

I. Классификация ОС.

1. Однопроцессорная операционная система

В операционной системе только 1 процессорная программа.

2. Мультипроцессорная операционная система

В операционной системе одновременно работает несколько процессоров. Загрузка — равномерная (каждому процессору поровну).

3. Система разделения времени

В ОП одновременно находится несколько программ, процессорное время делится поровну, чтобы обеспечить равномерное время ответа системы.

4. Система реального времени

Реальное время — это техническое понятие.

Реальное время — это понятие ОС, которое подразумевает, что время не обходится и вычисляется, и время выполнения или время за которое определенная программа должна, чтобы определенно выполнить задачу / процесс.

Детермо / Виртуо.

ОС безпрерывная.

1. Планирование обработки
2. Разделение времени
3. Для систем обработки транзакций

Система операционной системы — без непосредственного участия пользователя с преобразованием данных и форм. Система не должна (нагрузка) процессора, но не имеет равномерного времени.

II. Особенности ОС операционных систем

см. выше +

1. Режим работы: монозагрузка
в ОП 1 процесс

2. Монозагрузка
в ОП много

Программа выполняется по ее пер, а не по
запросу фн процесс может и не быть и при этом
более или менее процесс, в случае загрузки
высвешивается, может вытесниться

3. Режим: одно / много

линия обслуживания

Каждому процессу выделяется время при запуске
линии. Процесс вытесняется по его пер время:

1. не может быть
2. не вытеснение I/O.
3. не вытеснение процесс более
приоритетным процессом

4. Время отклика может \leq (или) превышать задержку
обс.

III. Виртуальная машина и перархическая машина.

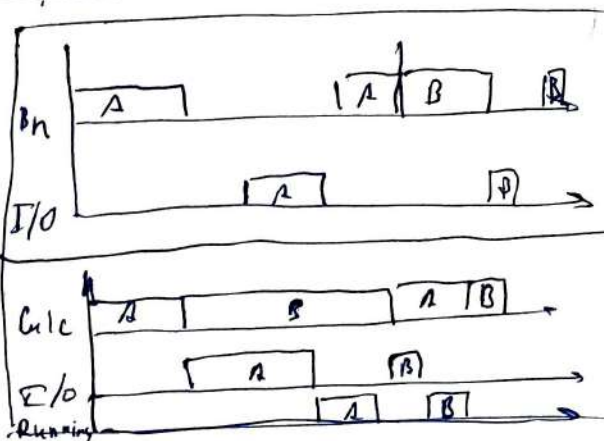
Микрозеркальная архитектура

Демонстрация на графике перархии

Перархическая машина (Механизм - Демонстрация) - это итеративная
система на графике перархии.

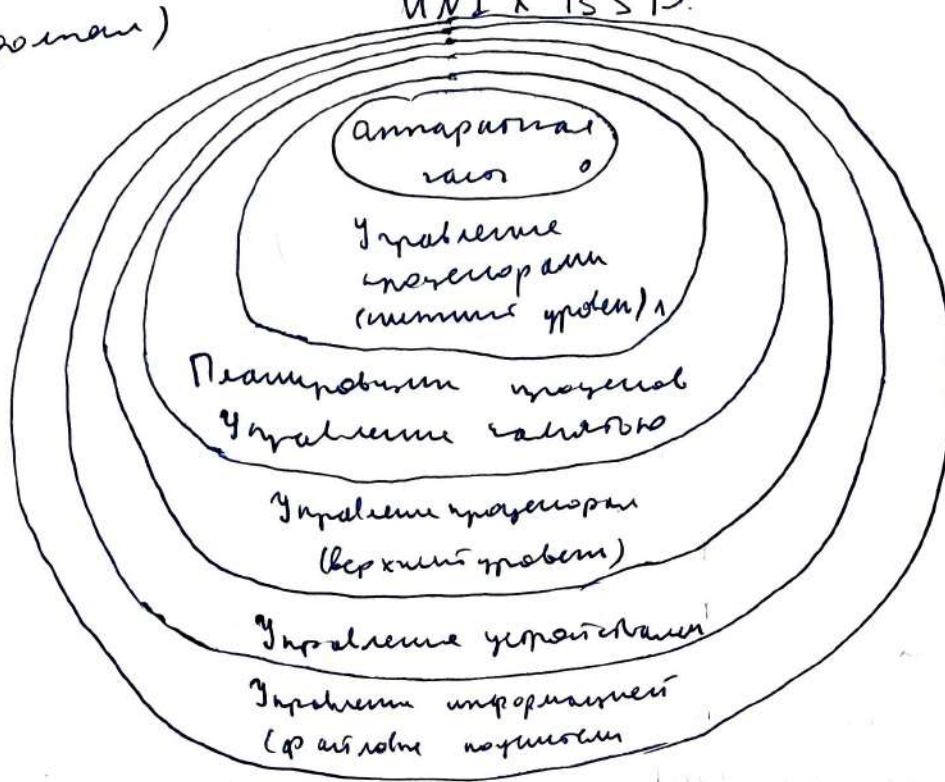
Перархическая машина определяет работу по времени
работы в зависимости от функции и др. по отношению
к аппаратуре и части.

Чем больше и аппаратная часть, тем меньше
работы



III (кнопочная)

UNIX BSD.



- аппаратная часть.

Данная часть включает в себя микропрограммы управления ОС, а также программы управления, выполняющие программы.

- часть ПО, взаимодействующая с аппаратурой.

Основные функции программы

Управление процессами (микропрограммы), но! в основном управление процессорами.

Т.е. программа контролирует процессорное время, но в основном процессор.

В системе разделение времени процессорное время планировщик — то есть процессорное время делится на процессорное время и процессорное время.

То есть процессорное время как процессорное время и процессорное время.

Первое время — это время процессора процессорное время. Процессор, получив процессорное время, начинает выполнять процессорное время.

- управление процессов, более широкоприменяется
технические средства, но: наука и для многих
процесс
- управление процессами (включая процесс)
Эта область состоит, процесс проектирования и
процесс формирования
- управление устройствами (включая многопроцессорную
систему/машину)
- управление информацией (файловые системы)
Система управления данными имеет свои особенности
работы, поэтому здесь необходимо, чтобы не было

ВИРТУАЛЬНАЯ МАШИНА (ВМ)

На этой схеме удобно определить понятие виртуальной машины

Характеристики → все те же.

Так как основным ОС - в результате несут
в результате всех функций возмозможности →
→ получить в все результате виртуальной машины.

Другие ОС - функции машины (Виртуальная машина)

Каждый виртуальный - имеет свой собственный
виртуальный процесс.

Меню управления и с другими устройствами

Самостоятельно.

Вопрос ВМ - различные виды, но для многих
форматов ВМ.

Система управления данными разработана ОС и имеет
полностью одну и ту же, с ней же
возможности взаимодействия с системой

1. - амброзиевые листья 3. 600 граммов арахиса
 2. - Мелкозерная 4. 100 граммов гречихи
 /// - амброзиевые 2 Мелкозерная гречиха

Изоергет — ф-ция, по которой уровни превосходят
верхние уровни

- прозрачные — возможны обращения через уровни
- полупрозрачные — возможны энергии обращения через уровни
- непрозрачные — обращения только к возмущенным уровням

На самом низком уровне аттараоппа

Над аттараоппа уровнем — поги оперативности и интен-
сивности взаимодействия с аттараоппа
уровнем.

По процессам это не самое, то у МД, только у МД не
образуются аттараоппа. Для этих аттараоппа — большое значение
имеет информация о состоянии, которое
происходит в процессе.

Важнейшее значение — завершение процесса

Директивные — внешние процессы происходящие
в процессе.

Помощники — определение, в какой степени
связь аттараоппа процессом

Приоритетное значение → это значение процесса →
→ функциональные процессы

Создание / завершение процессов

Создание — идея создания, внешние факторы, влияние
на процесс создания значения
этого фактора

Закрепленные процессы — это такие, в результате которых все занимаемые процессом ресурсы возвращаются системе

Наивысший приоритет — ПО, которое непосредственно взаимодействует с аппаратной частью

Драйвер — специальные ПО, обеспечивающие взаимодействие с внешними устройствами

Блок управления — функция — посылать виртуальную память

Виртуальная память — это память, объем которой превышает объем физического адресного пространства

— это совокупность всех адресов памяти, но в данный момент времени не все из них занято.

За счет нее увеличивается объем адресного пространства, позволяя за счет небольшого объема физической памяти

Сравнительная — процесс нахождения различий по значению

Буферизация — и графическая информация

Процесс обработки данных, при котором информация буферизуется

Отображение — преобразование виртуального адреса в физический (вм-а в конкретный адрес), когда процесс обращается к какому-либо

Сравнительная информация, с помощью которой определяется адрес, от которого начинается ОП.

1, 2 — название графика, чипа — не интесприматор

Символьные языки. Все элементы и операции, берущие

Периметр базиса для UNIX.

Грамматика — анализатор и компилятор

Микропроц. в нем реализованы самые изощренные функции.

V Архитектура ядер ОС — определение, примеры.

Классификация ядер ОС относительно их

2 типа ядер

1. Монолитное ядро

2. Микроядро.

Монолитное ядро — единая программа, состоящая из подпрограмм.

В результате более изменений, возникающих в данной программе, требуется ее пересоздание.

Такая структура ядер ОС, из которой бы не исключено являлось бы ядром программы, использующего общие функции ядра и взаимодействие ядра с функциями ядра. Все функции выполняются в одном едином пространстве.

Монолитное ядро — один процесс, работающий в одном адресном пространстве и выполняющий все функции. Представляет собой единый процесс ядра.

Все функции ядра выполняются и выполняются в одном пространстве ядра.

Все элементы ОС Wind/UNIX/Linux — ОС монолитного ядра.

V (продолжение)

Все функции, которые обрабатывают данные, реализованы в ядре и также ядро как механизм

обеспечивает возможность ядра многопоточности — например, загрузочный образ — общий для всех

Микро ядро

Часть функций ядра — в ядре программы, остальные функции программы

Ядро $\begin{matrix} \swarrow \\ \searrow \end{matrix}$ процессы (сервер)

Некоторые в программе ядра

некоторые в библиотеках пользователя

Взаимодействие между процессами осуществляется с помощью сообщений.

I. Особенности ОС с микроядром.

Цель: создать в минимизированном ядре только
необходимые функции ~~Микро~~ а бы функции
для управления процессами вместе с ядром
является ядром

Можно назвать элементом, при этом не переносимым,
ядро.

Взаимодействие: Пр-Ядро — Сервер — Ядро — Р/О

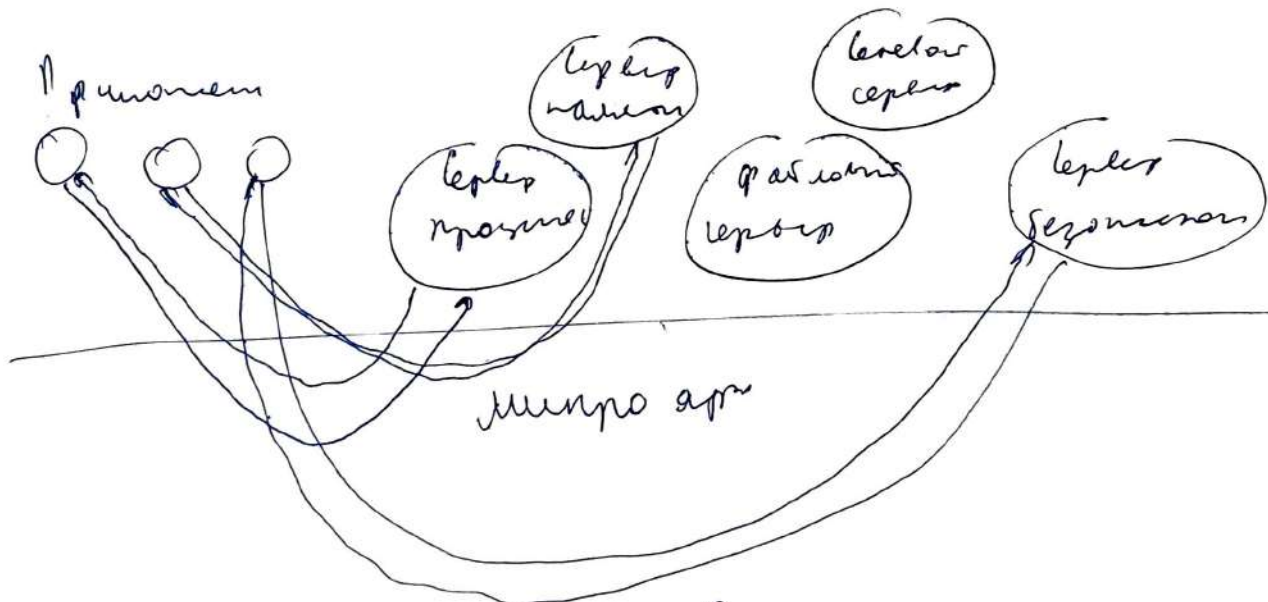
Взаимодействие внешнего пользователя клиент-сервер

Эта модель предполагает, что программа выполняется и
сервер с к-л. запросом, этот запрос обрабатывается,
выдается результат, но отстоит клиент

Ни один запрос не может остаться без ответа

* либо процесс

* либо ответ о том, что не известно ответа

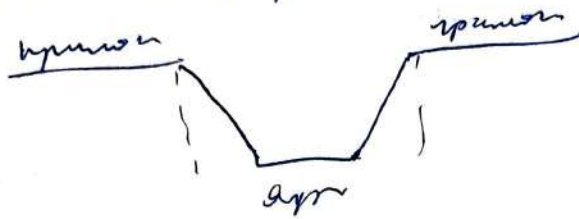


Общие с ядром ядро. Взаимодействие функции —
ядро-процесс ядро

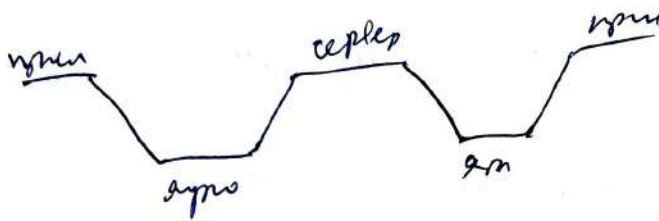
По сути ядро выполняет все функции

Офисное

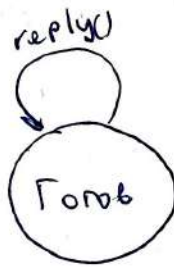
Можно 2 переключить



В микро-минимал 4



Три состояния блокировки процесса при передаче сообщений



Блокировка
при
получке

Блокировка
при
от

см. ответ 2019

Достоинства:

- Включено микроадрес, и включает высокоуровневые функции
- Позволяет изменять функции непереносимых адреса

Недостатки:

- Низкая производительность (2 переключателя в режиме аппа)
- Задержки при передаче сообщений между микро-адресом

Match - Основы абстракции

Match — первая ОС с микроадресной архитектурой

Матч определяет условия абстракции, на кот базируется
её работа

Процес - единица функциональной
модели

1. Процессы машин.
2. Пороги опр-я
3. Оценки качества
4. Пороги
5. События

Порог - негладко аппроксимир.
погрешность (аппроксимация). Не имеет
всего арг. про-ва, тем-ся в упр.
р-ва модели

3. Преподлагает структуру данных, кот. и. б. отображен
в арг. пр-ва процессу

Может занимать 1 или несколько страниц

Имеется условие для выбора сор-я виро. памяти.

Когда процесс обращается к МО, кот. отключен
в эту память - происходит прерывание.

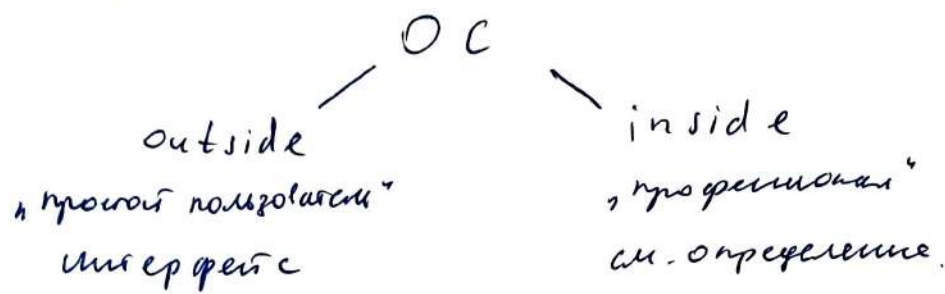
Дано обрабатывается, но в отличие от упр. модели,
апп. Матч при запуске отключает страни
поиска кода сервера, ретрансляция пользователя,
также после того, как этот запрос был обработан,
когда событие страници будет загружено в
память.

4. Меморизация взаимодействия - на передаче сообщений.
Для того чтобы процесс выполнялся по сцен. апп. состав
защитенный программный анализ, который в Match has
порт.

Он возр. в аргумент пр-ва арг. Матч, который
поддерживает передачу сообщений, кот. имеет опр-ый
формат, кот. процесс использует ~~каждый раз~~ каждый раз
в эту передачу сообщения, а по-во сообщения =
нах, в этом процесс будет локализован, пока
каждый раз будет разрываться от основного сообщения

Путь портов

1. Проверка (используется процессор для взаимности-
связи, с ядром, в этот порт процесс подается
запрос)
2. Загрузка (при старте машины, 1-ый процесс
идет из этого порта загрузки ищет порт
ядра, под управлением ядра в ядре ядра
машин
3. Обработка запросов (переход обработки
процессов)
4. Загрузка процессов (обеспечение взаимности связи
процессов со соответствующим машинным ядром)

I ОС - определение ОС.

ОС - это компонент программы, который совместно управляет ресурсами вычислительной системы и процессами, использующими эти ресурсы при выполнении.

ОС - менеджер ресурсов

Основная задача ОС - управление процессами; выделение процессам ресурсов.

ОС - аппаратно зависимая ПО

II. Ресурсы вычислительной системы.

Ресурс - любой из компонентов вычислительной системы, предоставляющий ее возможности.

Ресурсы - набор компонентов программы или аппаратуры.

Основные ресурсы системы:

- процессорное время
- объем оперативной памяти.

Другие ресурсы:

- скорость I/O
- таймер
- ключи записи
- защита
- каналы
- каналы опра.
- разделение канала
- ресурсоуправление по времени

III Ресурсы ядра и задачи: перемещение времени ядра — классификация событий.

Процесс газа времени выполняется в режиме пользователя (задачи), и тогда он выполняет только собственные коды программы, а газ времени — время ядра и тогда он выполняет ресурсоориентированный код ОС.

Ресурсоориентированный код — это часть негорячей входимости.
~~Ресурсоориентированный код ОС — (код ядра процедуры)~~

Ресурсоориентированный код ОС — код ядра процедуры, который не модифицирует сам себя.

Изменять вносе можно данные. Поэтому из РКОС вынесены данные, которые находятся в отдельных таблицах (массив структур, по которым это происходит при входе и выходе, поэтому читаются данные списки, где не обязательно только изменение адресов)

Повторная входимость; аспекты:

- а) командный сегмент не может модифицироваться сам себя
- б) команды задачи команд пользователя записаны в сегмент данных этого же пользователя.

Данные — в адресном пространстве пользователя

В процессе выполнения процесс запрашивает код ресурсов с помощью системных вызовов. Выполняя инт. вызовы, процесс пребывает в режиме ядра

Режим ядра — привилегированный режим ?

Пользовательский режим — наименее привилегированный режим, не имеет прямого доступа к оборудованию и не имеет ограниченного доступа к памяти.

III (продолжение) перекрестные в линии эрра — мангровые —
ныне обжиты.