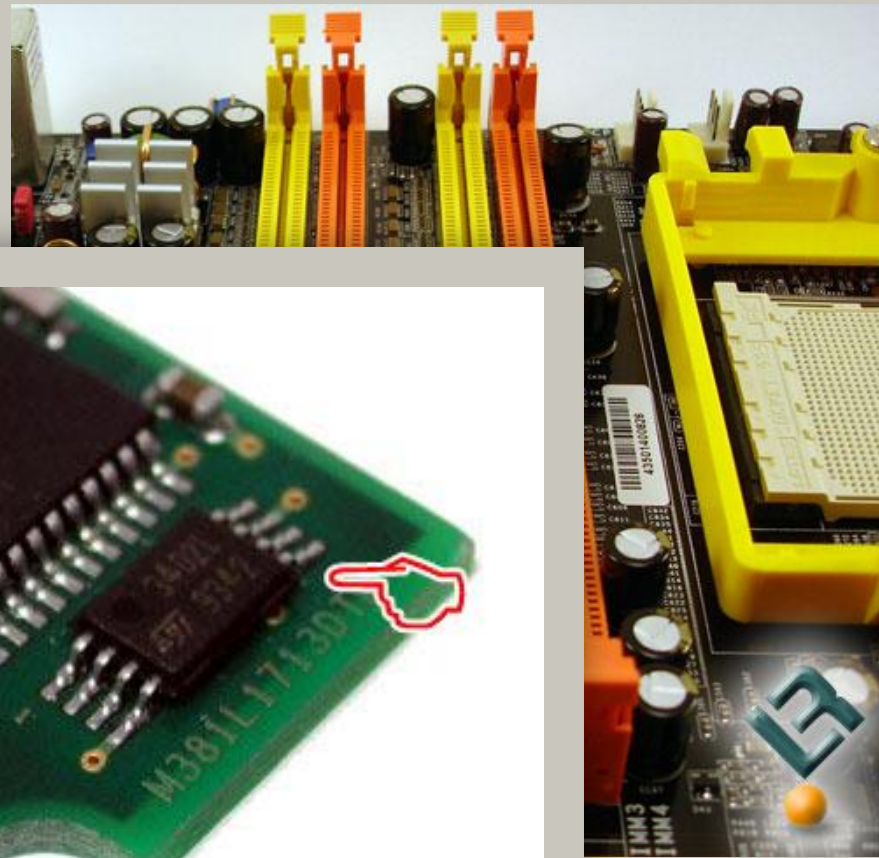


# УПРАВЛЕНИЕ ОПЕРАТИВНОЙ ПАМЯТЬЮ

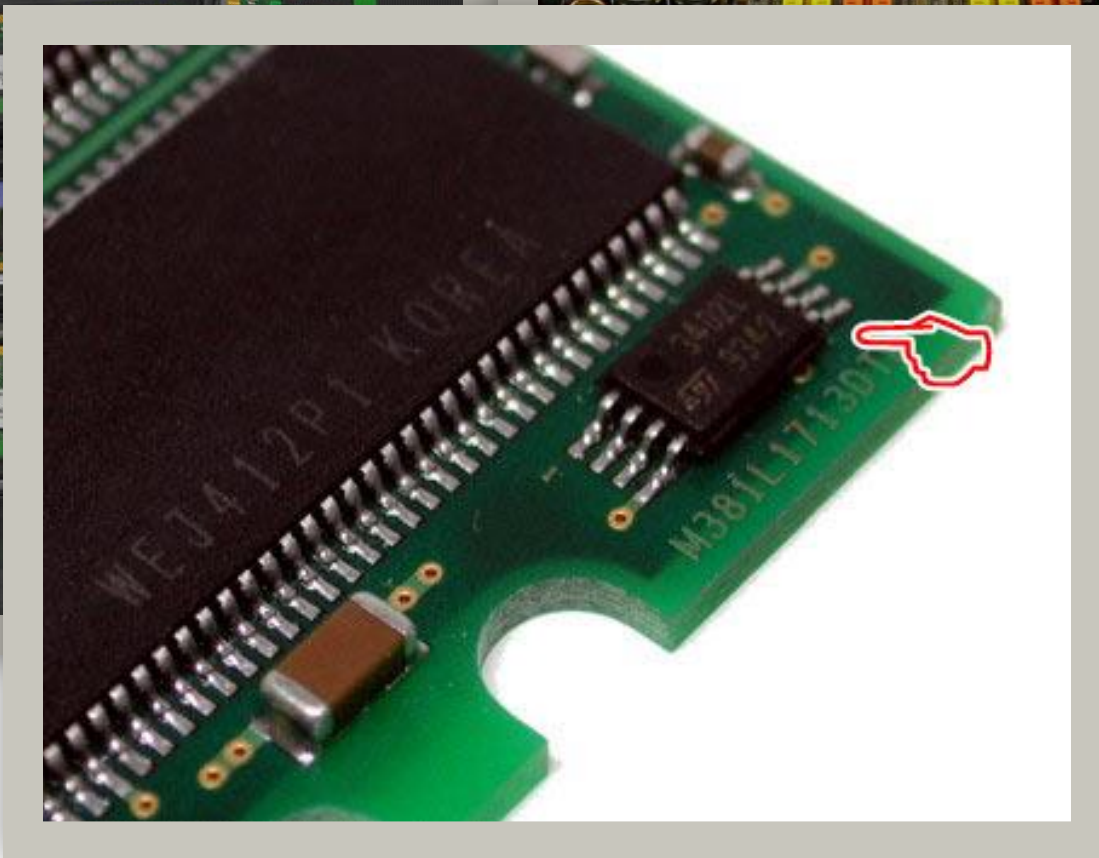
# КАК УСТРОЕНА ОПЕРАТИВНАЯ ПАМЯТЬ?



МОДУЛИ



ТИ



КОНТРОЛЕР SPD

МАРШРУТИЗАЦИЯ ЗАПРОСОВ НА ДОСТУП К ПАМЯТИ....

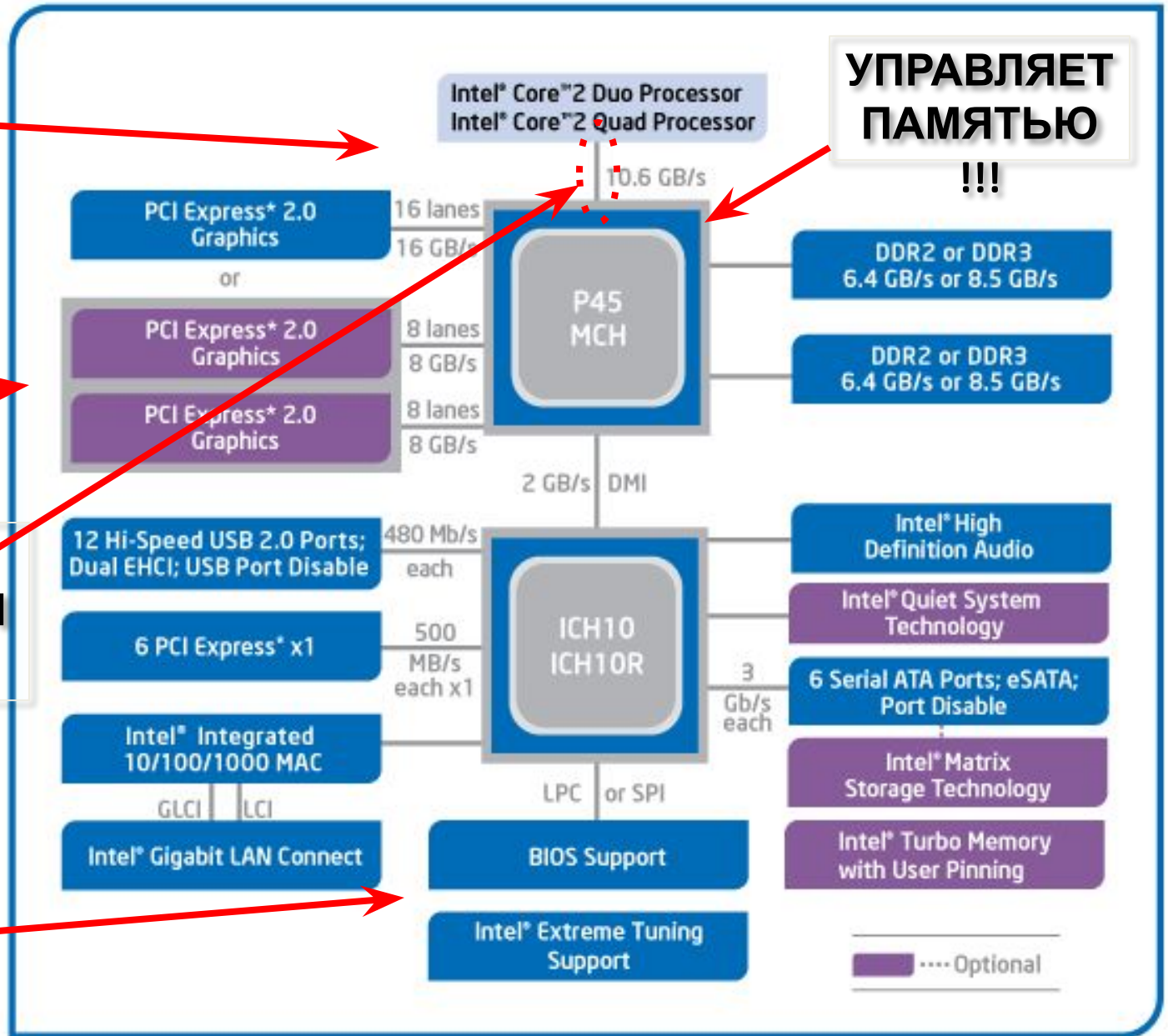
ПРОЦЕССОР  
«ПРОСИТ»  
ДОСТУП К  
ПАМЯТИ  
И

УСТРОЙСТВА ТОЖЕ 😊

ОПРЕДЕЛЯЕТ  
МАКСИМАЛЬНЫЙ  
РАЗМЕР

НЕКОТОРЫЕ  
«ПРИКИДЫ-  
ВАЮТСЯ»  
ПАМЯТЬЮ

УПРАВЛЯЕТ  
ПАМЯТЬЮ  
!!!



Intel® P45 Express Chipset Block Diagram



## КАК УЗНАТЬ СКОЛЬКО ФИЗИЧЕСКОЙ ПАМЯТИ ЕСТЬ?

Физическая память (КБ)	
Всего	1038188
Доступно	414720
Системный кэш	434260

Диспетчер задач Windows

1.33 ГГц, 0,99 ГБ ОЗУ  
Расширение физических

Свойства «Мой компьютер»  
Windows

Total physical memory: 261,582,848 bytes = 249.46 MB  
Free physical memory: 72,847,360 bytes = 69.47 MB

KinfoCenter в Linux

В общем случае – операционная система узнает о  
размере оперативной памяти и карте ее  
использования с помощью BIOS!!!

*Может ли объем оперативной памяти измениться «на лету»?*

Перед менеджером памяти операционной системы ставятся следующие задачи:

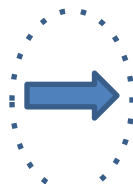
- *контроль за состоянием* каждой единицы памяти: свободны они или заняты (т.е. они распределены и используются каким-либо процессом);
- *распределение виртуальной памяти*, т.е. определение кому, когда и сколько памяти выделять;
- *выделение физической памяти*, т.е. определение конкретных областей памяти, которые будут выделены тому или иному процессу;
- *освобождение памяти* после полного завершения процесса или при его временной остановке.

# ПРЕДСТАВЛЕНИЕ ПАМЯТИ ДЛЯ ПРОЦЕССОРА...

```
#include <stdio.h>
int main(){
    int x[100];
}
```

## ЛОГИЧЕСКИ

Используется программами



## ЛИНЕЙНЫЙ

Определяет место в виртуальном адресном пространстве



## ФИЗИЧЕСКИ

Определяет место в оперативной памяти (аппаратный адрес)

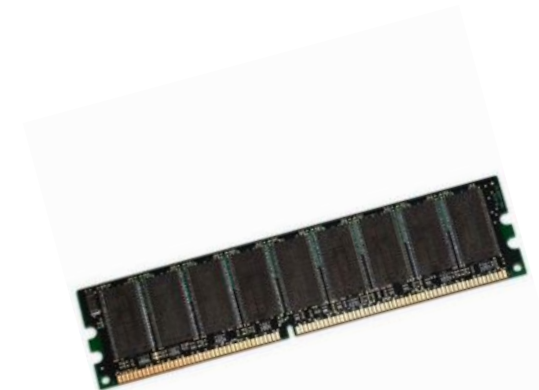


0x0

4 ГБайт

0xFFFFFFFF

```
0011001011001001010100001010001
0010111001010101011101100000110
11010100000011101010111010001110
10001110110110001010111000110011
```



Преобразование адресов, сегментация

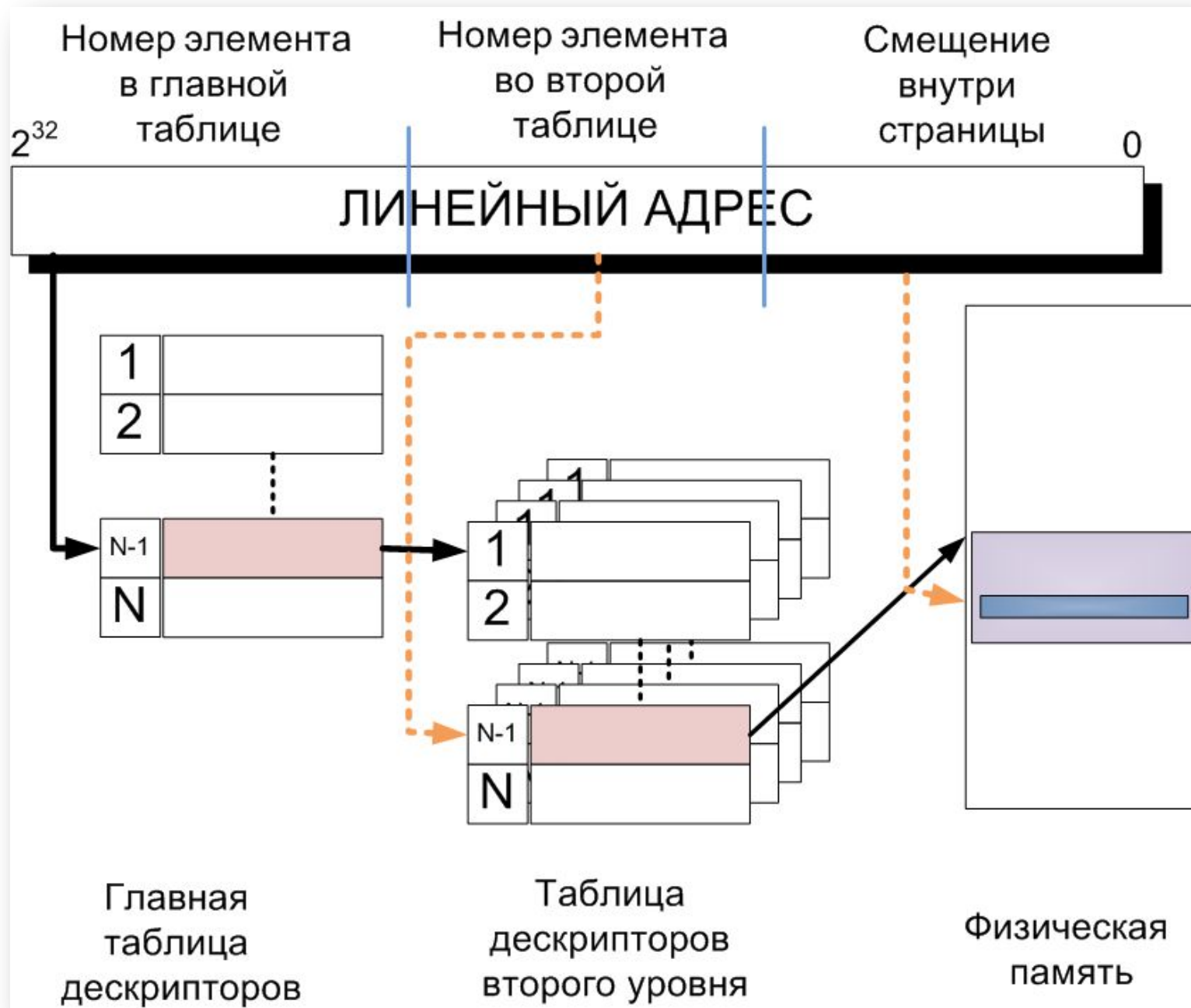
**ПРОБЛЕМА 1**

Страничное представление памяти, подкачка

**ПРОБЛЕМА 2**

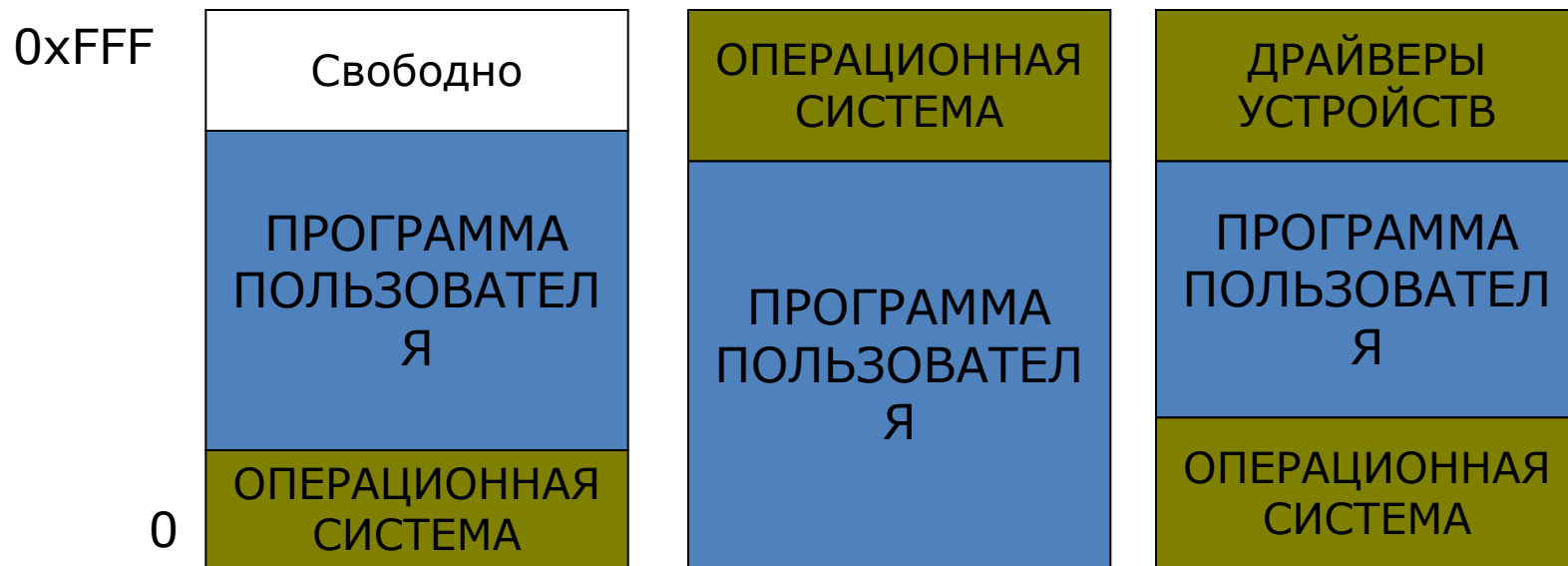
На аппаратном уровне все эти функции выполняет Memory Management Unit (MMU)

# СТРАНИЧНОЕ РАЗДЕЛЕНИЕ ПАМЯТИ



TLB?



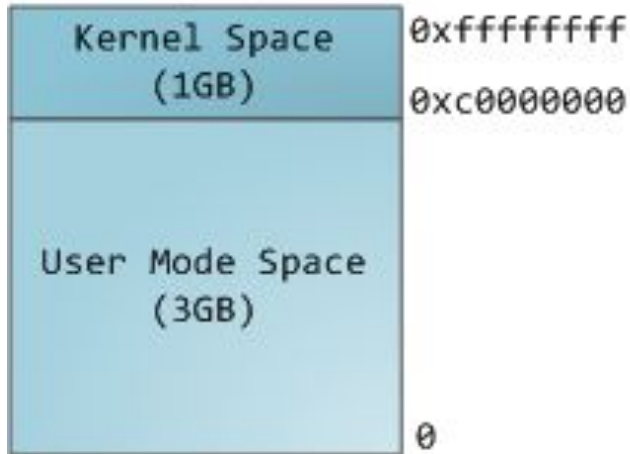


В памяти каждого процесса должно быть:

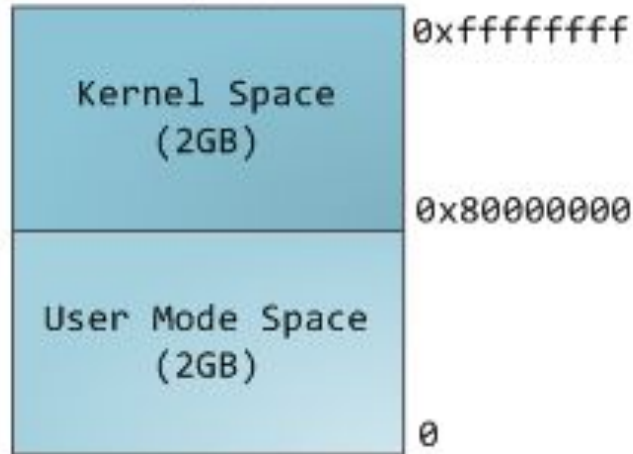
- операционная система
- драйверы устройств
- библиотеки функций
- выполняющаяся программа
- данные
- и т.п.

# ПРИМЕРЫ РАСПРЕДЕЛЕНИЯ АДРЕСНОГО ПРОСТРАНСТВА ПРОЦЕССОВ В ОПЕРАЦИОННЫХ СИСТЕМАХ LINUX и WINDOWS

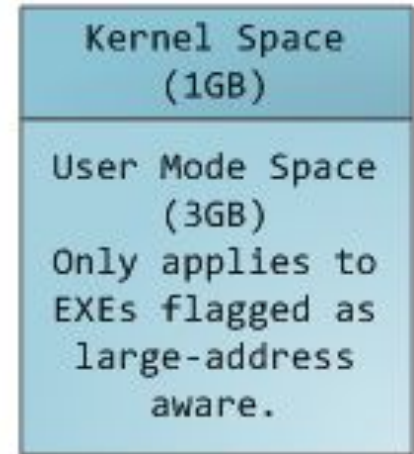
Linux User/Kernel  
Memory Split



Windows, default  
memory split



Windows booted  
with /3GB switch

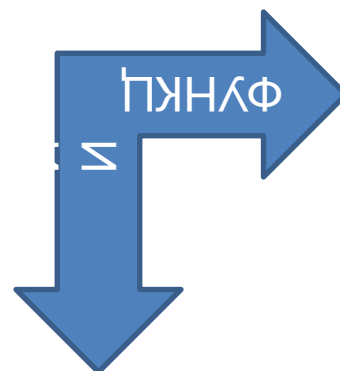


Адресное пространство процесса разделяется на две части (ОС и приложение)

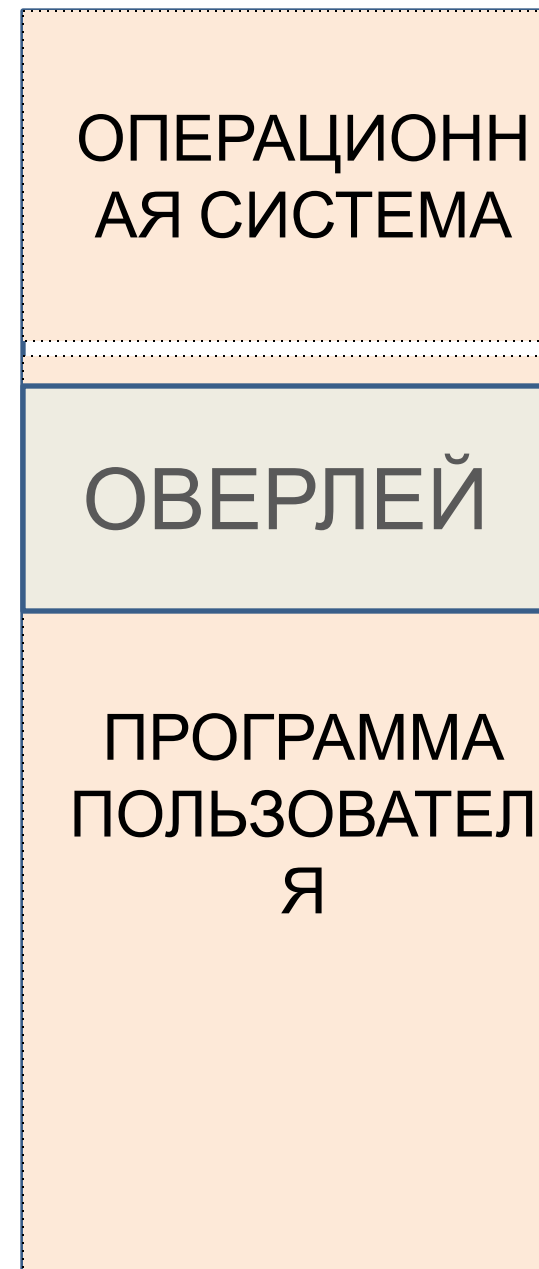
# Как быть, если оперативной памяти не хватает?

Используем функции по очереди и загружаем их при необходимости в память.

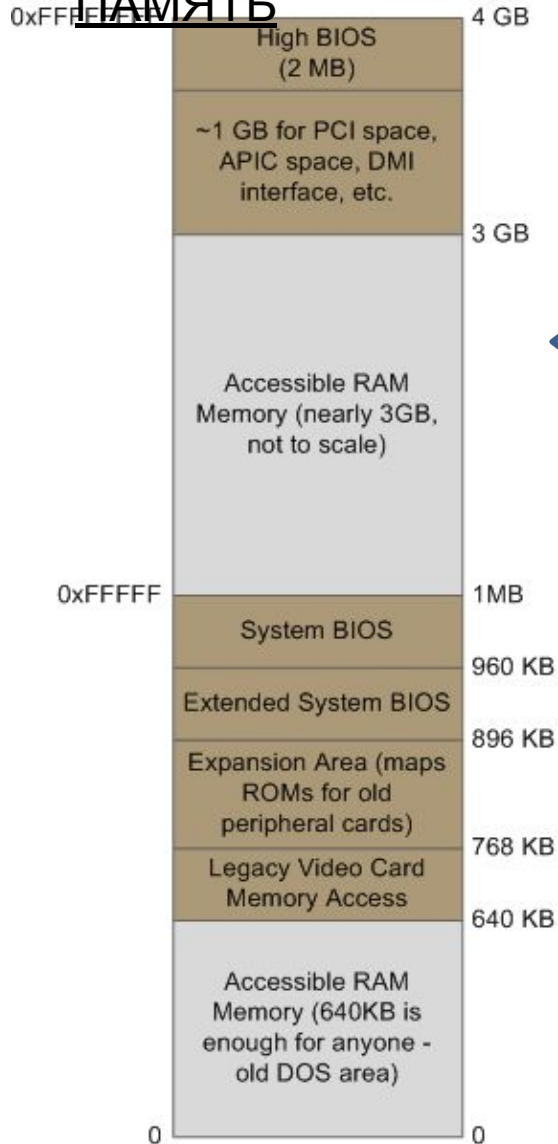
Такой подход стал родоначальником подкачки (свопинга), AWE, PAE...



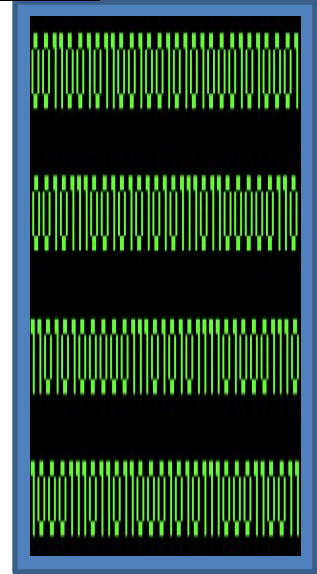
ОВЕРЛЕЙ



ВИРТУАЛЬНАЯ  
ПАМЯТЬ



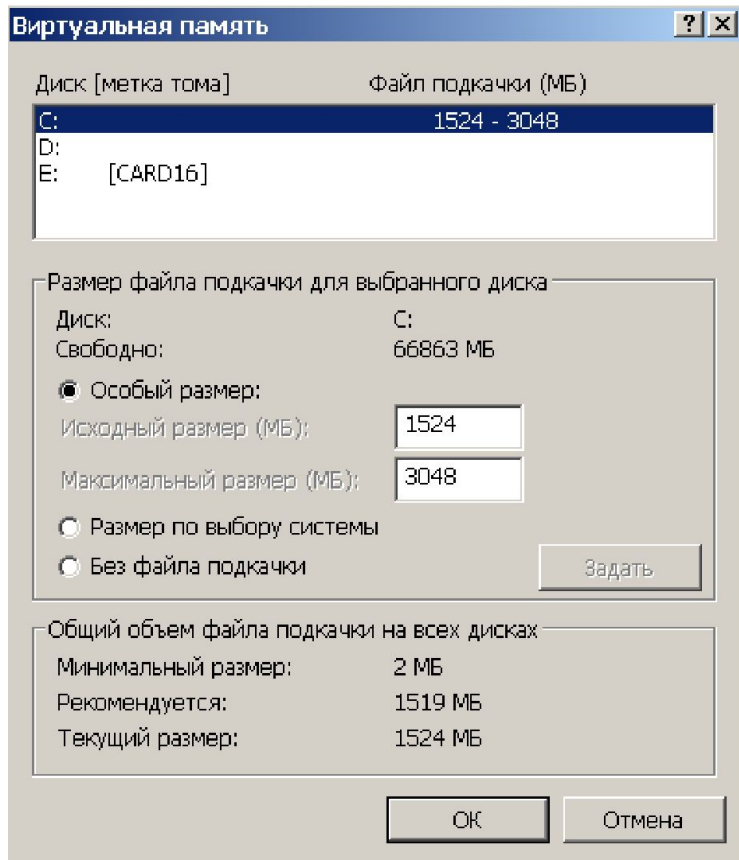
ФИЗИЧЕСКАЯ  
ПАМЯТЬ



ОБЛАСТЬ  
ПОДКАЧКИ

Страница?  
Прерывание  
page fault?

# НАСТРОЙКА ПОДКАЧКИ



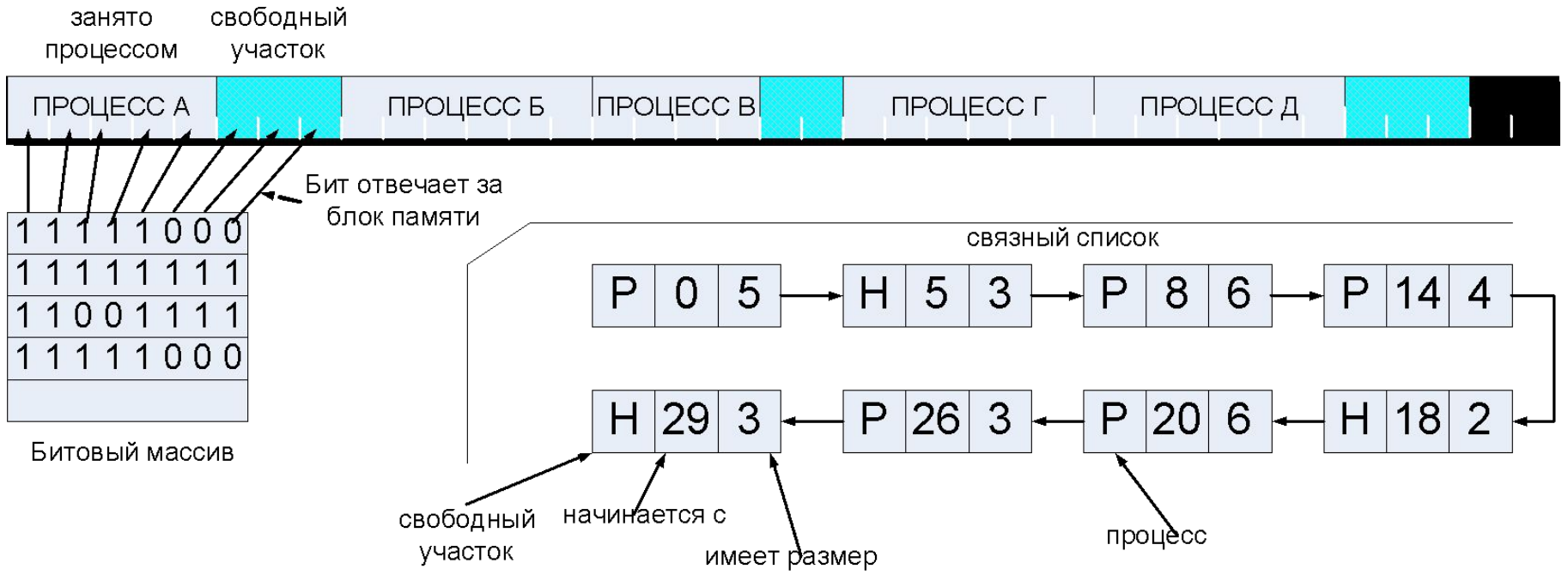
Настройка подкачки в MS  
Windows



В Linux – области свопинга задаются командой `swapon....` Пример – области свопинга в `/etc/fstab...`

демонстрация ...

# КАК ХРАНИТЬ ИНФОРМАЦИЮ О ВЫДЕЛЕНИИ ПАМЯТИ?

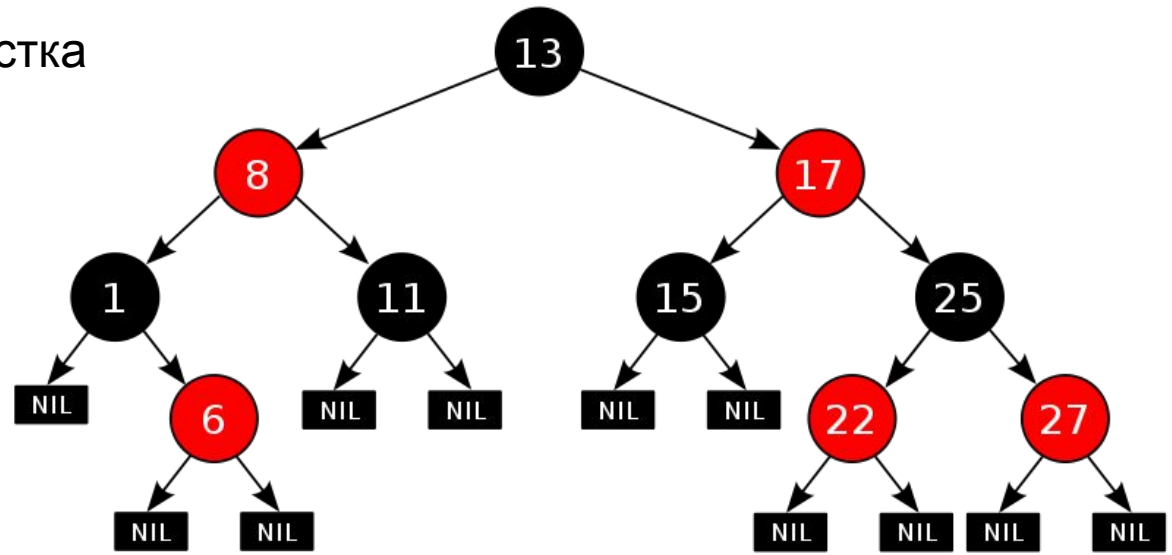


Проблема – поиск нужного участка памяти?

Сбалансированные AVL деревья....

Г.М. Адельсон-Вельского и Е.М. Ландиса

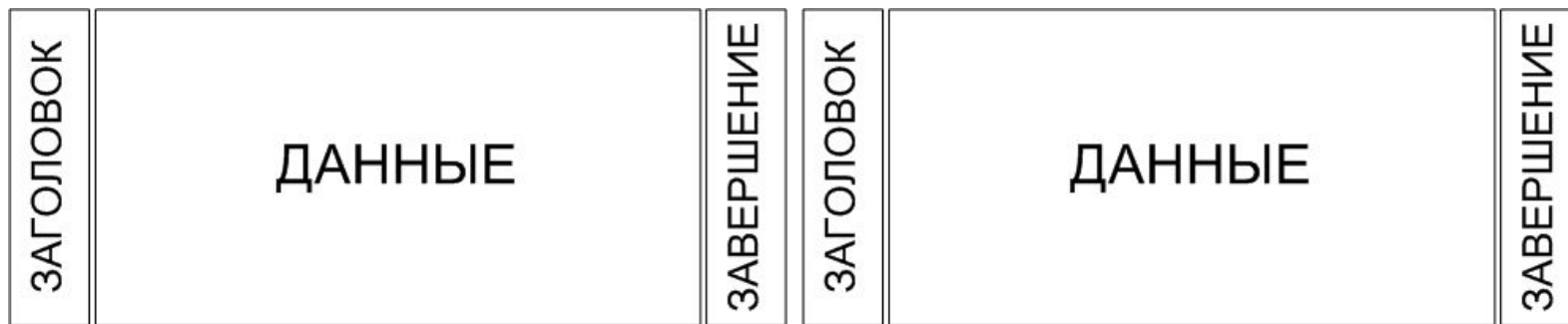
Фрагментация.....



- *первый подходящий*. Выделяется первый же подходящий блок памяти. Если он больше, чем требуется, то он делится на две части: выделенную и свободную.
- *следующий подходящий*. Модификация предыдущего. Каждый раз поиск начинается не с начала памяти, а с последнего найденного участка.
- *самый подходящий*. Ищется свободный участок памяти с максимально соответствующим размером.
- *самый неподходящий*. Ищется свободный участок, который наихудшим образом подходит запросу. После чего он делится на две части.
- *быстрый подходящий*. Информация обо всех свободных участках памяти помещается в ассоциативный контейнер исходя из их размера.
- *алгоритм близнецов (buddy)*. Память делится на участки разных размеров....

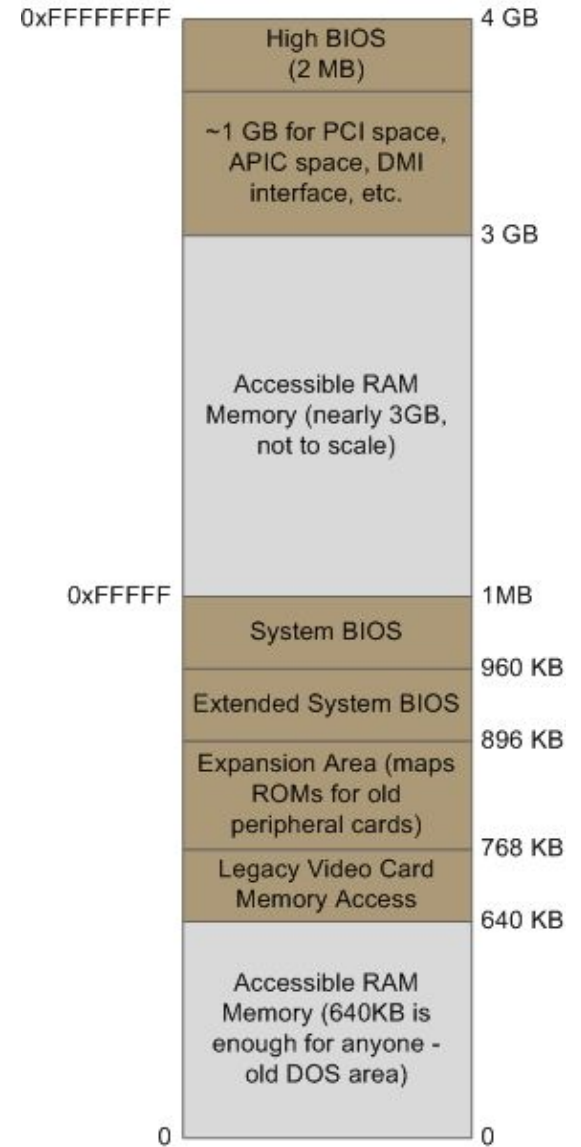
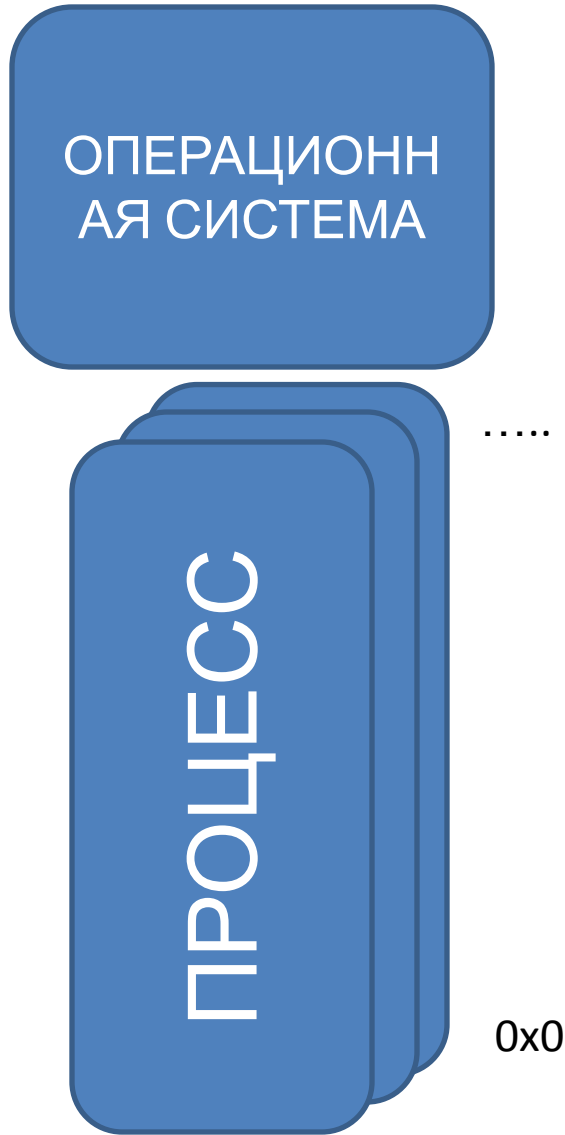
## Двухуровневое выделение памяти:

- выделение виртуального адресного пространства (ядро)
- организация использования памяти маленькими порциями (процесс, библиотека)

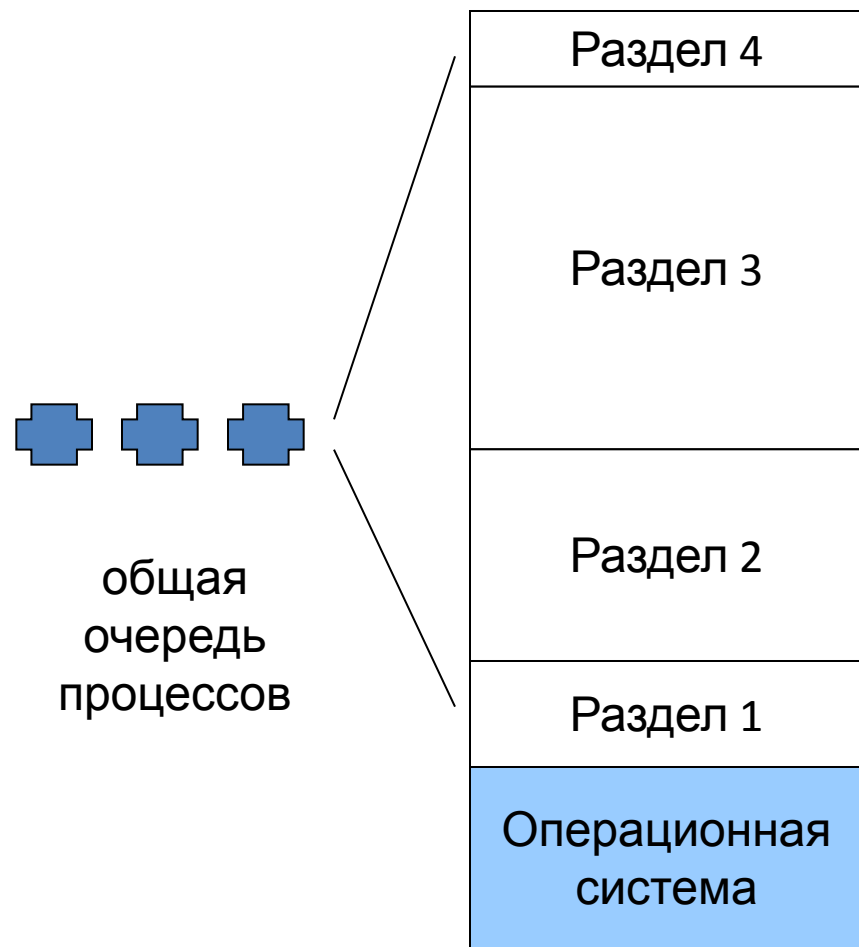
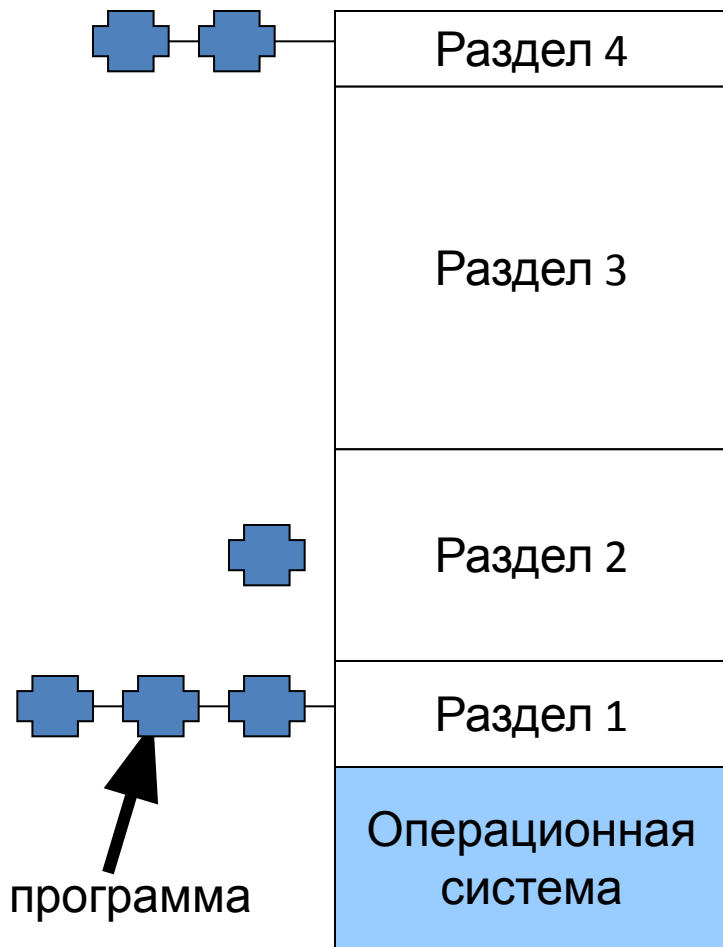




# КОГДА ПРОЦЕССОВ МНОГО....



# КОГДА ПРОЦЕССОВ МНОГО.... ФИКСИРОВАННЫЕ РАЗДЕЛЫ



Альтернатива – динамические разделы, рабочий набор...

## АЛГОРИТМЫ УТИЛИЗАЦИИ СТРАНИЦ

- идеальный (оптимальный) алгоритм.
- алгоритм FIFO;
- алгоритм «вторая попытка»
- не использовавшаяся в последнее время страница - NRU;
- страница, не использовавшаяся дольше всего;
- алгоритм «часы»
- и т.п.

## Контрольные вопросы

- Чем определяется максимальный размер используемой оперативной памяти?
- В чем отличие логического адреса от линейного? Линейного от физического?
- Что такое сегментация? Зачем она используется?
- В чем преимущество виртуального адресного пространства?
- Что такое подкачка страниц? Область подкачки? Как это влияет на безопасность системы?
- Зачем используется микросхема SPD?
- Что такое карта ресурсов? Зачем она используется?
- Какую информацию поддерживает менеджер виртуальной памяти? Диспетчер подкачки страниц?