


Лекция 2.

Функциональные компоненты ОС автономного компьютера

Функциональные компоненты операционной системы автономного компьютера

- Функции операционной системы автономного компьютера обычно группируются либо в соответствии с типами локальных ресурсов, которыми управляет ОС, либо в соответствии со специфическими задачами, применимыми ко всем ресурсам.
- Иногда такие группы функций называют подсистемами.

Подсистемы управления ресурсами



подсистема
управления
процессами

подсистема
управления памятью

подсистема
управления файлами и
внешними
устройствами

Подсистемы, общие для всех ресурсов:

Подсистема
пользовательского
интерфейса

Подсистема защиты
данных

Подсистема
администрирования

Классификация подсистем ОС автономного ПК



Подсистема управления процессами

Основные понятия:

- Под **процессом** в общем случае понимается программа в стадии выполнения
- **Процесс** можно также определить как некоторую заявку на потребление системных ресурсов.
- В **мультипрограммной** операционной системе одновременно может существовать несколько процессов
- Часть процессов порождается по инициативе пользователей и их приложений, такие процессы обычно называют **пользовательскими**
- Процессы, называемые **системными**, инициализируются самой операционной системой для выполнения своих функций.
- Совокупность всех областей оперативной памяти, выделенных операционной системой процессу, называется его **адресным пространством**.

Основные функции подсистемы управления процессами

- создание и уничтожение процессов (т.е. структур данных, связанных с процессами)
- поддержание очередей заявок процессов на ресурсы
- защита ресурсов, выделенных данному процессу, от остальных процессов
- организация совместное использование ресурсов
- обеспечение прерываний и возобновлений некоторого процесса
- функции синхронизации процессов, позволяющие процессу приостанавливать свое выполнение до наступления какого-либо события в системе
- предоставление средств межпроцессного взаимодействия

Подсистема управления памятью (основные функции)

- распределение имеющейся физической памяти между всеми существующими в системе в данный момент процессами (выделение и освобождение памяти)
- загрузка кодов и данных процессов в отведенные им области памяти
- настройка адресно-зависимых частей кодов процесса на физические адреса выделенной области
- защита областей памяти каждого процесса (избирательная способность предохранять выполняемую задачу от записи или чтения памяти, выделенной другой задаче)

Подсистема управления файлами и внешними устройствами

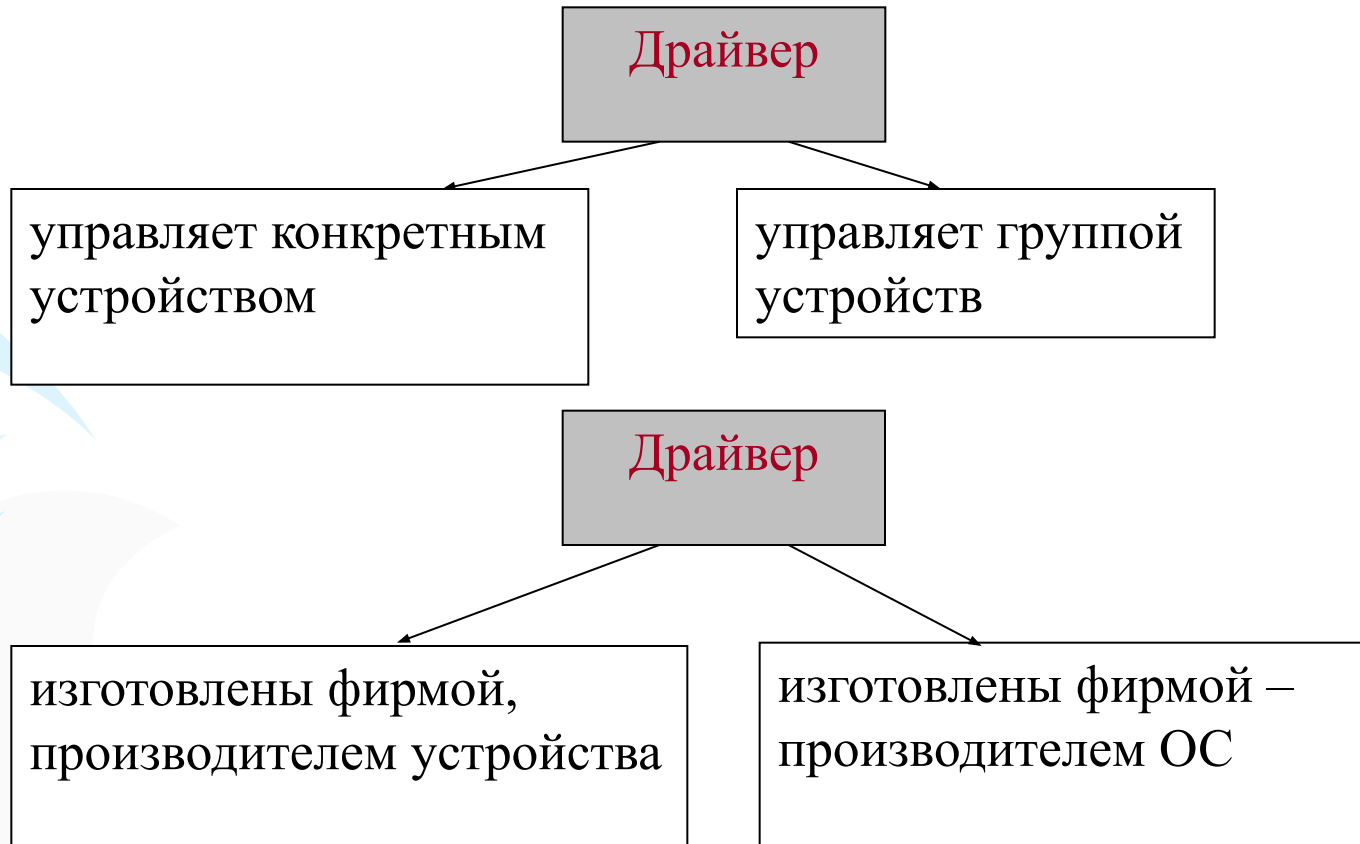
Основные определения:

- **Файл** - простая неструктурированная последовательность байтов, имеющей символьное имя.
- **Драйвер** - программа, управляющая конкретной моделью внешнего устройства и учитывающая все его особенности

Файловая система ОС выполняет:

- преобразование символьных имен файлов, с которыми работает пользователь или прикладной программист, в физические адреса данных на диске,
- организует совместный доступ к файлам,
- защищает их от несанкционированного доступа.

Классификация драйверов



Подсистемы защиты данных

Безопасность данных вычислительной системы обеспечивается:

- средствами отказоустойчивости ОС, направленными на защиту от сбоев и отказов аппаратуры и ошибок программного обеспечения
- средствами защиты от несанкционированного доступа (ОС защищает данные от ошибочного или злонамеренного поведения пользователей системы).

Отказоустойчивость

Поддержка отказоустойчивости реализуется операционной системой, как правило, на основе :

- резервирования
- использования отказоустойчивых дисковых систем
- использования восстанавливаемых ФС

Подсистема пользовательского интерфейса

Возможности операционной системы доступны прикладному программисту в виде набора функций, называющегося интерфейсом прикладного программирования (Application Programming Interface, API).

API- функции используются:

- когда для выполнения тех или иных действий им требуется особый статус, которым обладает только операционная система
- с помощью него программисты могут создавать приложения для этой операционной системы.

Интерфейс прикладного программирования

- для Windows-систем – Win32
- для UNIX-систем - POSIX

Пользовательский интерфейс

- Современные ОС поддерживают развитые функции пользовательского интерфейса для интерактивной работы за терминалами двух типов: алфавитно-цифровыми и графическими.
- При работе за алфавитно-цифровым терминалом пользователь имеет в своем распоряжении систему команд, мощность которой отражает функциональные возможности данной ОС.
- Обычно командный язык ОС позволяет запускать и останавливать приложения, выполнять различные операции с файлами и каталогами, получать информацию о состоянии ОС (количество работающих процессов, объем свободного пространства на дисках и т. п.), администрировать систему.

Пользовательский интерфейс

- Команды могут вводиться не только в интерактивном режиме с терминала, но и считываться из так называемого командного файла, содержащего некоторую последовательность команд.
- Программный модуль ОС, ответственный за чтение отдельных команд или же последовательности команд из командного файла, иногда называют **командным интерпретатором**.
- Ввод команды может быть упрощен, если операционная система поддерживает графический пользовательский интерфейс.