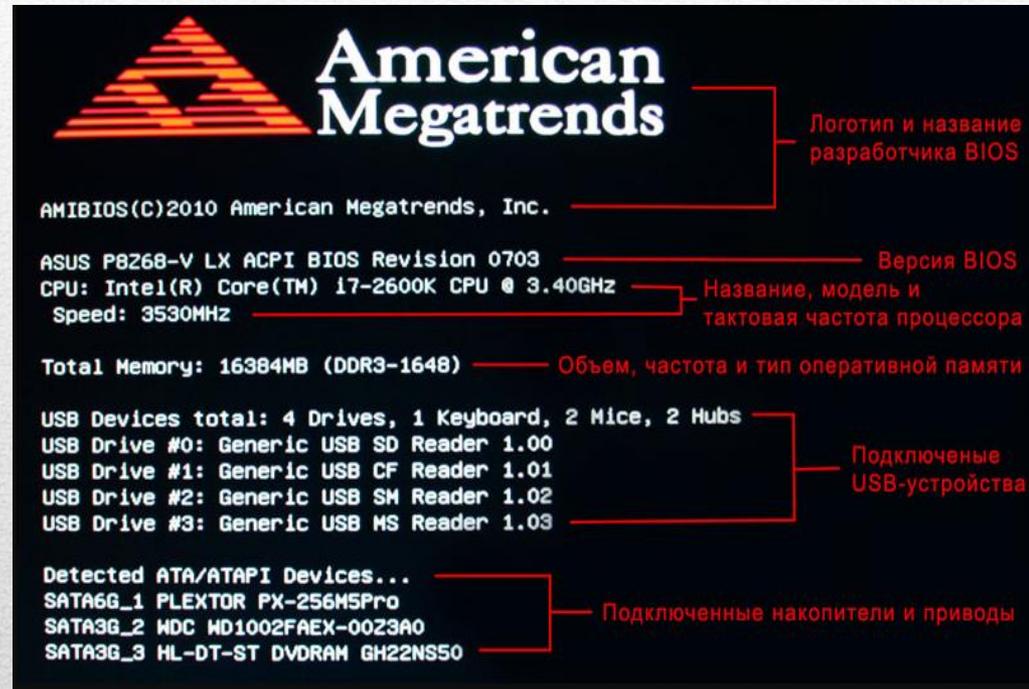
- *таблица описания разделов диска (partition table).* Располагается в MBR по смещению 0x1BE и занимает 64 байта (16 байт на 1 раздел);
 - *сигнатура MBR.* Последние два байта MBR должны содержать число **AA55h**. По наличию этой сигнатуры BIOS проверяет, что первый блок был загружен успешно.
-

Структура MBR

Смещение (Offset)	Размер (Size)	Содержимое (Contents)
0	446	Программа анализа Partition Table и загрузки System Bootstrap с активного раздела жесткого диска
+1BEh	16	Partition 1 entry (Описатель раздела)
+1CEh	16	Partition 2 entry
+1DEh	16	Partition 3 entry
+1EEh	16	Partition 3 entry
+1FEh	16	Сигнатура (AA55h)

Процесс загрузки ОС

- После подачи питания выполняется **POST**.
Начинается тестирование работоспособности основных компонентов ПК.





American Megatrends

Логотип и название
разработчика BIOS

AMIBIOS(C)2010 American Megatrends, Inc.

ASUS P8Z68-V LX ACPI BIOS Revision 0703

Версия BIOS

CPU: Intel(R) Core(TM) i7-2600K CPU @ 3.40GHz

Название, модель и
тактовая частота процессора

Speed: 3530MHz

Total Memory: 16384MB (DDR3-1648)

Объем, частота и тип оперативной памяти

USB Devices total: 4 Drives, 1 Keyboard, 2 Mice, 2 Hubs

USB Drive #0: Generic USB SD Reader 1.00

USB Drive #1: Generic USB CF Reader 1.01

USB Drive #2: Generic USB SM Reader 1.02

USB Drive #3: Generic USB MS Reader 1.03

Подключенные
USB-устройства

Detected ATA/ATAPI Devices...

SATA6G_1 PLEXTOR PX-256M5Pro

SATA3G_2 WDC WD1002FAEX-00Z3A0

SATA3G_3 HL-DT-ST DVDRAM GH22NS50

Подключенные накопители и приводы

- После этого начинает работать процедура – системный ROM BIOS, которая обеспечивает подготовку к начальной загрузке модулей ОЗУ.

Эту функцию осуществляет программа начальной загрузки (*начальный загрузчик*) (**ROM Bootstrap Routine**).

Эта программа определяет первое готовое устройство из списка разрешенных и доступных и пытается загрузить с него в ОЗУ короткую главную **программу-загрузчик (non-system bootstrap из MBR)** и ему передаётся управление.

Main

Advanced

Power

Boot

▶ **Boot Device Priority**

▶ Hard Disk Drives

▶ Boot Settings Configuration

▶ Security

BIOS SETUP UTILITY

Boot

Boot Device Priority

1st Boot Device	[gPXE]
2nd Boot Device	[PXE:NUIDIA Boot Ag]
3rd Boot Device	[PXE:2-NUIDIA Boot]
4th Boot Device	[Hard Drive]

Specifies the boot sequence from the available devices.

A device enclosed in parenthesis has been disabled in the corresponding type menu.



Если диск не системный или его загрузочный сектор поврежден выдается сообщение:

Non-System disk or disk error Replace and press any key when ready

Если операционной системы нет ни на одном из носителей, компьютер выдаст сообщение:

Operating system not found

- Если в ОЗУ успешно считан с винчестера MBR, то управление передается программе **IPL1 (Initial Program Loading)** начальная загрузка программы. Эта программа сканирует содержание Partition Table. Активный (загружаемый) раздел содержит код 80h, остальные разделы должны быть помечены кодом 00h.

Главный загрузчик **IPL1** (non-system bootstrap) определяет на диске активный раздел.



Программа **IPL1** может выдавать на экран три сообщения:

Invalid Partition Table

Если Partition Table содержит более одного загружаемого раздела.

Error loading operation system

Если Boot-сектор активного раздела не удастся считать в память.

Missing operating system

Если в Boot-секторе отсутствует сигнатура 0AA55h, это сообщение может появляться, если винчестер в SETUP выставлен неверно

- Управление передается программе **IPL2**

Первый сектор активного раздела имеет несколько названий: PBS (Partition Boot Sector), Boot Sector, Boot Record, корневая (загрузочная запись), начальный загрузчик.

Первый сектор активного раздела содержит программу Bootstrap или **IPL2 (Initial Program Loading)**.

IPL2 запускает менеджер загрузки Windows 7 (файл bootmgr, который находится в корневой папке активного раздела).

- Менеджер загрузки bootmgr считывает данные конфигурации системы ПК, которые находятся в файле BCD (Boot Configuration data). При наличии нескольких записей в файле BCD будет отображено меню выбора ОС.
Файл BCD находится в папке Boot активного раздела.
-

Выберите операционную систему для запуска или нажмите клавишу TAB
программу:

(Используйте клавиши со стрелками для выделения, затем нажмите кл

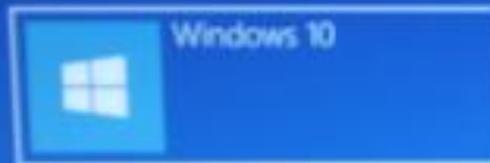
Windows XP

Windows 7

Чтобы задать дополнительный

Выберите операционную систему

Windows 10 будет загружена автоматически через 4 мин 58 с.



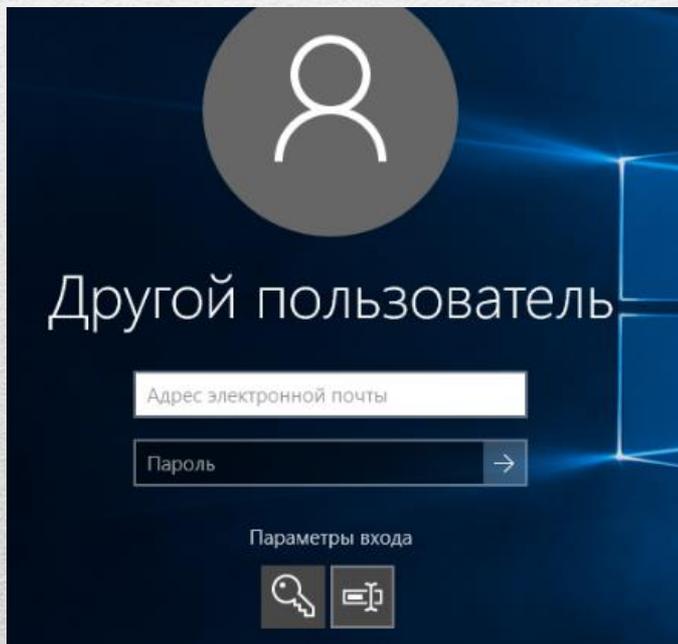
Изменить значения по умолчанию или выбрать другие
параметры

- После выбора системы запускается модуль загрузки ОС **Winload.exe**, компоненты ядра ОС Ntoskrnl.exe, Hal.dll, системные службы и другие компоненты ОС.
Данный этап сопровождается логотипом **Windows**.
-

A blurred, glowing version of the Windows logo, consisting of four colored panes (red, green, blue, yellow) arranged in a cross pattern, set against a black background.

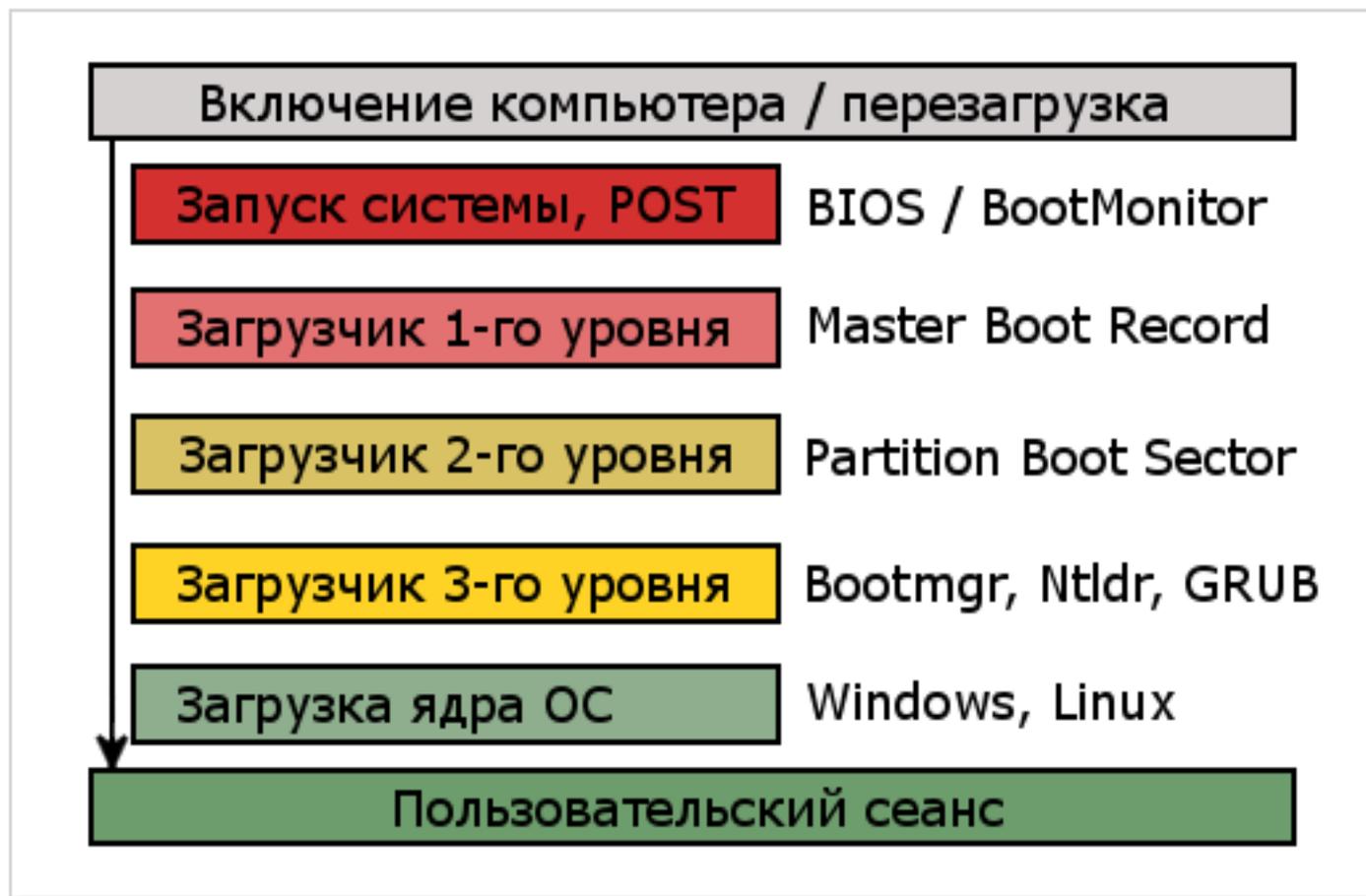
Запуск Windows

- Загружается процесс winlogon.exe, который управляет входом пользователя в систему.



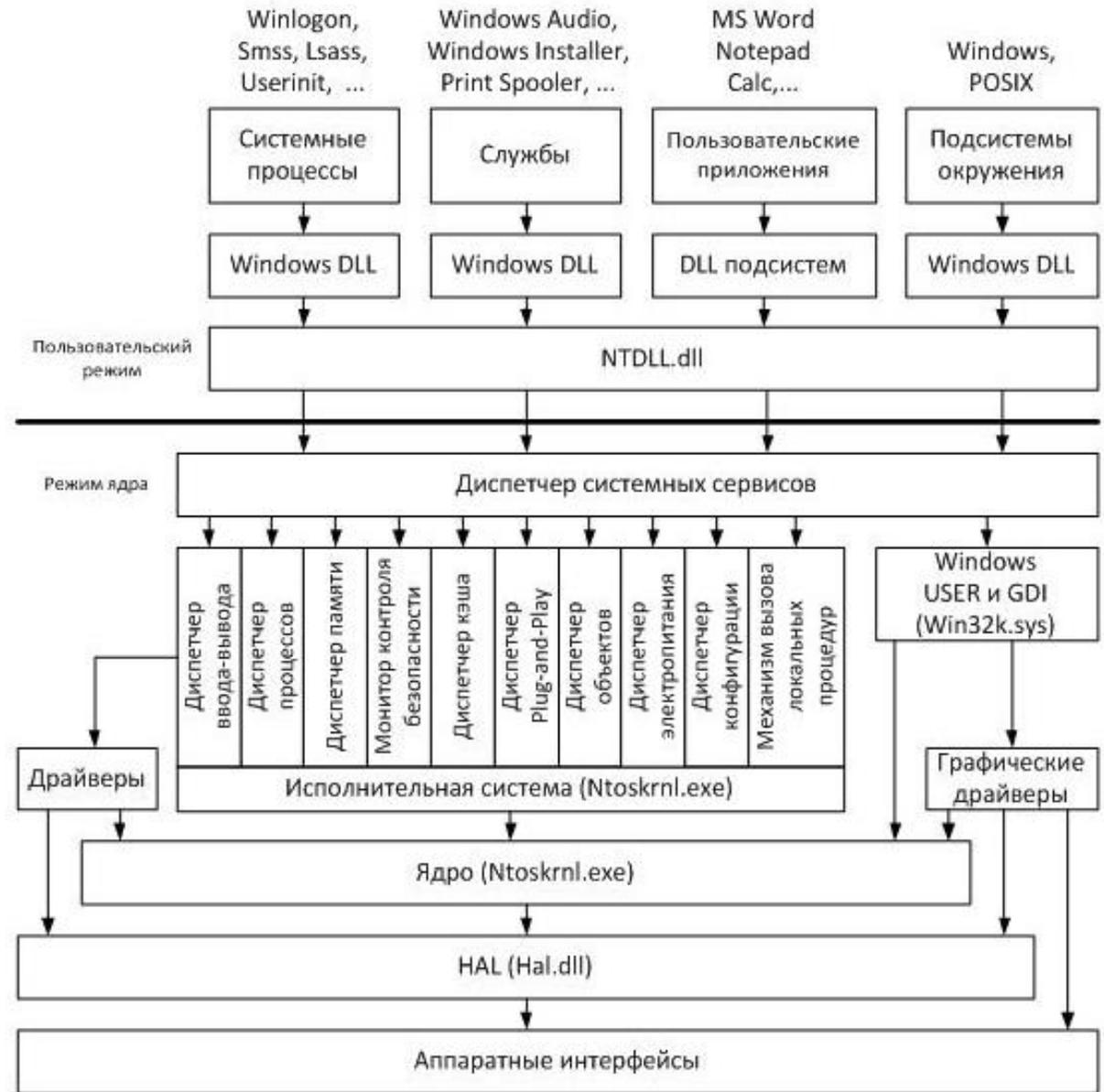
- В процессе входа в систему запускаются элементы автозагрузки, которые прописаны в реестре и в папке Автозагрузка.
-

Основные определения и этапы загрузки операционной системы



Системные файлы Windows 7

Системный файл - это любой файл на компьютере, который требуется для запуска и функционирования операционной системы Windows. Системные файлы обычно находятся в папке Windows или «Program Files».



GPT (GUID Partition Table)

GPT (Таблица разделов GUID, GUID Partition Table) - стандарт формата размещения таблиц разделов на физическом жестком диске.

Он является частью Расширяемого микропрограммного интерфейса (EFI) — стандарта, предложенного Intel на смену BIOS.

EFI использует GPT там, где BIOS использует MBR.



Отличается GPT от MBR она такими параметрами:

- Отсутствие координат CHS, поддерживается работа только с доработанной версией LBA;
 - GPT хранит на накопителе две свои копии — одна в начале диска, а другая в конце. Такое решение позволит реанимировать сектор через хранящуюся копию в случае повреждения;
 - Переработано устройство структуры;
 - Проверка корректности заголовка происходит с помощью UEFI с использованием контрольной суммы.
-



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

Контроллеры современных IDE дисков в качестве основного режима трансляции адреса используют LBA.

Суть LBA состоит в том, что каждый блок, адресуемый на жёстком диске, имеет свой номер, целое число, начиная с нуля и т. д. (то есть первый блок LBA=0, второй LBA=1, ...)

LBA 0 = Цилиндр 0/Головка 0/Сектор 1

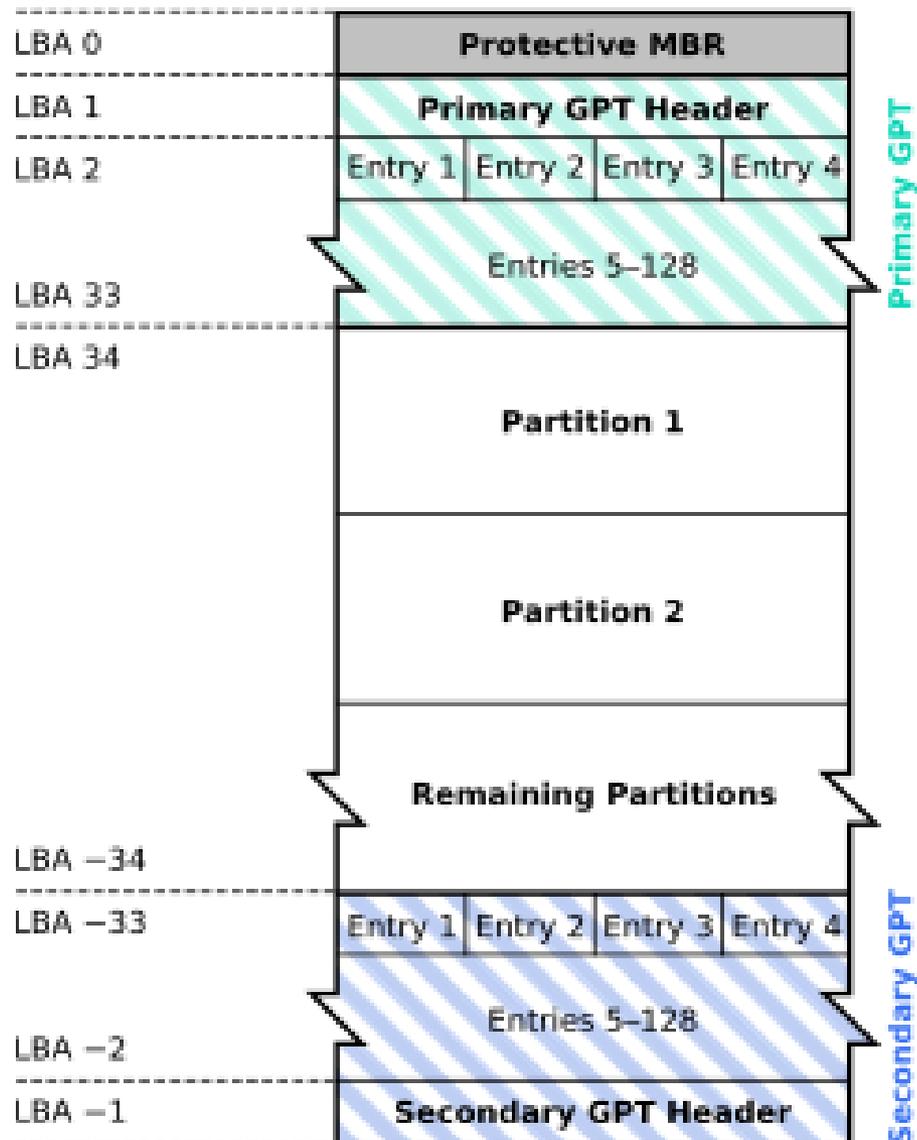
$$LBA = [(Cylinder \cdot noofheads + head) \cdot sectors/track] + (Sector - 1)$$

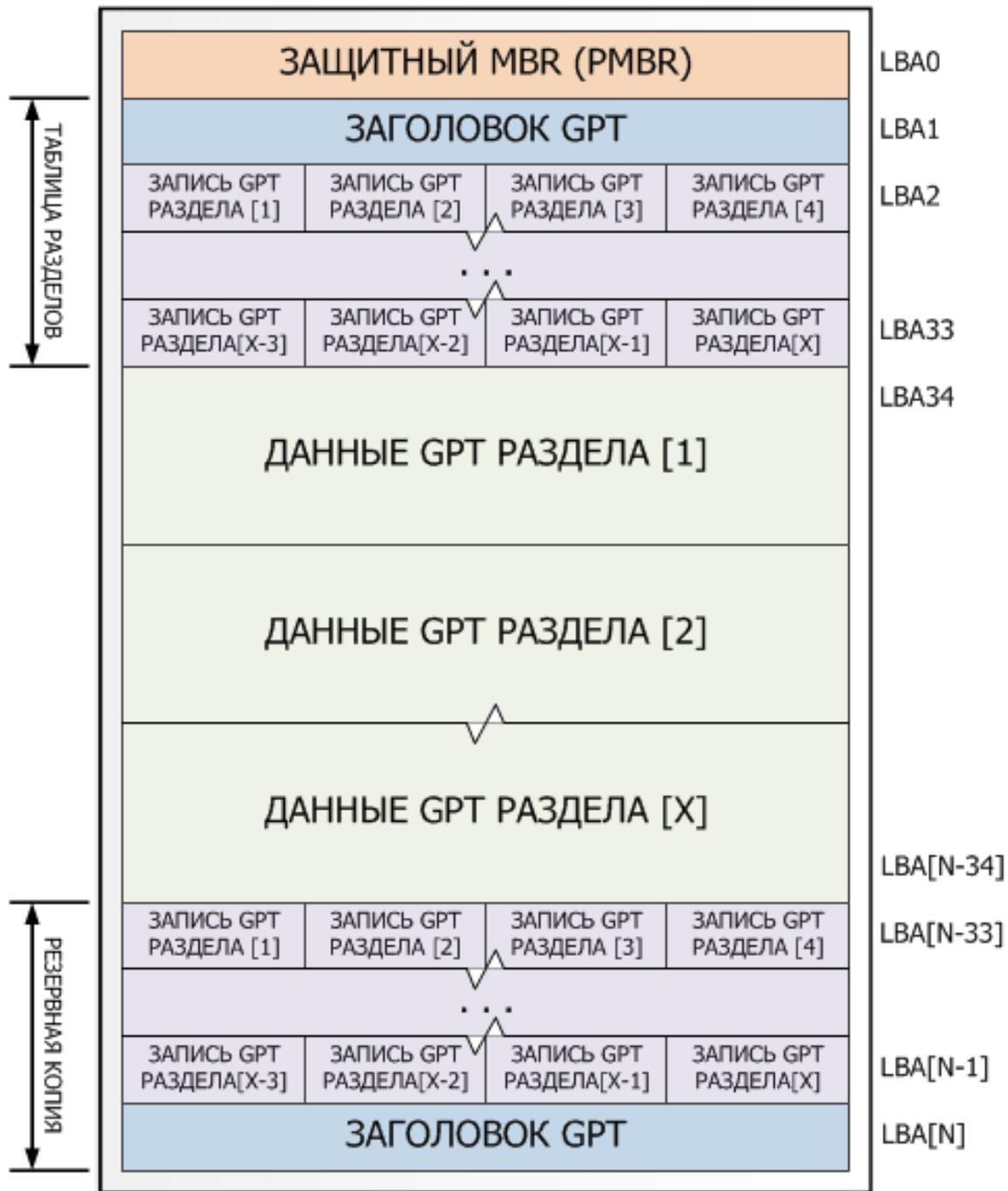
- LBA — адрес блока по LBA.
 - Cylinder — номер цилиндра.
 - noofheads — количество головок.
 - head — номер выбранной головки.
 - sectors/track — количество секторов на одной дорожке.
 - Sector — номер сектора.
-

На этом рисунке приведена диаграмма, схематично поясняющая формат GUID Partition Table.

Каждый логический блок (LBA) имеет размер ровно 512 байт. Отрицательные адреса логических блоков говорят о том, что их нумерация начинается с конца тома (диска), причем последний адресуемый блок имеет адрес -1 .

GUID Partition Table Scheme





Файловые системы

Файловая система – это часть ОС, назначение которой состоит в том, чтобы предоставить пользователю удобный интерфейс при работе с данными, хранящимися на диске, и обеспечить совместное использование файлов несколькими пользователями и процессами.

Операционная система предоставляет прикладным программам интерфейс доступа к файлам, но при этом рассматривает данные, из которых состоят файлы, по-своему.

Набор данных — это более низкоуровневое представление данных, чем файл. Его организация во многом определяется свойствами носителя данных.

Набор данных имеет *уникальное имя*, содержит все *данные*, которые содержит файл, но структура этих данных определяется ядром операционной системы. Кроме того, в набор данных может входить *служебная информация*, недоступная обычным программам.

Все дисковое пространство, используемое файловой системой, разбивается на отдельные блоки — кластеры (обычно имеющие размер 1, 2, 4, 8 или 16 Кбайт).

Каждый кластер имеет свой номер и хранит либо данные пользователя, либо служебную информацию.

Размер кластера устанавливается при создании файловой системы.

Например, служебный блок может хранить последовательность номеров блоков (кластеров), входящих в набор данных

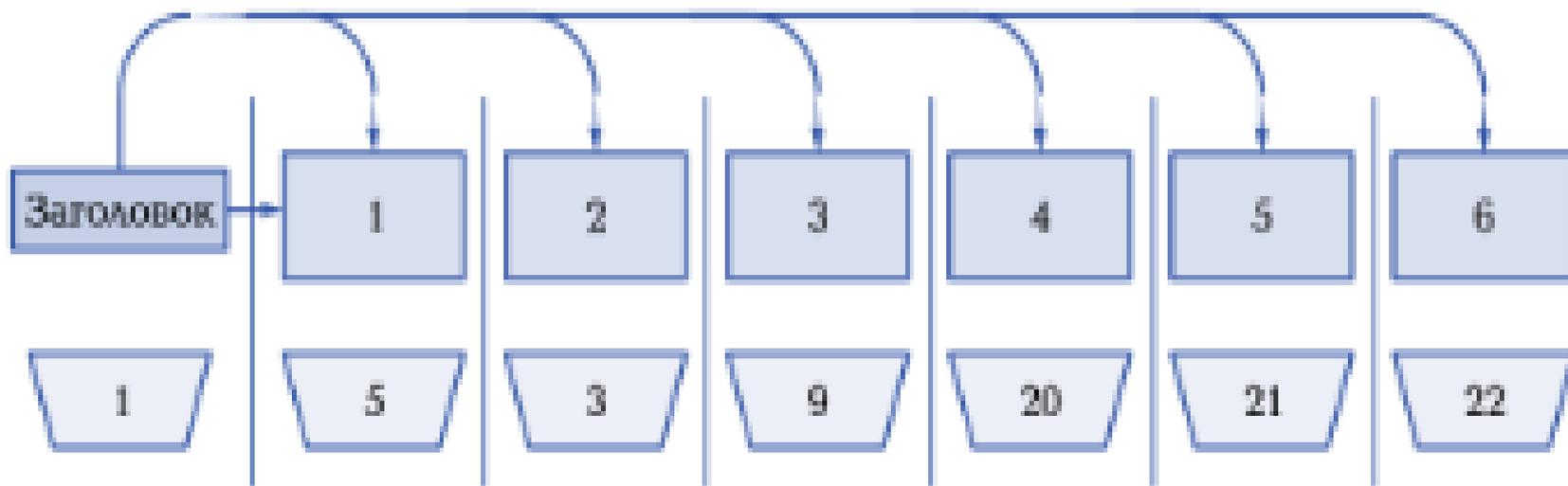


Рис. 2.1. Файловая система со ссылками на все блоки набора данных

Например, наборы данных размещаются в последовательных кластерах, в этом случае в служебном блоке достаточно хранить номера первого и последнего кластеров

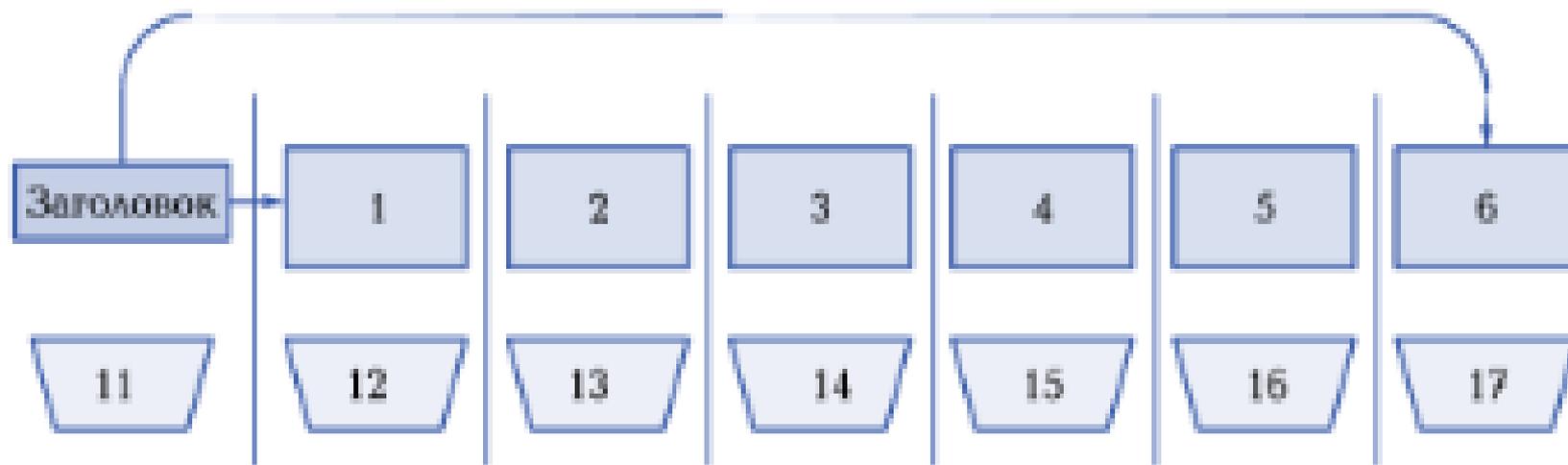


Рис. 2.2. Файловая система с последовательной организацией блоков данных

Еще один способ организации заключается в выделении в каждом блоке небольшой служебной области, в которой хранится номер следующего блока набора данных

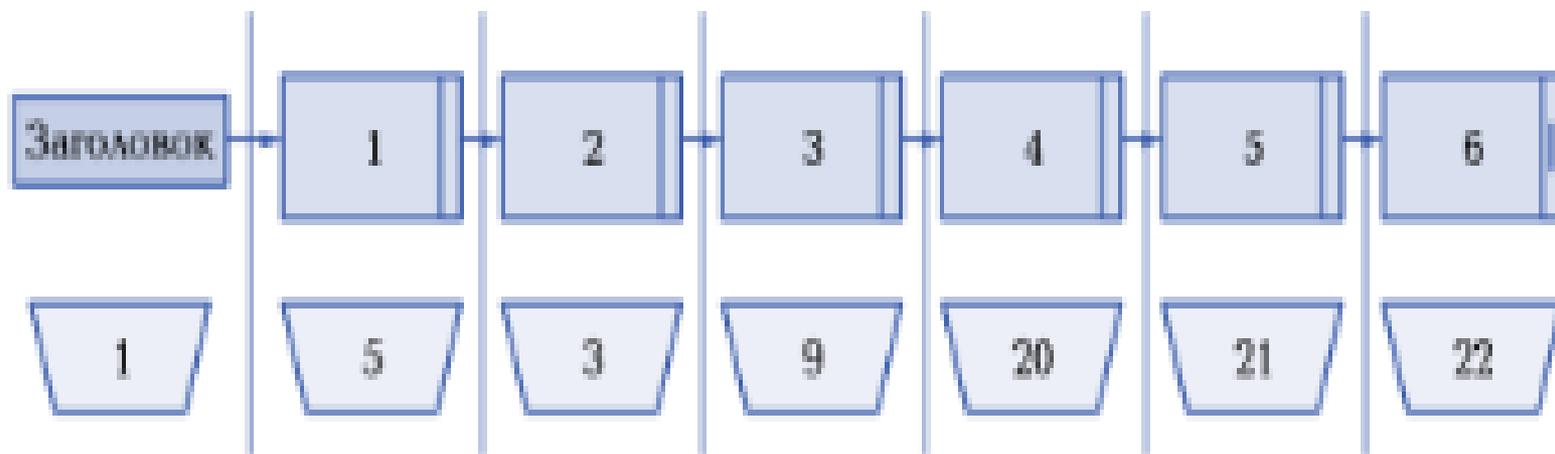


Рис. 2.3. Файловая система со списковой организацией блоков данных ;



Соглашения о структуре наборов данных на носителе являются частью файловой системы, которая состоит из трех компонент:

- 1) соглашения о структуре хранения данных на носителе — определение типов информации, которая может быть размещена на носителе (например, пользовательская/служебная информация), а также набора правил, согласно которым данные размещаются на носителе;
 - 2) соглашения о правилах обработки данных — набор правил, согласно которым данные обрабатываются, например «файл должен быть открыт до того, как в него будут записаны данные»;
 - 3) процедуры обработки данных, которые входят в состав ядра ОС и подчиняются определенным выше соглашениям.
-



В качестве идентификатора набора данных для ОС обычно служит номер кластера служебного блока. Операционная система поддерживает соответствие между идентификаторами наборов данных и именами файлов.



Соответствие имен файлов и идентификаторов наборов данных хранится в специальной области дискового пространства, называемой таблицей размещения файлов.

Каждому набору данных может быть сопоставлено несколько имен файлов.

Каждое такое соответствие представляет собой отдельную запись в таблице размещения файлов и называется в UNIX-системах *жесткой ссылкой*.



Кроме жестких ссылок существуют

символические ссылки — файлы специального вида, хранящие имя файла, на который указывает эта ссылка

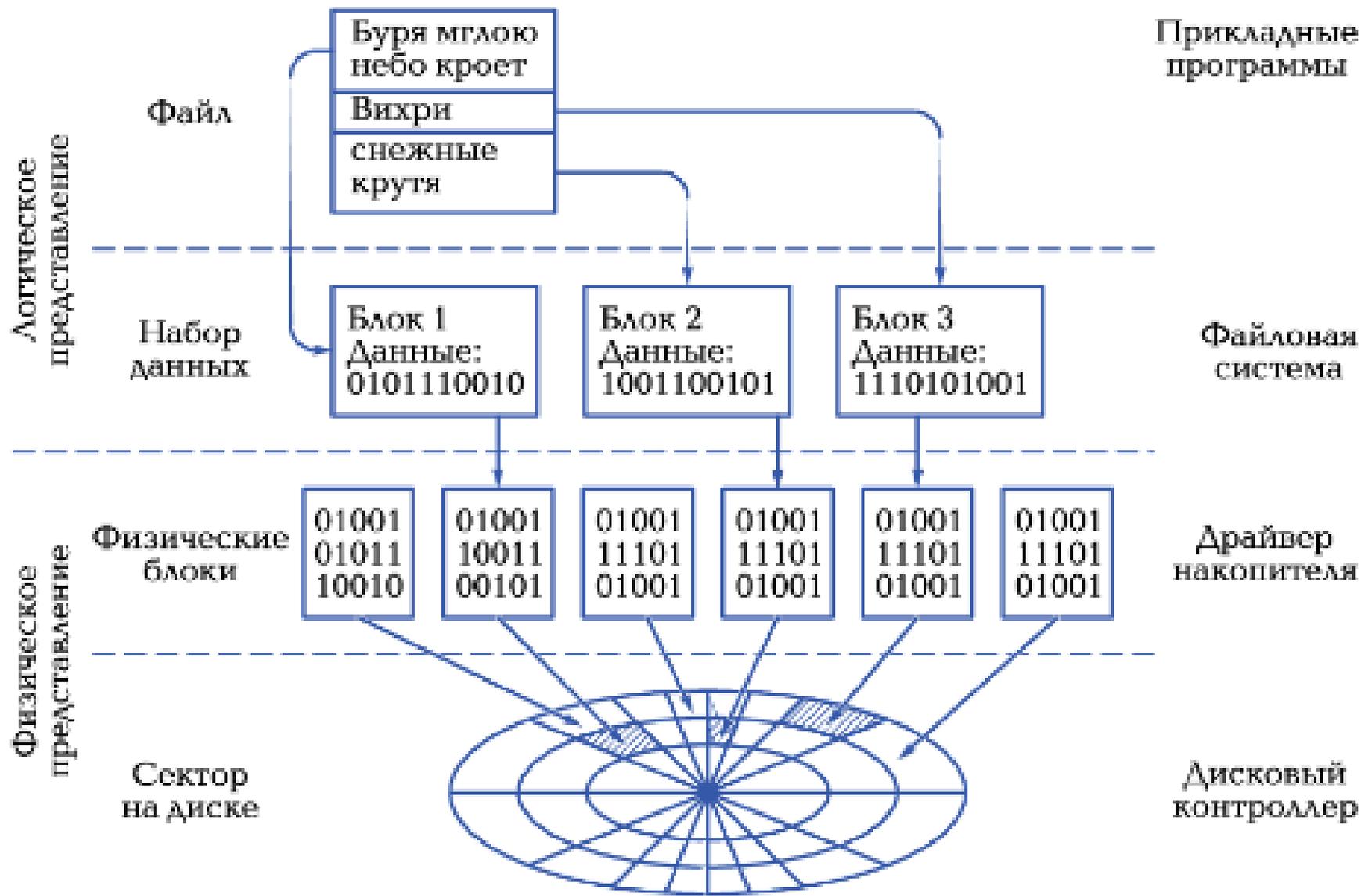


Рис. 2.5. Уровни представления данных:



Рассмотренные выше представления данных задают логическую структуру данных — структуру, удобную для использования программным обеспечением



Если рассматривать физическую структуру данных, то на самом низком уровне данные представлены в виде магнитных доменов (на жестких или гибких дисках), которые объединяются в *сектора* — *минимально адресуемые области диска. Один магнитный домен представляет собой один бит информации, размер сектора обычно равен 512 байт.*

Размер сектора может варьироваться в диапазоне от 512 до 4 096 байт.



Идентифицируется сектор при помощи номера считывающей головки, номера дорожки и номера сектора на дорожке.

При считывании или записи сектора проходят через дисковый контроллер, который управляется драйвером данного дискового накопителя, входящим в ядро ОС.



Интерфейс драйвера с дисковым контроллером определяет режим доступа к устройству:

- при *символьном* доступе информация считывается последовательно по одному символу,
 - при *блочном* — блоками фиксированного размера (обычно кратного размеру сектора).
-

Функции файловой системы:

- определение соглашения об организации данных на носителях информации;
 - определение соглашения о способе доступа к данным (последовательном или произвольном);
 - определение соглашения об именовании файлов;
 - определение соглашения о логической структуре данных;
 - определение набора методов, входящих в состав ядра ОС и предназначенных для работы с данными на носителях по указанным выше соглашениям
-

Каталоги

Каталог — это файл специального вида, в котором записаны имена и атрибуты содержащихся в нем файлов и каталогов.

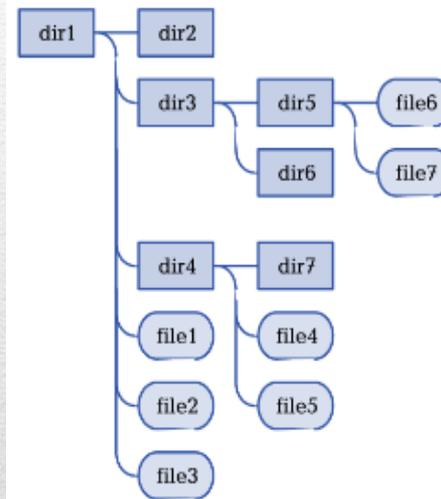


Рис. 2.6. Пример распределения файлов по каталогам



Файл — это именованное место на диске для хранения данных определенного типа.

Полное имя файла — текстовая строка, в которой через специальные символы-разделители указаны имена всех каталогов, задающие путь от корневого каталога до самого файла.

В качестве разделителя в UNIX-системах используется символ косой черты «/».

Например, для файла file6 полным именем будет /dir1/dir3/dir5/file6.

Первая косая черта обозначает *корневой каталог*, т.е. полным именем файла file1, находящегося в корневом каталоге, будет /file1.

Путь, задаваемый полным именем, называется абсолютным путем файла или каталога.

в UNIX-системах для указания того, что путь считается от текущего каталога, часто используют мнемоническое обозначение «.».

Если текущим каталогом является dir3, то относительно него относительный путь к файлу file6 будет задаваться именем

./dir5/file6

Для задания в пути родительского каталога используется мнемоника «..».

Например, если текущим каталогом является каталог dir7, то относительное имя файла file1 будет выглядеть как

../../file1

Первые две точки указывают на каталог, родительский для dir7, — каталог dir4, а вторые две точки — на каталог, родительский для dir4, — каталог dir1.

Системные файлы в UNIX-подобных операционных системах распределены по отдельным каталогам.

Это позволяет структурировать различные файлы по их назначению.

Большинство UNIX-систем имеют стандартную структуру системных каталогов

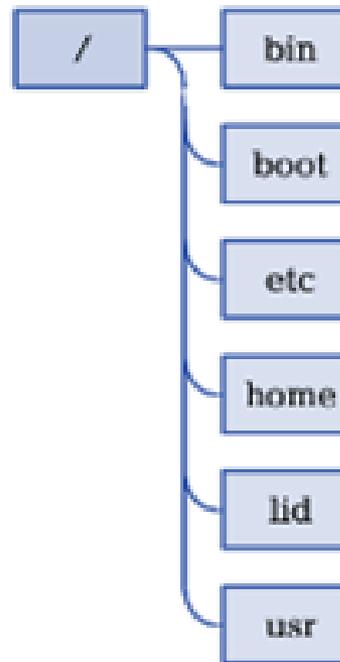


Рис. 2.7. Основные каталоги UNIX-систем

Файловые системы можно классифицировать на следующие категории:

- Для носителей с произвольным доступом (жесткий диск): FAT, FAT16, FAT32, NTFS, EXT и др.;
 - Для оптических носителей (CD, DVD): ISO9660, HFS, UDF и др.;
 - Виртуальные файловые системы: AEFS и др.;
 - Сетевые файловые системы: NFS, CIFS, SSHFS, GmailFS и др.;
 - Для флеш-памяти: YAFFS, ExtremeFFS, exFAT.
-

Принцип организации файловых систем Windows

FAT (file allocation table) «таблица размещения файлов».

В начале располагался загрузочный сектор, затем — две таблицы размещения файлов (File Allocation Table — FAT, которые и дали название файловой системе, причем вторая FAT является полной копией первой и хранится на случай ее повреждения), корневой каталог и файлы.

Загрузочный сектор	Информация о ФС	Дополнительные секторы	FAT #1	FAT #2	Корневой каталог (FAT12/16)	Данные
--------------------	-----------------	------------------------	--------	--------	-----------------------------	--------

Рис. 2.12. Структура файловой системы FAT



Все дисковое пространство в этой файловой системе разбивается на **кластеры**, а таблицы FAT содержат служебные ячейки для каждого кластера.

Информация о файле представляет собой список.

Если ячейка соответствует последнему кластеру в файле, то она содержит значение **FFFFh**.

Неиспользуемые кластеры помечены **0000h**, удаленные (bad – плохой кластер)— **FFF7h**. При удалении файла делается только запись в каталоге, а цепочка кластеров не разрушается.

В настоящее время существует четыре версии FAT: **FAT12**, **FAT16**, **FAT32** и **exFAT**.



Ее особенность заключается в том, что она не предназначена для обработки и хранения больших файлов, а также на нее будет довольно проблематично установить тяжелые программы.

Однако FAT32 универсальна, и при создании внешнего жесткого диска она используется для того, чтобы сохраненные файлы можно было считать с любого телевизора или проигрывателя.



Рис. 1.4. Пример размещения файлов в файловой системе FAT

Недостатки FAT, состоят в следующем:

- файловые системы FAT чрезвычайно подвержены фрагментации данных;
 - перерасходуют дисковое пространство для достаточно больших объемов логических разделов;
 - не поддерживают механизмы восстановления от ошибок, позволяющие восстанавливать работоспособность файловой системы в случае возникновения сбоев или внезапной перезагрузки;
 - не имеют поддержки POSIX, что препятствовало использованию операционных систем Windows в ряде организаций и компаний;
 - не имеют средств разделения данных между пользователями и средств управления правами доступа.
-



Файловая система NTFS основные свойства:

Надежность и восстанавливаемость.

NTFS спроектирована таким образом, что операции ввода/вывода представляют собой транзакции. Эти транзакции неделимы, т.е. требуемая операция либо завершается полностью, либо не выполняется вообще. Это позволяет безболезненно совершать операции отката состояния файловой системы в случае сбоя;

Безопасность и контроль доступа к данным.

Файловая система NTFS трактует файлы и каталоги как защищенные объекты в соответствии с общей архитектурой безопасности Windows. Это позволяет ограничивать доступ к данным определенным пользователям или группам пользователей;



Эффективное распределение дискового пространства. Файловая система NTFS поддерживает целый ряд механизмов управления дисковым пространством, таких как поддержка сжатия каталогов и дисков, поддержка работы с разреженными файлами, наличие специализированного API для дефрагментации данных;

Поддержка POSIX, жестких ссылок, длинных имен, шифрования данных, поддержка работы с логическими томами размером больше 4 Гбайт.

Загрузочный сектор	Мастер File Table	Резерв MFT	Системные файлы	Обычные файлы	Копия первых 16 записей MFT	Обычные файлы
--------------------	-------------------	------------	-----------------	---------------	-----------------------------	---------------

Рис. 2.13. Структура файловой системы NTFS;

Файловая система NTFS содержит несколько системных областей и областей хранения данных.

Одной из основных системных областей является область **Master File Table** (MFT). Данная область содержит информацию обо всех файлах в системе, включая и саму MFT. Когда создается новый файл или какой-либо файл удаляется из системы, то соответствующие изменения производятся и в MFT.

Первые 16 записей в таблице MFT зарезервированы для хранения информации о служебных файлах.
