

Назначение и функции операционных систем

(часть 2)

Управление процессорами

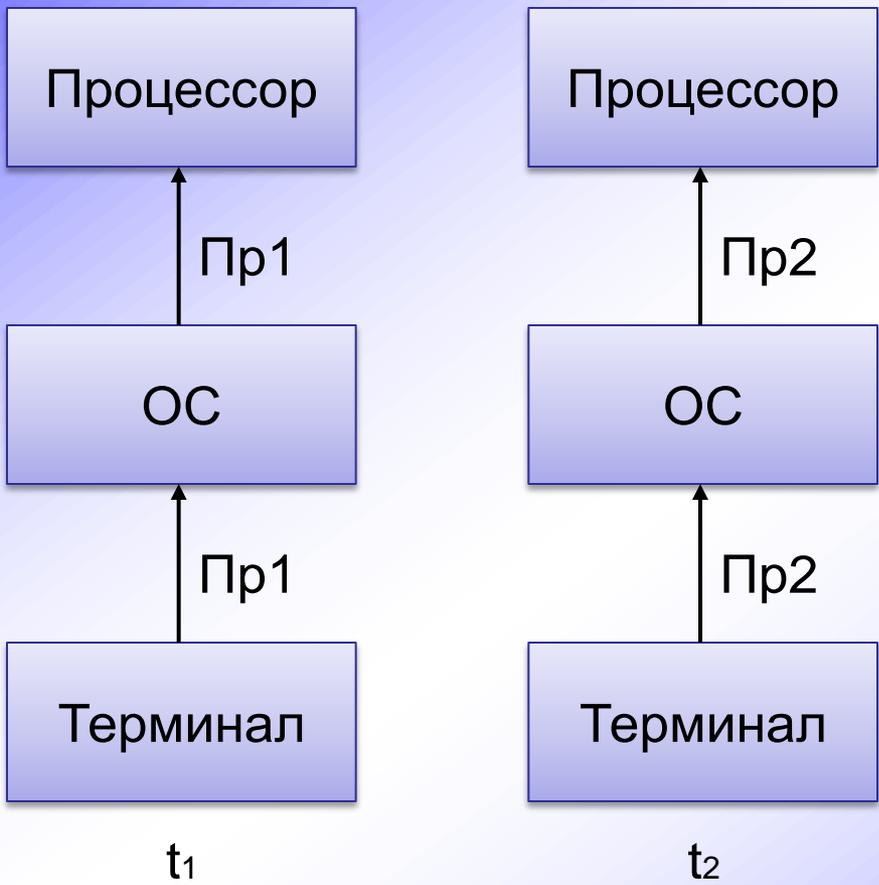
ОС в зависимости от особенностей управления процессорами

- Многозадачные и однозадачные.
- Многопользовательские и однопользовательские.
- Многопроцессорные и однопроцессорные.
- Поддерживающие многопоточную обработку и не поддерживающие её.

Многозадачные и однозадачные ОС

- Многозадачность – способ организации вычислительного процесса, при котором на одном процессоре попеременно выполняется сразу несколько программ и, как следствие, совместно используются различные ресурсы компьютера (Windows, UNIX).
- В однозадачных ОС в один момент времени может быть запущена только одна программа (DOS), поэтому не требуются функции по управлению совместно используемыми ресурсами.

Работа однозадачных ОС

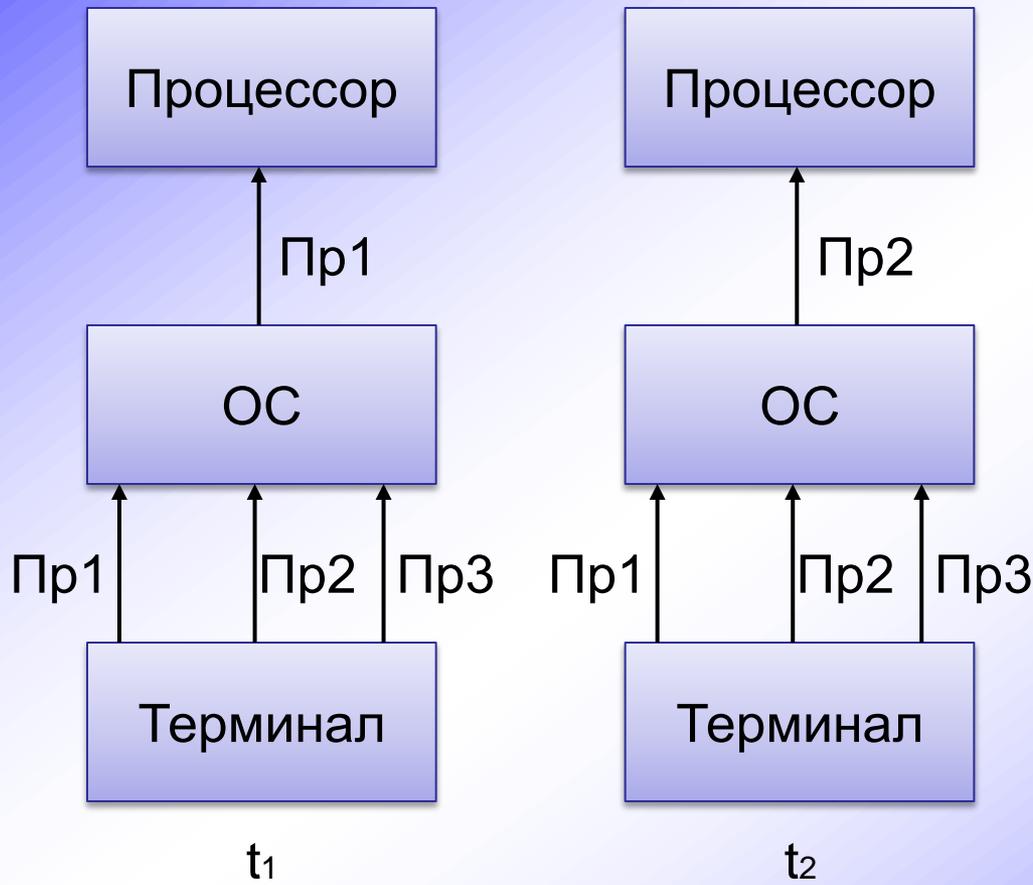


Распределение процессорного времени:

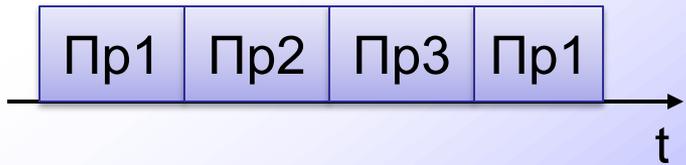


t – процессорное время.
Пр – процесс.

Работа многозадачных ОС

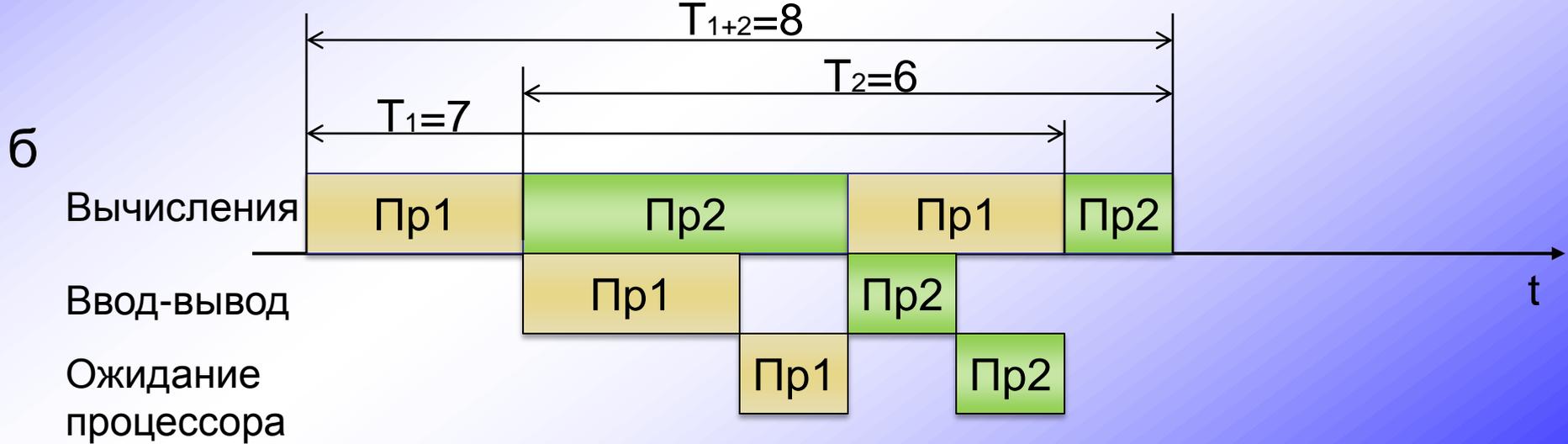
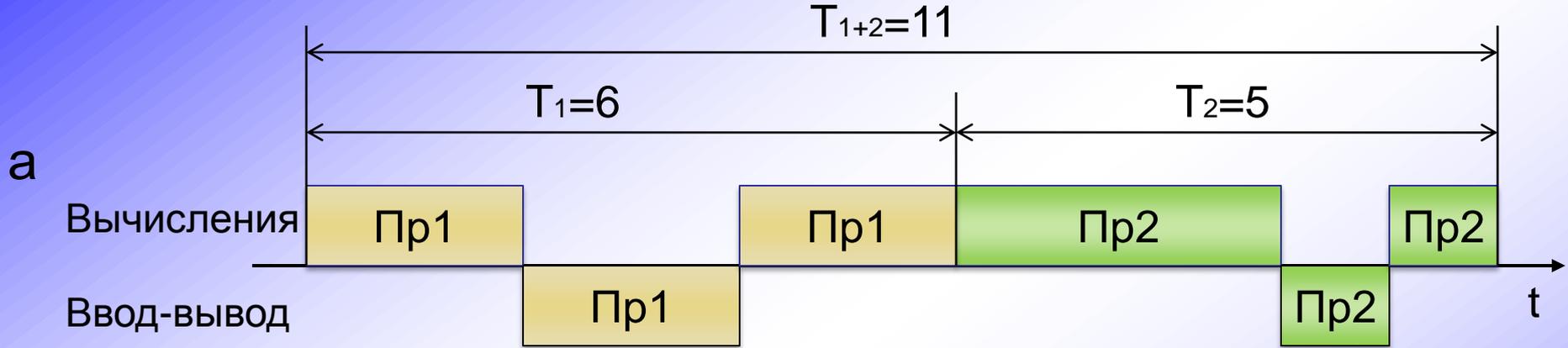


Распределение процессорного времени:



t – процессорное время.
Pr – процесс.

Время выполнения двух задач



а – однозадачная система; б – многозадачная система?

Многопользовательские и однопользовательские ОС

Многопользовательские ОС в отличие от однопользовательских характеризуются наличием нескольких терминалов для работы с вычислительной системой (процессорное время разделяется между процессами разных терминалов).

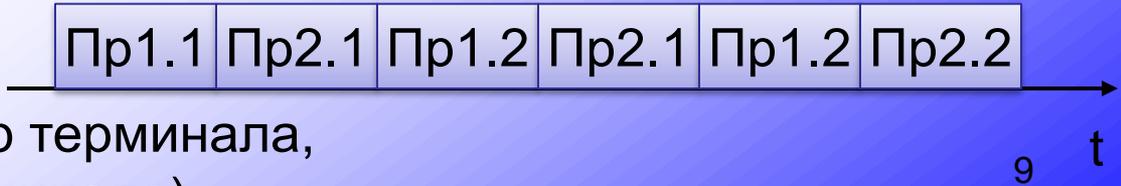
ОС содержит средства защиты информации каждого пользователя от несанкционированного доступа других пользователей.

Работа

МНОГОПОЛЬЗОВАТЕЛЬСКИХ ОС



Распределение процессорного времени:



t – процессорное время.

Pr n.m – процесс (n – номер терминала,

m – номер процесса на терминале).

Многопроцессорные и однопроцессорные ОС

- Способ организации вычислительного процесса в системах с несколькими процессорами, при котором несколько задач могут одновременно выполняться на нескольких процессорах системы.
- Классификация ОС по способу организации вычислительного процесса в системах с многопроцессорной архитектурой: симметричные и асимметричные ОС.

Симметричная архитектура многопроцессорной системы

- Однородность всех процессоров и единообразие включения процессоров в общую схему системы.
- Традиционные симметричные мультипроцессорные конфигурации предоставляют одну большую память всем процессорам.

Особенности симметричной архитектуры

- Масштабирование по вертикали – требование единого корпуса, как следствие ограничение числа процессоров до 4 или 8.
- Быстрый обмен данными, как следствие высокая производительность приложений, в которых несколько задач активно взаимодействуют между собой.

Асимметричная архитектура мультипроцессорной системы

- Разные процессоры могут отличаться своими характеристиками и функциональной ролью.
- Процессоры могут иметь различные схемы подключения, набор периферийных устройств, способы взаимодействия с устройствами.

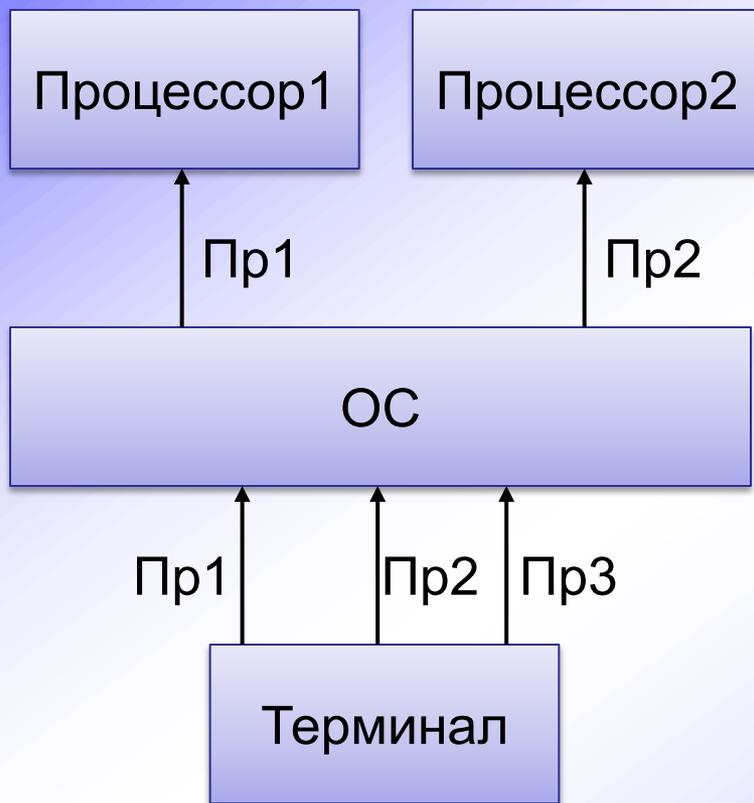
Особенности асимметричной архитектуры

Масштабирование по горизонтали – система может состоять из нескольких устройств (кластеров), содержащих один или несколько процессоров.

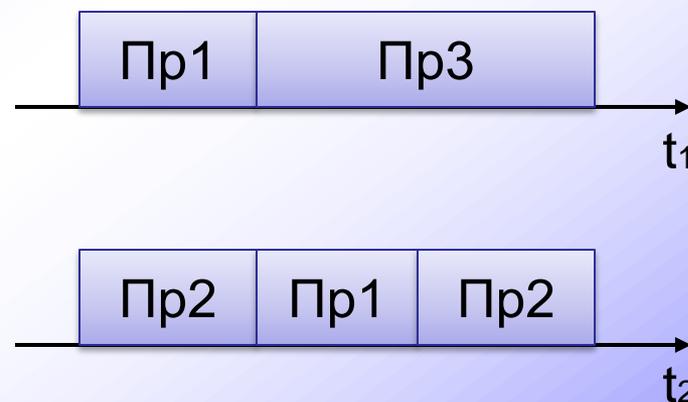
Симметричные ОС

- Все процессоры равноправно участвуют как в управлении вычислительным процессом, так и в выполнении прикладных задач.
- Модули ОС могут одновременно обслуживаться различными процессорами (программы ОС должны быть реентерабельны).
- Осуществима только при симметричной многопроцессорной архитектуре.

Работа симметричных многопроцессорных ОС



Распределение процессорного времени:



t_1 – время работы процессора 1.

t_2 – время работы процессора 2.

Пр – процесс.

Асимметричные ОС

- Выделение одного из процессоров в качестве «ведущего», на котором работает ОС и который управляет остальными «ведомыми» процессорами.
- «Ведущий» распределяет задачи и ресурсы, а «ведомые» используются в качестве обрабатывающих и выполняют прикладные задачи.

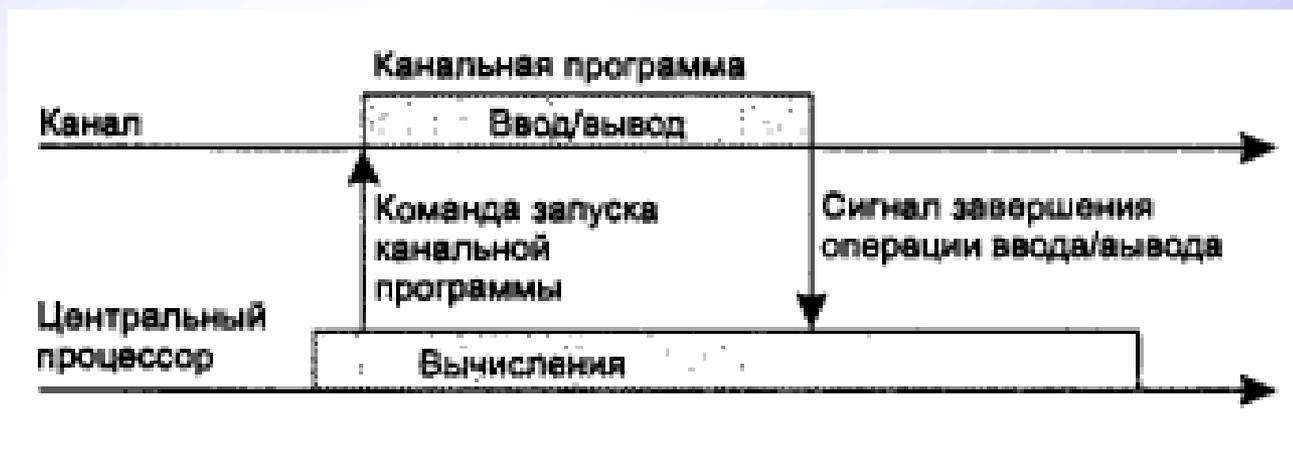
Работа асимметричных многопроцессорных ОС



Совмещение во времени операций ввода-вывода и вычислений

При использовании асимметричных ОС возможно наличие специализированного процессора ввода-вывода (канала):

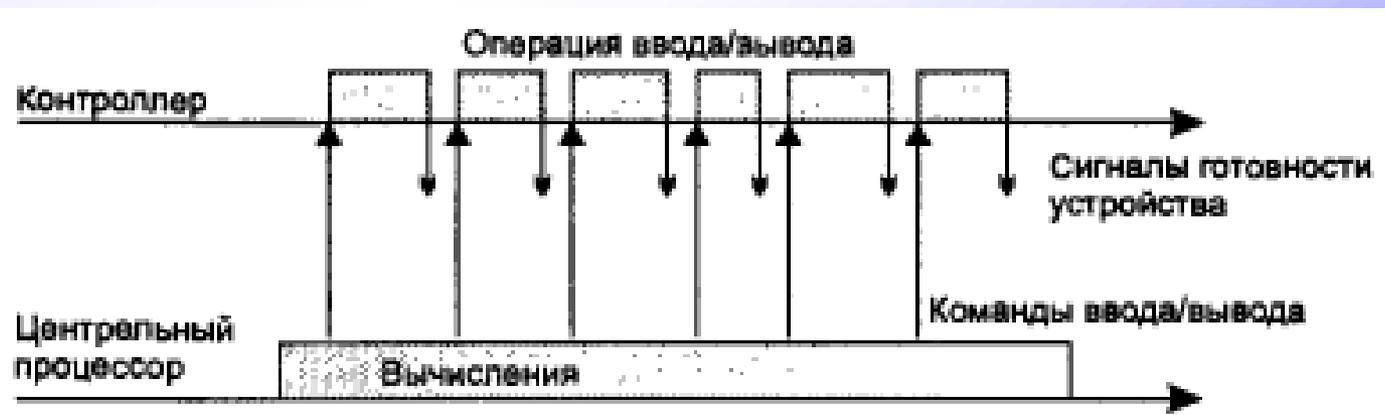
- у канала имеется своя система команд;
- в системе команд ЦП есть инструкция запуска канальной программы.



Совмещение во времени операций ввода-вывода и вычислений

При использовании однопроцессорных или симметричных ОС необходимо наличие контроллеров ввода-вывода, обрабатывающих команды ЦП:

- работа контроллера медленнее, чем работа ЦП, поэтому ЦП выполняет вычисления между передачами команд контроллерам;
- готовность контроллера к работе определяется сигналом прерывания либо опросом контроллеров центральным процессором.



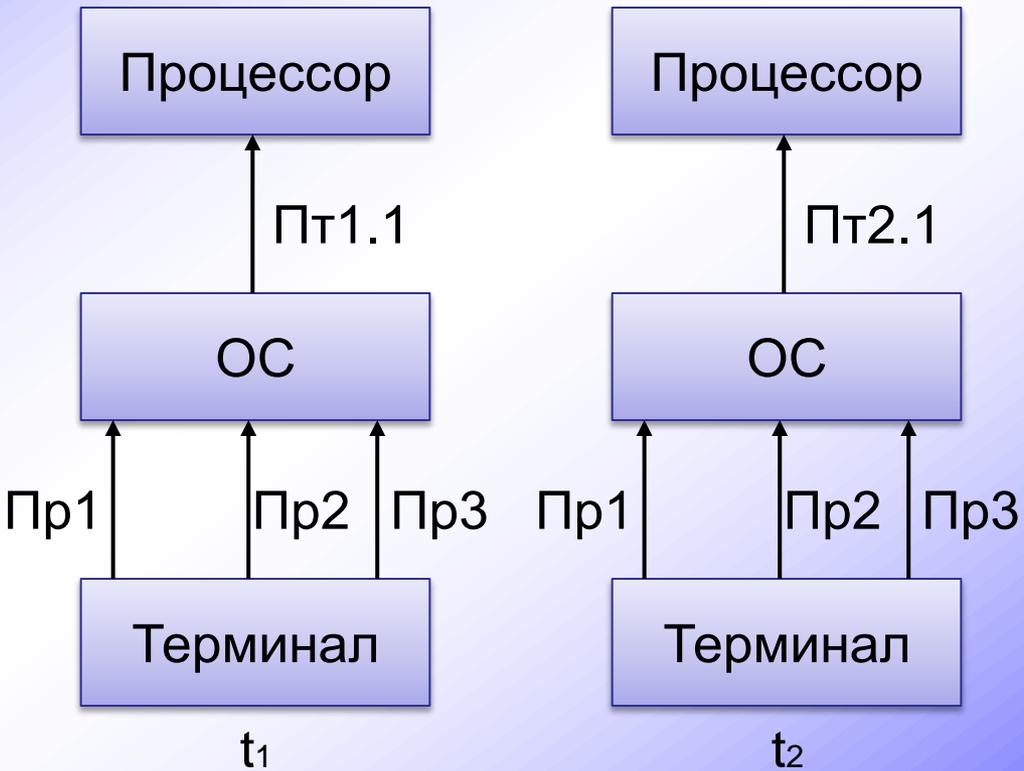
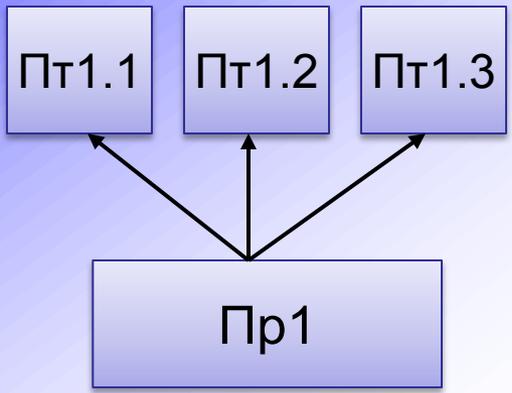
Особенности симметричных и асимметричных ОС

- Более простая реконфигурация симметричных ОС при выходе из строя одного из процессоров.
- На основе асимметричной архитектуры может использоваться только асимметричная ОС.

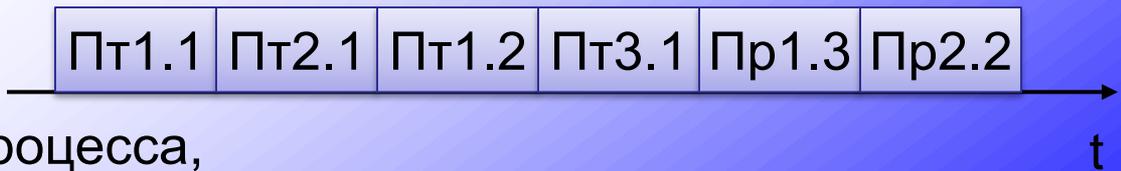
Многопоточные и однопоточные ОС

- Многопоточность – свойство ОС, заключающееся в возможности существования независимо выполняющихся частей процесса – потоков.
- Фактически, многопоточность является реализацией многозадачности в рамках процесса.

Работа многопоточных ОС



Распределение процессорного времени:



t – процессорное время.

Пр – процесс.

Пт n.m – поток (n – номер процесса, m – номер потока в рамках процесса).

Эффективность работы ОС

Эффективность работы ОС

- Эффективность работы ОС зависит от эффективности алгоритмов управления ресурсами вычислительной системы.
- Для различных задач применяются разные критерии эффективности работы ОС.

Критерии эффективности работы ОС

- Пропускная способность – количество задач, выполняемых вычислительной системой в единицу времени.
- Удобство работы пользователей.
- Реактивность системы – способность системы выдерживать заранее заданные интервалы времени между запуском программы и получением результата.

Классификация ОС в зависимости от критерия эффективности

- Системы пакетной обработки (пропускная способность).
- Системы разделения времени (удобство работы пользователей).
- Системы реального времени (реактивность системы).

Функционирование систем пакетной обработки

- Формирование пакета заданий.
- При выполнении очередной задачи выбирается следующая задача, использующая свободные ресурсы (следствие – существует вероятность невыполнения задачи в течение определённого интервала времени).

Результат использования систем пакетной обработки

- Уменьшение времени выполнения всего пакета задач – увеличение эффективности работы аппаратуры.
- Увеличение времени выполнения любой отдельной задачи – снижение эффективности работы пользователя.

Функционирование систем разделения времени

- Цель – повышение удобства и эффективности работы пользователя.
- Пользователю предоставляется возможность интерактивной работы сразу с несколькими приложениями за счёт принудительной периодической остановки работы приложения. Всем запущенным приложениям попеременно выделяется квант процессорного времени.

Результат использования систем разделения времени

- Снижение эффективности работы аппаратуры – принимается на выполнение не «выгодная», а каждая задача, а также происходит выделение процессорного времени для переключения с задачи на задачу.
- Увеличение эффективности работы пользователя – время отклика каждого приложения невелико, поэтому создаётся ощущение параллельного выполнения приложений.

Функционирование систем реального времени

- Цель – управление техническими объектами и технологическими процессами, когда существует предельное время реакции системы (время между запуском программы и получением результата).
- Выбор программы осуществляется по прерываниям (исходя из текущего состояния объекта) или по расписанию плановых работ.

Требования к системам реального времени

- Быстрое переключение с задачи на задачу, быстрая обработка сигналов прерывания.
- Наличие резерва ресурсов аппаратуры на случай пиковых нагрузок.

Структура и типы ядра ОС

Многослойная структура ядра



Типовые средства аппаратной поддержки ОС

- Средства поддержки привилегированного режима.
- Система прерываний.
- Средства защиты областей памяти.
- Средства трансляции адресов.
- Средства переключения процессов.
- Системный таймер.

Средства поддержки привилегированного режима

- Обычно основаны на системном регистре процессора («слово состояния» процессора).
- Функция – проверка допустимости выполнения активной программой инструкций процессора при текущем режиме привилегированности.

Средства трансляции адресов

- Таблицы трансляции хранятся в оперативной памяти, а процессор содержит указатели на эти области оперативной памяти.
- Функция – преобразование виртуальных адресов, содержащихся в коде процесса, в адреса физической памяти.

Средства переключения процессов

- Контексты процессов хранятся в областях оперативной памяти, поддерживаемых указателями процессора.
- Функция – сохранение контекста приостанавливаемого процесса и восстановление контекста активизирующегося процесса.

Система прерываний

Функция – синхронизация выполнения процессов по отношению к устройствам ввода-вывода за счёт прерывания выполнения текущей задачи и вызова процедуры обработки прерывания.

Системный таймер

- Чаще всего – быстродействующий регистр-счётчик.
- Функция – слежение за расходом процессорного времени каким-либо процессом.

Средства защиты областей памяти

Функция – сравнение уровней привилегий текущего кода процессора и сегмента памяти, к которому производится обращение, при операциях чтения, записи или выполнения.

Машинно-зависимые компоненты ОС

- Программные модули, отражающие специфику аппаратной платформы компьютера и позволяющие отделить более высокие слои от особенностей аппаратуры.
- Данный слой, в отличие от более высоких, изменяется для различных аппаратных платформ. За счёт изменения только этого слоя достигается аппаратная независимость ОС.

Машинно-зависимые компоненты ОС



Базовые механизмы ядра

- Переключение контекстов процессов.
- Диспетчеризация прерываний.
- Перемещение страниц из памяти на диск и обратно.

Менеджеры ресурсов

- Учёт свободных и используемых ресурсов (определённого типа для каждого менеджера).
- Планирование распределения ресурсов.

Возможно разбиение на слои: драйверы устройств (нижние), драйверы файловых систем, протоколов сетевых служб (верхние).

Интерфейс системных ВЫЗОВОВ

- Обслуживание системных вызовов.
- Предоставление доступа приложению к ресурсам системы без указания деталей их физического расположения.

Организация аппаратной независимости при помощи машинно-зависимых компонентов ОС

- Ограничение числа поддерживаемых аппаратных конфигураций.
- Применение в качестве машинно-зависимых компонентов низкоуровневых драйверов устройств.
- Использование BIOS для хранения базовых драйверов устройств, поддерживаемых при данной аппаратной конфигурации.

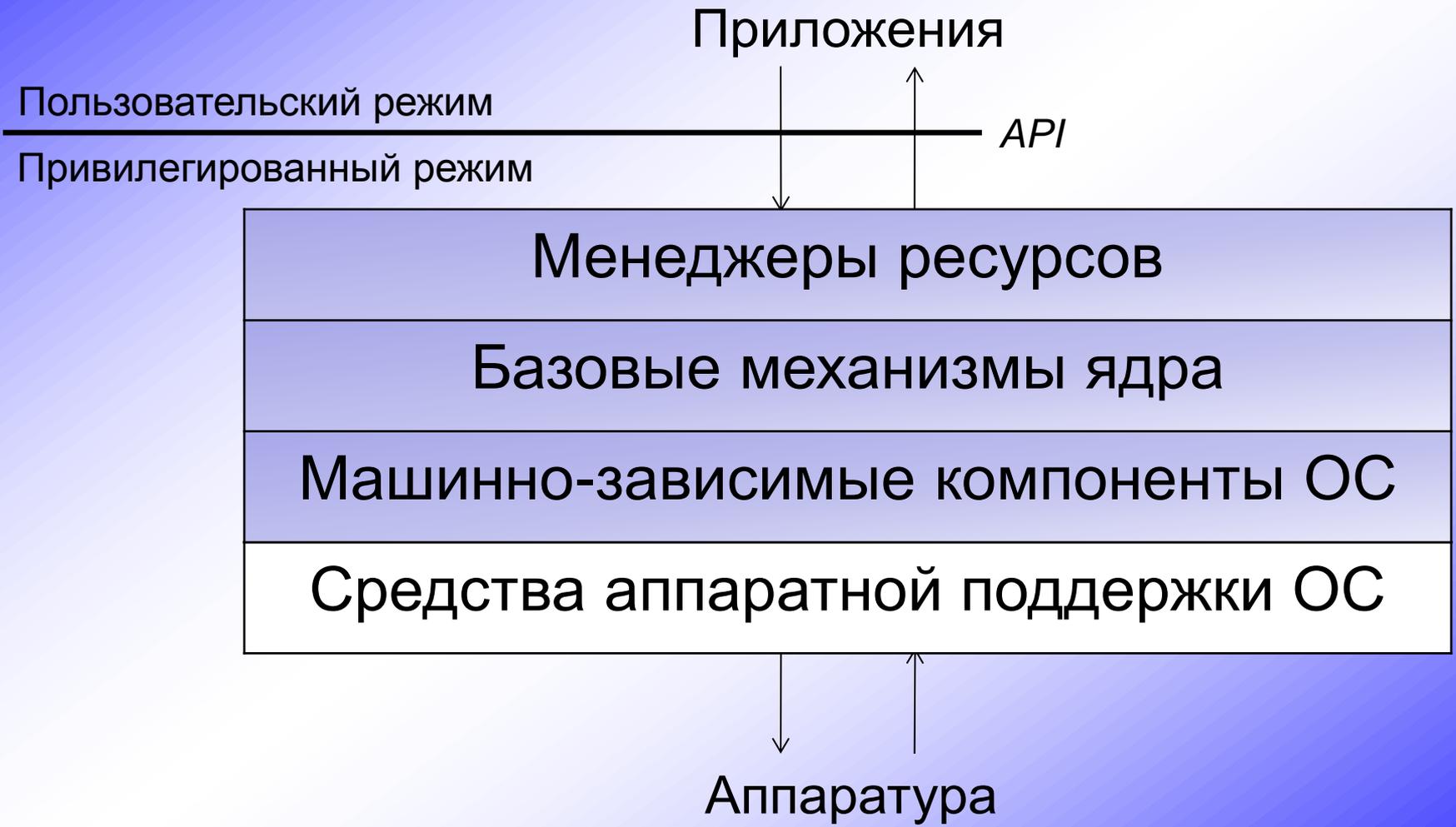
Типы ядра ОС

- Монолитное.
- Модульное.
- Микроядро.
- Гибридное.
- Наноядро.
- Экзоядро.

Монолитное ядро

- Монолитное ядро – архитектура ядра, при которой все подсистемы ОС включены в ядро и работают в одном адресном пространстве.
- Недостатки – высокие требования к объёму оперативной памяти; усложнение отладки, понимания кода ядра; усложнение добавления новых функций; необходимость перекомпиляции ядра при добавлении новых устройств.

Монолитное ядро



Модульное ядро

- Модульное ядро – усовершенствованная модификация архитектуры монолитных ядер операционных систем компьютеров.
- Вместо перекомпиляции ядра используется механизм подгрузки модулей ядра, поддерживающих различные устройства. Все модули работают в адресном пространстве ядра.
- Модульное ядро предоставляет программный интерфейс (API) для связывания модулей с ядром, для обеспечения динамической подгрузки и выгрузки модулей.

Микроядро

Микроядро предоставляет небольшой набор системных вызовов, реализующих базовые сервисы операционной системы:

- управление адресным пространством оперативной памяти;
- управление адресным пространством виртуальной памяти;
- управление процессами и потоками;
- средства межпроцессного взаимодействия.

Микроядро

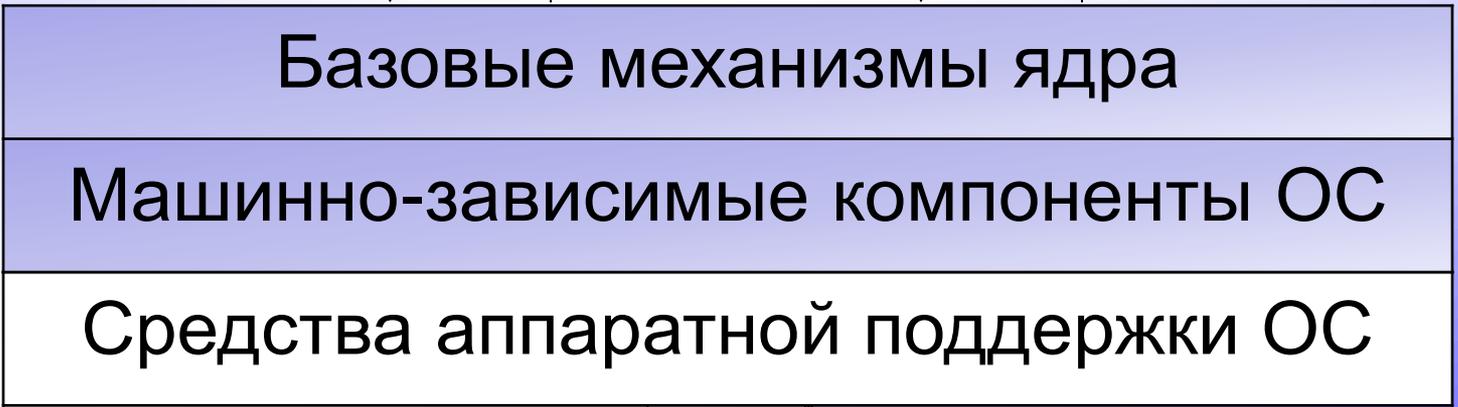
Приложения

Менеджеры
ресурсов

Пользовательский режим

Привилегированный режим

API

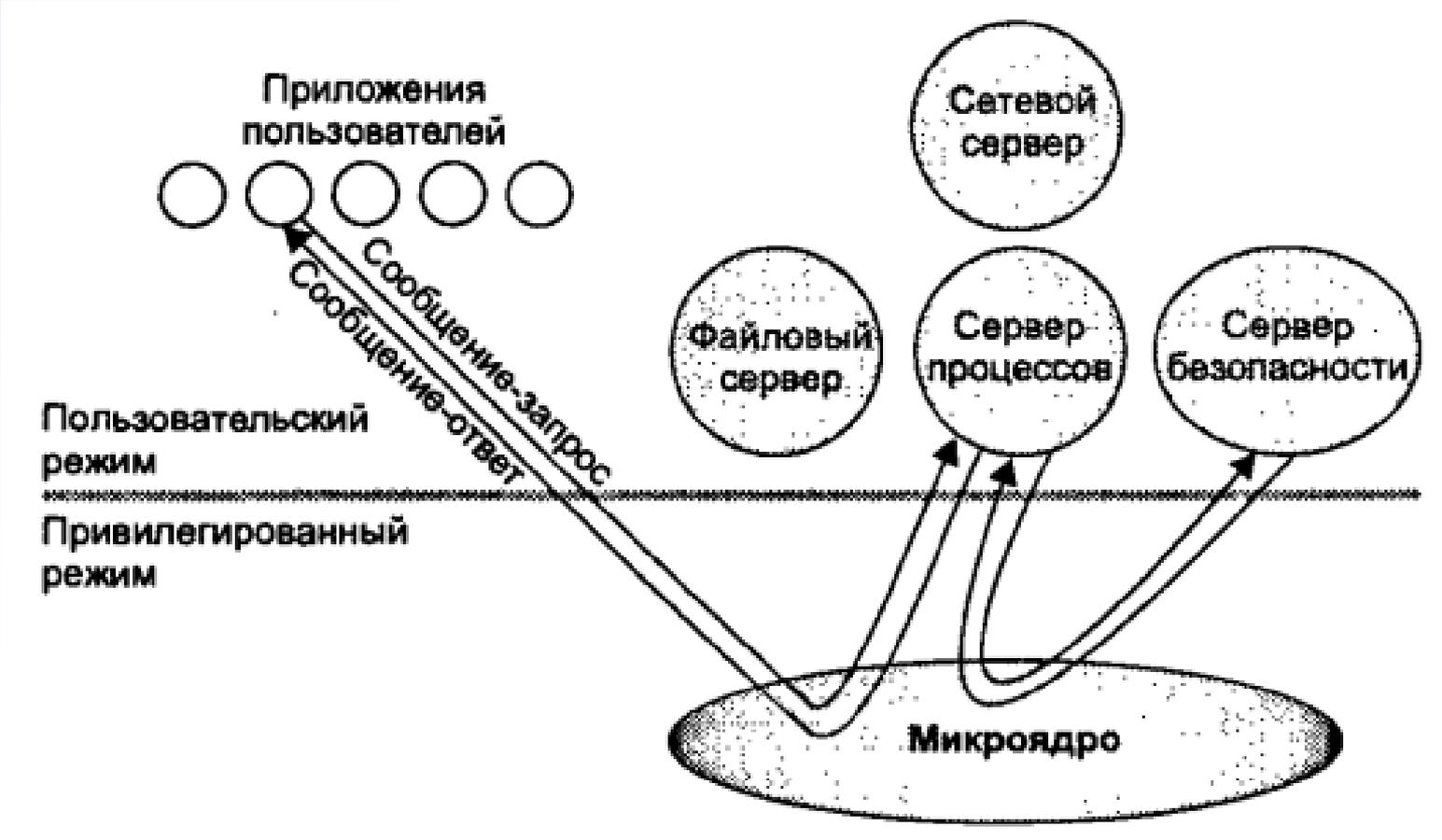


Аппаратура

Особенности микроядерной архитектуры

- Микроядро выполняется в привилегированном режиме.
- В микроядро входят машинно-зависимые модули и базовые функции ядра. Более высокоуровневые функции являются приложениями.
- Работа микроядерной ОС основана на модели клиент-сервер, при этом микроядро – транспортное средство между клиентскими и серверными приложениями (файловый, сетевой серверы, драйверы).

Принцип работы микроядра



Преимущества и недостатки микроядерной архитектуры

- Высокая степень переносимости, расширяемости.
- Более высокая надёжность, так как исключено влияние различных серверов друг на друга.
- Приспособленность для распределённых вычислений.
- Падение производительности из-за более частого переключения между привилегированным (микроядро) и пользовательским (клиентские и серверные приложения) режимами.

Гибридное ядро

Приложения

Менеджеры ресурсов
(ФС, сетевые протоколы)

Пользовательский режим

Привилегированный режим

API

Менеджеры ресурсов (драйверы устройств)

Базовые механизмы ядра

Машинно-зависимые компоненты ОС

Средства аппаратной поддержки ОС

Аппаратура

Экзоядро

- Экзоядро – архитектура ядра ОС, предоставляющая лишь функции для межпроцессного взаимодействия и безопасного выделения и освобождения ресурсов (в т.ч. контроля доступа).
- Машинно-зависимые компоненты выносятся в библиотеку пользовательского уровня.
- Обеспечивает большую эффективность по сравнению с микроядром за счёт отсутствия необходимости в переключении между процессами при каждом обращении к оборудованию.

Наноядро

- Наноядро – архитектура ядра ОС, в рамках которой ядро выполняет только задачу обработки аппаратных прерываний, генерируемых устройствами компьютера. После обработки прерываний наноядро посылает информацию о результатах обработки вышележащему программному обеспечению при помощи того же механизма прерываний.
- Может использоваться для обеспечения переносимости ОС; в качестве гипервизора.

Гипервизор

- Гипервизор – программа или аппаратная схема, обеспечивающая одновременное, параллельное выполнение нескольких ОС на одном компьютере.
- Гипервизор обеспечивает изоляцию ОС друг от друга, защиту и безопасность, разделение ресурсов между различными запущенными ОС и управление ресурсами.
- Гипервизор может предоставлять работающим под его управлением на одном компьютере ОС средства связи и взаимодействия между собой.

Требования к ОС

- Расширяемость – возможность вносить дополнения и изменения в структуру ОС без нарушения целостности системы.
- Переносимость – возможность работы ОС на различной аппаратной конфигурации.
- Совместимость – возможность запуска в ОС приложения, написанного для другой ОС.

Требования к ОС

- Надёжность и отказоустойчивость – наличие защиты от внутренних и внешних ошибок, сбоев. Поддержка аппаратных средств обеспечения отказоустойчивости.
- Безопасность – наличие защиты ресурсов ОС от НСД, включая данные, хранимые локально и передаваемые по сети.
- Производительность – возможность использования ресурсов ОС с наибольшей эффективностью.

Рассмотренные вопросы

- Подходы к управлению процессорами.
- Критерии эффективности работы ОС и классификация ОС на основе этих критериев.
- Многослойная структура ядра ОС.
- Типы ядра ОС.

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