

Модели разработки прог. систем

1. Каскадная

(+) детерминированность времени и затрат

(-) возможное устаревание к моменту реализации (создание системы, которую сразу необх. Дораб.)

2. Каскадно-итерационная

(+) максимальное удовлетворение требований заказчика

(-) недетерминированность времени и затрат

3. Спиральная

(+) детерминированность выполнения каждого этапа, возм-ть рассм-я системы до завершения

(-) недетерминированность времени и затрат на конеч. Продукт

Виртуальная машина или виртуальный компьютер - программное расширение возможностей реального компьютера.

Использование ТЕГ в ячейках памяти: **Контроль целостности данных, Контроль доступа к командам/данным, Контроль доступа к машинным типам данных (фиксация кода типа данных).**

Расслоение памяти

Преимущества: параллельное считывание из разных банков, проявляется преимущество при работе с кэшированием,

Кэш-память (процессор - ОП)

- Сокращается количество обращений к ОЗУ
- Существенно увеличивается скорость доступа к памяти в случае использования ОЗУ с «расслоением»
- Усложнение логики процессора

Аппарат прерываний - аппарат, который позволяет организовывать стандартную обработку всех прерываний, возникающих при функционировании вычислительной системы.

Проблемы, возникающие при исполнении программ

Вложенные обращения к подпрограммам, Накладные расходы при смене обрабатываемой программы, Перемещаемость программы по ОЗУ, Фрагментация памяти

Регистровое окно — это набор регистров, по количеству меньший числа физ. Регистров процессора, который в каждый момент времени доступен для программы пользователя.

Недостатки аппаратного стека: Поскольку стек располагается в оперативной памяти, то при каждой обработке прерывания процессору придется обращаться к оперативной памяти, что сильно снижает производительность системы при частых возникновениях прерываний.

Страничной организации памяти

Преимущества:

Решаются все проблемы, связанные с исполнением программ (см. выше)

нет необходимости держать в оперативной памяти весь исполняемый процесс (можно держать только несколько страниц процесса, остальное может находится на ВЗУ - отквачено)

Недостатки:

страничная фрагментация, или **внутренняя** (скрытая) **фрагментация**: если в странице используется хотя бы один байт, то вся страница отводится процессу

если таблица страниц целиком располагается на регистровой памяти, то в силу дороговизны последней размеры подобной таблицы будут слишком малы
проблема при смене обрабатываемой программы

SMP

(+) Простота реализации

(-) ограничения на количество подключаемых процессоров

(-) Проблема синхронизации КЭШа (нужен кэш с отслеживанием)

NUMA

- (+) Степень параллелизма выше, чем в SMP
- (-) Централизация (ограничение ресурсом шины)
- (-) Использование когерентных кэшей загружает шину служебной информацией

ccNUMA

- (+) позволяет строить системы из сотен процессорных элементов
- (-) сложность
- (-) Загрузка общей шины служебной информацией

MPP

- (-) дороговизна
- (+) Высокая эффективность при решении определённого класса задач
- (-) Узкая специализация

COW

- (+) «прозрачность» архитектуры
- (+) «относительная «универсальность» - возможность применения для решения широкого круга задач
- (-) топология (взаимодействие большого числа элементов)

Выделенный канал

- (+) качество связи
- (-) браконьерство
- (-) деградация телефонной сети
- (-) снижение возможности коммутации

Коммутируемый канал

- (-) недетерминированное качество связи
- (+) не держим ресурс

Сеть коммутации каналов

Преимущества:

После установления соединения сеть находится в состоянии готовности
Требования к коммуникационному оборудованию минимальны
Минимизируются накладные расходы по передаче данных
Детерминированная пропускная способность

Недостатки:

Требование избыточности сети
Период ожидания соединения (канала) недетерминирован
Неэффективное использование выделенного канала
В случае сбоя или отказа повторная передача информации

Сеть коммутации сообщений

- (+) Отсутствие занятости канала на недетерминированный промежуток времени
- (-) Сообщения могут быть произвольного размера - необходимость наличия средств буферизации неопределённых характеристик
- (-) Необходимость в специализированном коммуникационном оборудовании и ПО
- (-) Повтор передачи всего сообщения в случае сбоя

Сеть коммутации пакетов

- (+) Так как известна топология сети и характеристики её элементов, то возможно определение требований в коммутационных узлах ⇒ возможна оценка размера буфера и времени доставки пакетов
- (-) Увеличение трафика из-за наличия заголовочной информации
- (-) Проблема сортировки пакетов

TCP обеспечивает надежную доставку данных с обнаружением и исправлением ошибок и с установлением логического соединения.

UDP отправляет пакеты с данными, не контролируя их доставку.

ОС с монолитным ядром

- (+) для конкретного состава функциональности и логики ядра это будет наиболее эффективное решение
- (-) тяжело вносить изменения в ядро

Многослойные ОС

- (+) простая модификация
- (-) избыточность информации по системе
- (-) замена или модернизация «один в один»

Микроядерная архитектура

- (+) хорошо расширяема
- (+) подходит для многопроцессорных систем
- (-) накладные расходы (из-за разбиения запроса на мн-во запросов)
- (+) нет существенных ограничений на кол-во драйверов

Файл как последовательность записей переменной или постоянной длины

- (-) внутренняя фрагментация
- (-) сложность вставки или удаления из середины файла

Иерархическая структура файла

- (+) удобное редактирование
- (-) сложная реализация

Атрибуты файла – параметры, характеризующие свойства и состояния файла.

Одноуровневая файловая система

- (+) простота и удобство доступа
- (-) коллизии имен
- (-) не предполагает многопольз. Работы

Двухуровневая файловая система

- (+) предполагает работу нескольких пользователей
- (-) коллизия имен для одного пользователя
- (-)неудобно хранить все файлы одного пользователя в одном месте

модель непрерывных файлов

- (-) внутренняя фрагментация
- (-) фрагментация между файлами
- (+)отсутствие фрагментации файла по диску
- (-)фрагментация своб. Пр-ва
- (+)простота и высокая произв-ть
- (-) увеличение размера существ. Файла

Файлы, имеющие организацию связанного списка

Достоинства

- Отсутствие фрагментации свободного пространства (за исключением блочной фрагментации)
- Простота реализации
- Эффективный последовательный доступ

Недостатки

- Сложность (неэффективность) организации прямого доступа
- Фрагментация файла по диску
- Наличие ссылки в блоке файла (ситуации чтения 2-х блоков при необходимости чтения данных объемом один блок)

FAT

- (+) Возможность использования всего блока для хранения данных файла
- (+) Оптимизация прямого доступа (при полном или частичном размещении таблицы в ОЗУ)
- (-) Желательно размещение всей таблицы в ОЗУ

Индексные дескрипторы

- (+) Нет необходимости в размещении в ОЗУ информации всей FAT всех файлов системы, в памяти размещаются атрибуты, связанные только с открытыми файлами
- (-) Размер файла и размер индексного узла

sfs

- (-) Концентрация важной информации в суперблоке
- (-) Проблема надежности
- (-) Фрагментация файла по диску
- (-) Ограничения на возможную длину имени файла
- (+) Оптимизация в работе со списками номеров свободных индексных дескрипторов и блоков
- (+) Организация косвенной адресации блоков файлов

Одиночное непрерывное распределение ОП

- (+) простота (миним. Требования к аппаратуре)
- (-) Часть памяти не используется
- (-) Процессом/заданием память занимается все время выполнения
- (-) Ограничение на размеры задания

Распределение перемещаемыми разделами

- (+) Простое средство организации мультипрограммирования

(+)Простые средства аппаратной поддержки

(+)Простые алгоритмы

(-)Фрагментация внутри разделов

(-)Ограничение предельного размера прикладных процессов размером максимального физического раздела ОЗУ

(-)Весь процесс размещается в памяти — возможно неэффективное использование

Распределение перемещаемыми разделами

(+)Ликвидация фрагментации

(-)Ограничение размером физической памяти

(-)Затраты на перекомпоновку (на компрессию)

Страничное распределение (осн. Недостаток – фрагментация внутри страниц)

TLB

(-)связан с большим размером таблицы страниц

(-) изменение таблицы при смене контекстов

(-) проблема при организации мультипроцессорирования

(+) прямая адресация (вирт -> физ стр.)

Хэш-таблицы

(+) обход проблемы большого размера таблицы страниц

(-) перемещение списка коллизий

Инvertированная таблица страниц(PID, VP, offset)

(+)не нужно перегружать таблицу при смене контекстов

(-) небыстрый поиск по таблице

Алгоритмы откочки страниц на ВЗУ

FIFO

(-) возможна откочка интенсивно используемой страницы

(+) простота реализации

LRU (матрица)

(+) грамотно учитывает интенсивность использования

(-)сложная реализация

NFU (Count[i]+R[i])

(-) помнит «старую активность»

(+)при большой активности, возможно переполнение счетчика

(+)грамотный учет активности использования страницы

Вообще про страничное распределение

(+) хорошая виртуализация

(+) решена проблема фрагментации пр-ва

(+) защита памяти

(+) прога не д.б. вся в памяти

(-) сложность организации

(-) адресное пр-во процесса неоднородно (нет поддержки логич. структур)

Сегментное распределение

(+) простота организации

(-) размещение целиком процесса в памяти

(-) фрагментация

Сегментно-страничное распределение

(+) возможно переключаться между страничной и сегментной организацией

Кэширование обменов с внешними устройствами в UNIX

(+) уменьшается число реальных обращений к ВУ

(-) возможное несоответствие содержимого буфера и ВУ (например, если отключилось питание)