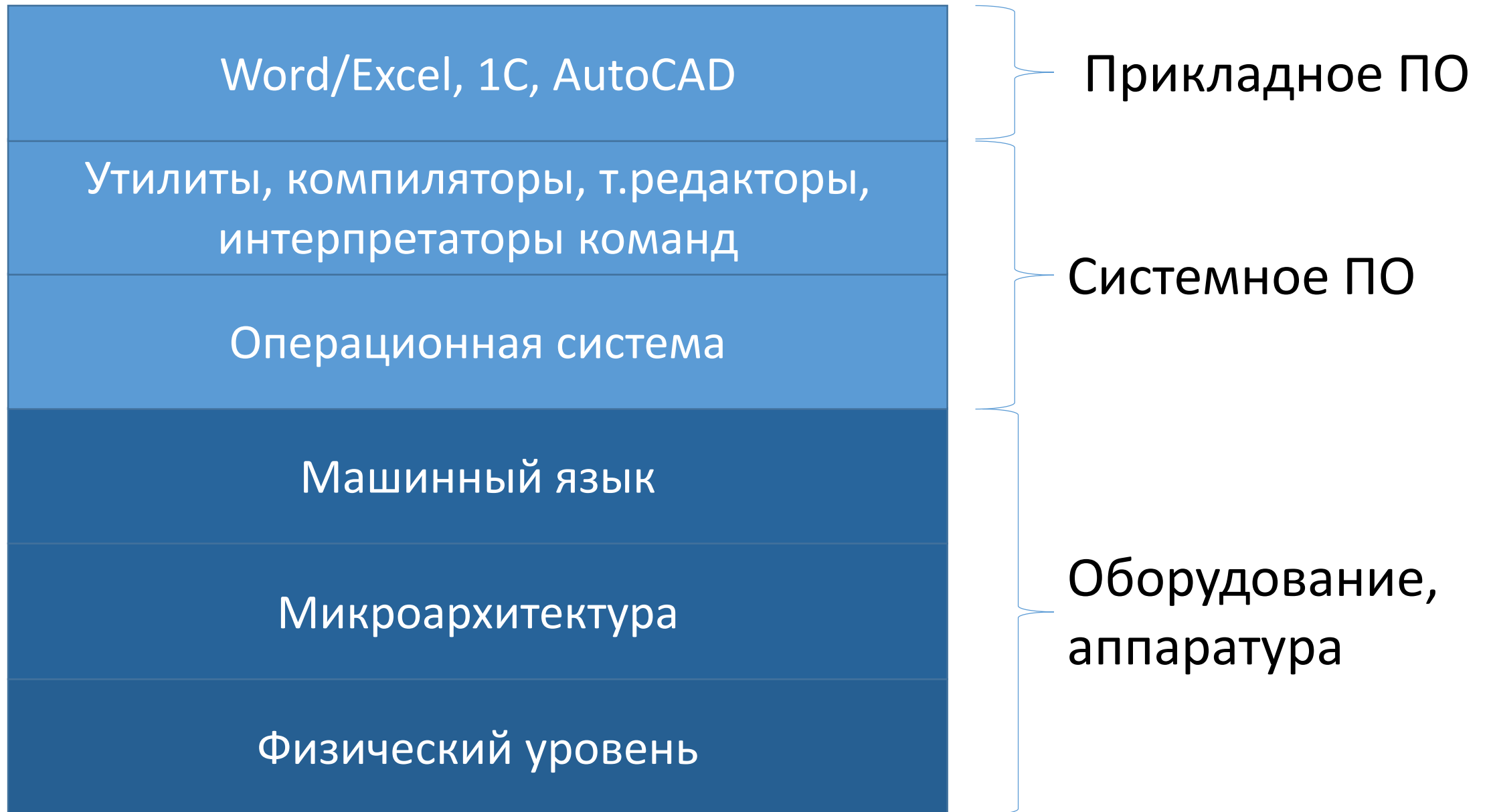


# Состав компьютерной системы



# Функции ОС

## ОС как расширенная машина

Предоставление удобного интерфейса для доступа к оборудованию, одни и те же функции позволяют работать с однотипным оборудованием от разных производителей и с разными возможностями

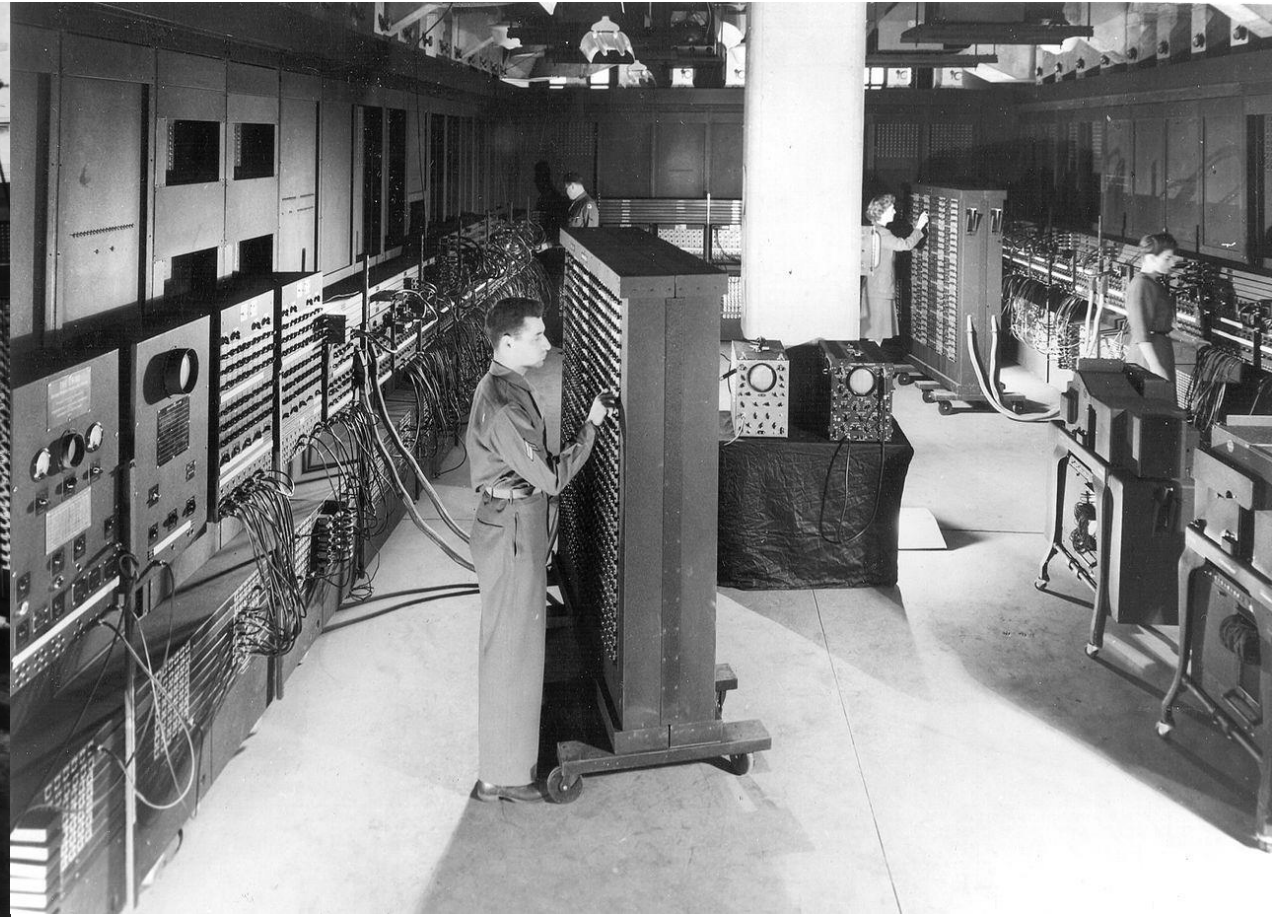
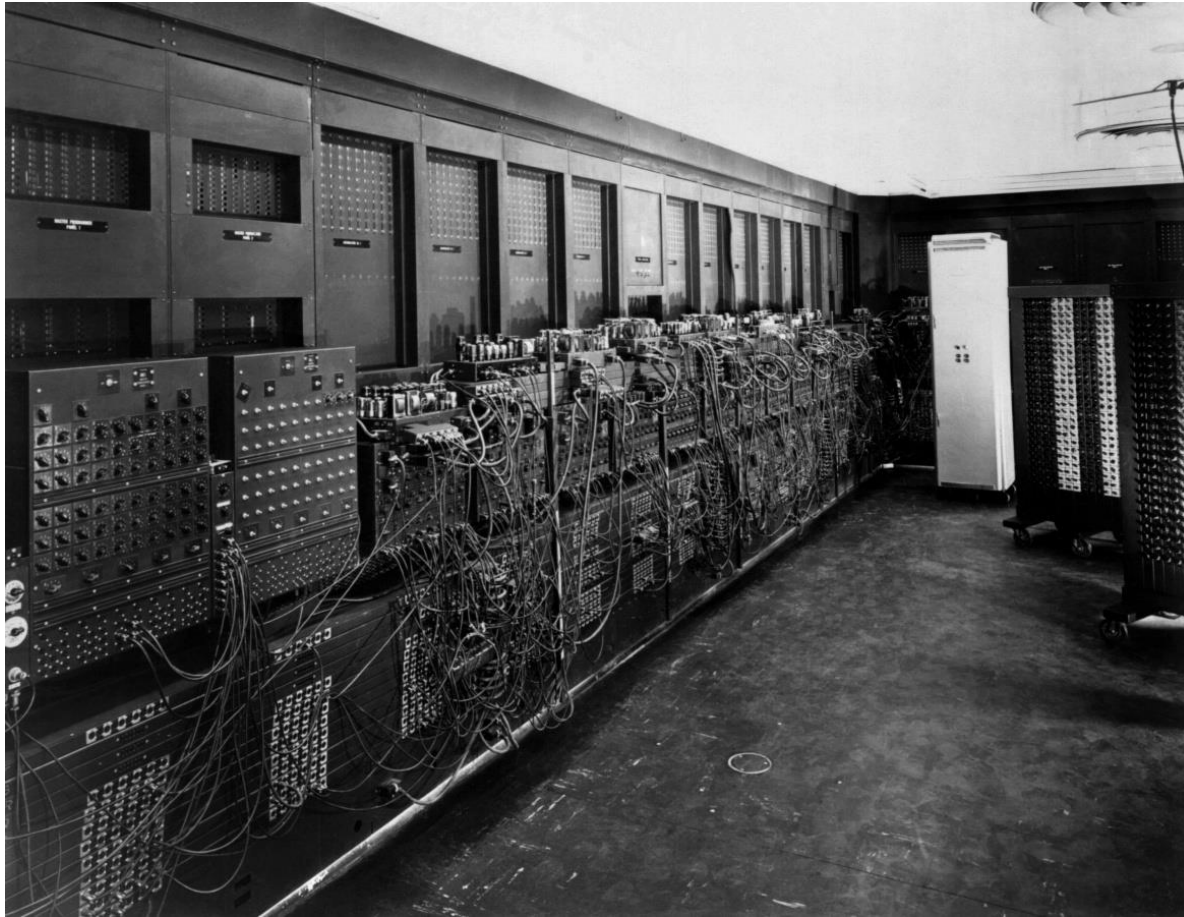
## ОС как менеджер ресурсов

Распределение ресурсов как по очередности использования(ЦП), так и по использованию разных областей устройств (память)

# История ОС

**Первое поколение 1945-1955**

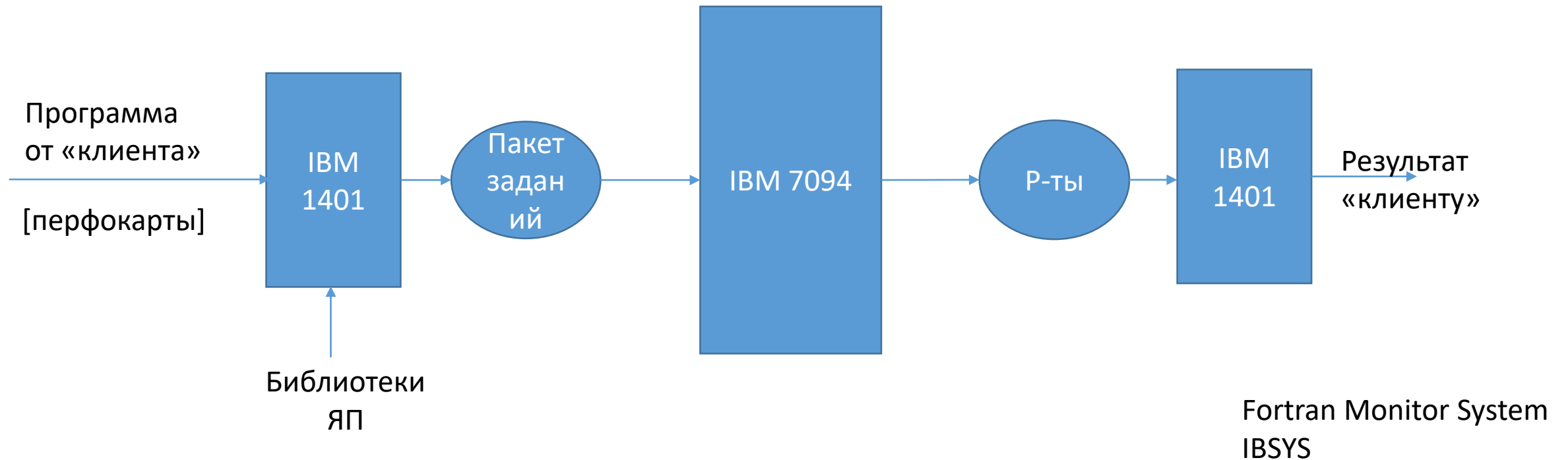
Нет операционных систем



## Второе поколение 1955-1965

Системы пакетной обработки, появились ЯП Fortran, Ada

Схема организации работы пакетной обработки



# История ОС

## Третье поколение 1965-1980

Системы разделения времени

- «Универсальный компьютер» - IBM/360, OS/360
- «1байт=8бит» - IBM/360
- «Доступный компьютер» - DEC PDP-1 1961  
Цена 5% (120к\$) от цены IBM 7094
- PDP-1 эволюционировал в VAX/VMS к 1990гг
- MULTICS – амбициозный [исследовательский] проект многопользовательской ОС
- «ОС написанная на ЯП высокого уровня» - Unix
- Стандартизация ОС UNIX - POSIX





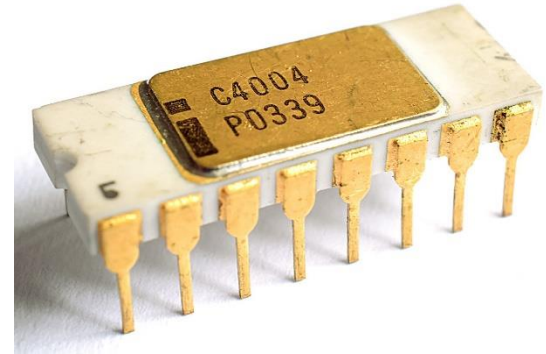
## Четвертое поколение 1980-н.вр. Персональные ЭВМ и развитие компьютерных сетей

«Первый микропроцессор» Intel 4004 (1970г) Intel 8080 (1974г)

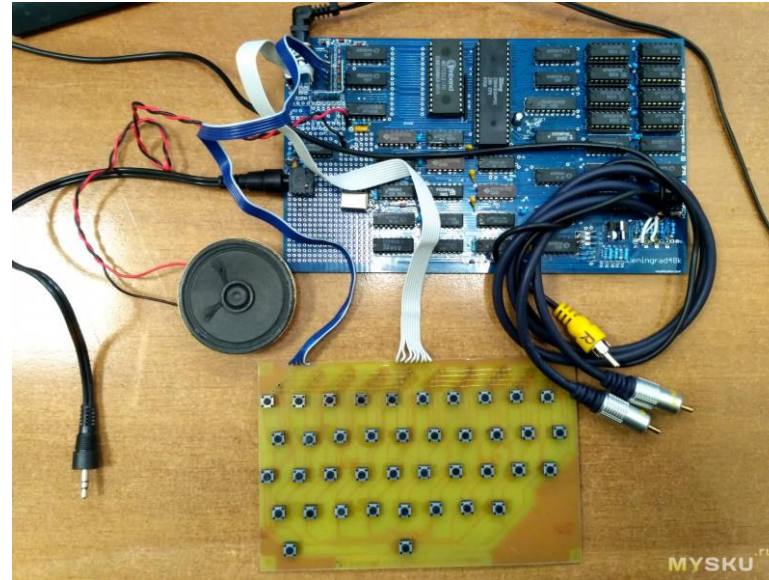
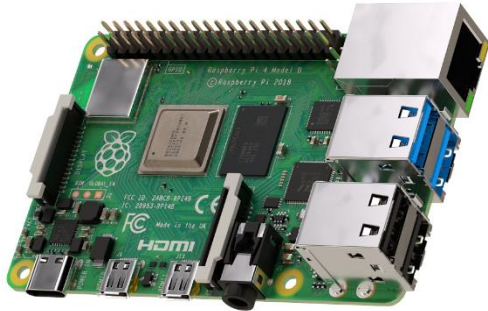
«Удачный процессор» - Zilog Z80 (1976г)

«Персональная ЭВМ» IBM – IBM/PC

«Компьютер за 100фунтов» - Spectrum Z80



Миниатюризация компьютерной техники



Первые сети на основе телефонных каналов - x.25 - SprintNet, Compuserve

Некоммерческая «Сеть друзей» FidoNet (тип сети - OffLine, 1984г)

Проект военного ведомства США DARPA; IP-сети – Internet

Сети финансовых институтов – межбанковские платежи SWIFT, платежные системы Visa/MasterCard

# Типы операционных систем

1. ОС для мейнфреймов IBM SystemZ

2. Серверные ОС

3. Для ПК

4. Многопроцессорные

5. ОС реального времени

Управление технологическим оборудованием, обычно понимается – Время реакции на внешнее событие  $t \ll$  Проектное время реакции  $T$





# Типы операционных систем

## 6. Встраиваемые ОС

ОС FreeRTOS для  
микроконтроллера  
ATMega32  
(32к Flash + 1кб EEPROM + 2Кб ОЗУ)



## 7. ОС для смарт-карт



## 8. Распределенные ОС "MPI", beowulf





# Принципы разработки современных ОС

Концепция абстрактной машины:

- Иерархическая
- Независимость операций вв/выв от внешнего устройства
- «Соккрытие информации»

Оболочка
Процессы пользователей
каталоги
устройства
....
....
....
Система команд
Электрические схемы

# Требования, предъявляемые к операционным системам

1. Расширяемость
2. Переносимость
  1. Большая часть ОС написана на ЯП высокого уровня, компилятор есть на целевой платформе
  2. Код на машинном языке минимизирован
  3. Код на машинном языке локализован в небольшом кол-ве файлов
4. HAL Hardware Abstraction Level
  5. Учет специфики целевой платформы
3. Надежность и отказоустойчивость
4. Совместимость
  1. Эмуляция платформы (QEMM)
  2. Эмуляция системных вызовов (API) wine (WinApi), mono (.Net)
  3. Совместимость исходных кодов
5. Безопасность (“Orange Book”, 1983г; A, B, C: C1, C2: секретный вход, избирательный контроль доступа, средства аудита, защита памяти D; SELinux)
6. Производительность

## **Порядок загрузки ЭВМ**

1. Тестирование ЭВМ программами из энергонезависимой памяти
2. Запускается первичный загрузчик (из ПЗУ)
3. (первичный) Загружается вторичный загрузчик (с внешнего накопителя) и передается управление (GRUB – Unix)
4. Загружается ядро ОС, инициализируется и передается управление [Unix – INIT SystemV, SystemD]
5. Запускается ПО

## **Типы загрузчиков (в архитектуре IBM/PC):**

- MBR
- UEFI, GPT

# Система управления вводом/выводом

## 1. Работа с оборудованием

А. опросы оборудования

Б. Работа по прерываниям

а. блокировка обработки прерывания

в. Определяется тип прерывания и происходит вызов обработчика прерывания

г. Работает обработчик прерывания

д. снимается блокировка обработки прерывания

## 2. На уровне ОС/на прикладном уровне

> **через драйвера/API**

а. блокировка объект (устройство) / ожидаю возможности заблокировать

б. работаю с объектом (устройство)

в. Разблокировка объекта (устройства)

> **SPOOL** – обработчик запросов блокирует объект (устройства), считывает очередь

а. пользователи «кладут» задания в очередь

б. обработчик берет задания по одному

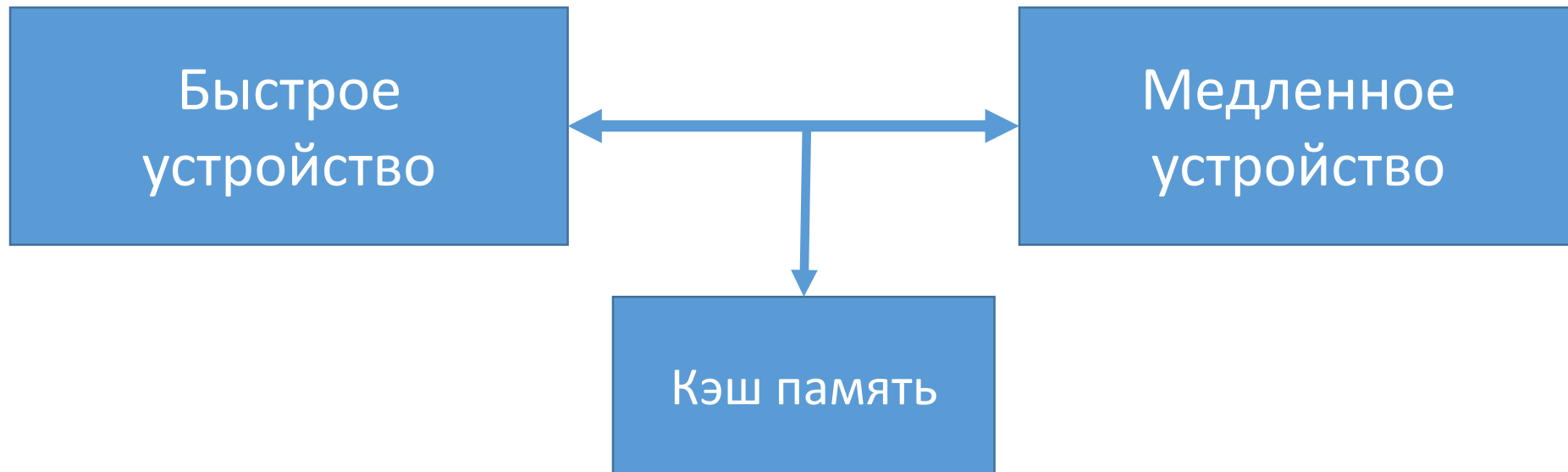
Б. Работает обработчик прерывания

HP Color LaserJet Pro MFP M477 PCL 6						
Принтер Документ Вид						
Документ	Состояние	Владелец	Число ...	Размер	Поставлено в очередь	Порт
Серверы до 100 000 руб - гара...	Постановка в очередь	alex	12	6,61 МБ	12:14:02 27.02.2021	
Серверы до 100 000 руб - гара...		alex	32	19,2 МБ	12:13:30 27.02.2021	
Системы управления печать...		alex	6	2,19 МБ	12:13:14 27.02.2021	
диспетчер печати туп — Янд...	Печать	alex	3	827 КБ	12:13:02 27.02.2021	HPColorLaserJetMFPM477fdw



# Кэширование

Схема взаимодействия устройств



Режимы работы кэш-памяти для устройств хранения (HDD/SSD):

## Write Through

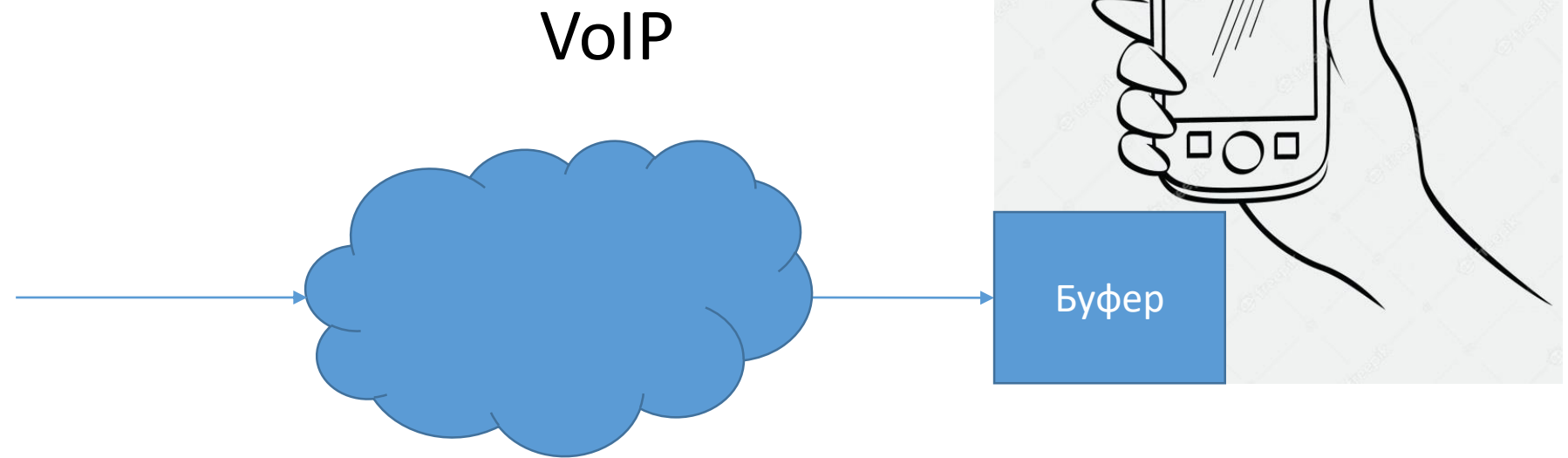
- Запись сразу
- Относительно медленная работа

## Write Back

- Запись отложена
- Есть риски утери информации

# Буферизация

Метод обмена с использованием буфера(памяти) для временного хранения данных



Равномерная обработка информации с точки зрения пользователя-получателя  
Отправка происходит быстро с точки зрения отправителя  
Может наблюдаться «отставание» при приеме данных получателем

# Определения

1. Системные вызовы (API)

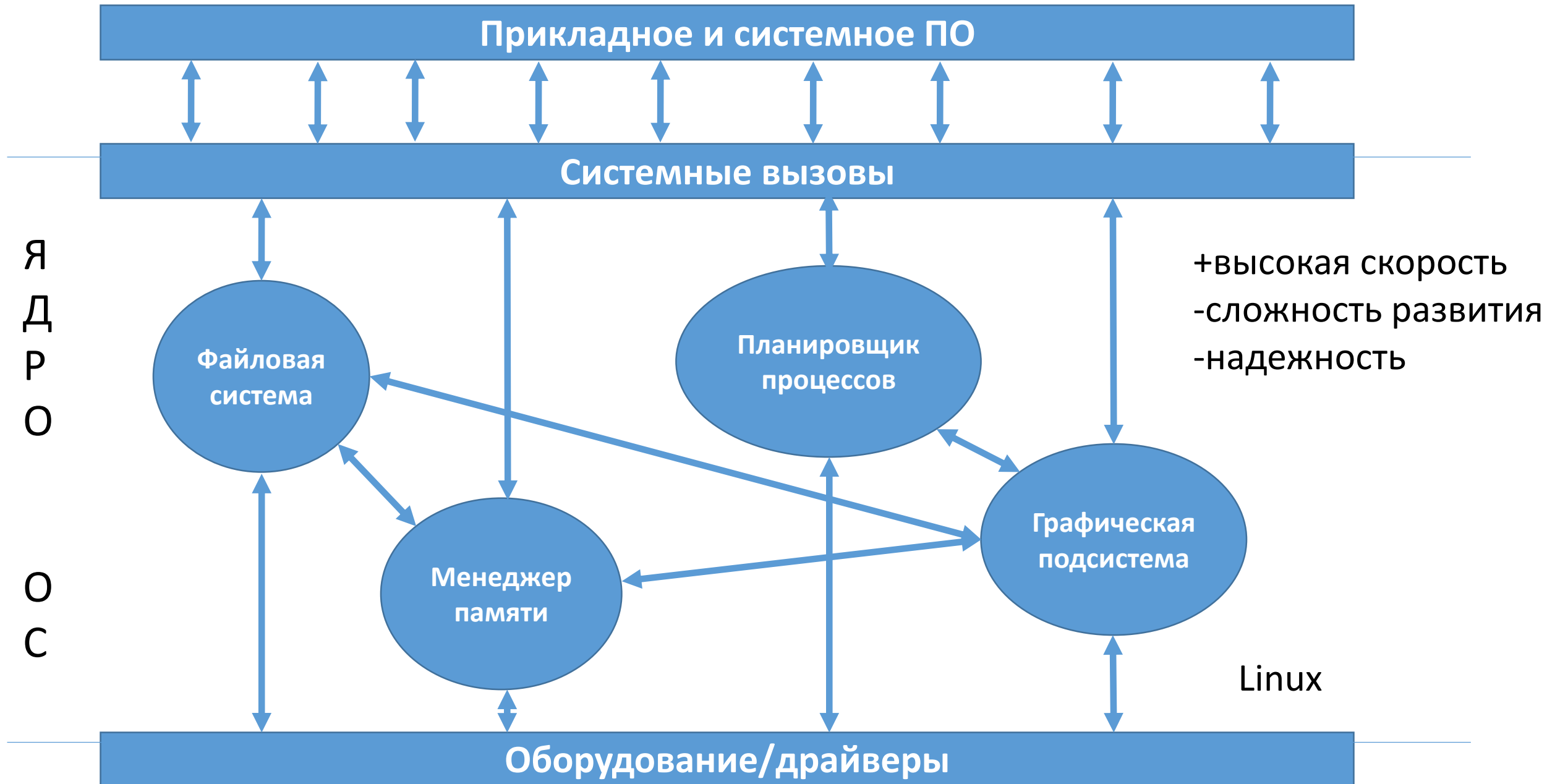
```
int fork();    int getpid();
```

2. Файлы и каталоги

3. Процессы и потоки/волокна thread/fiber

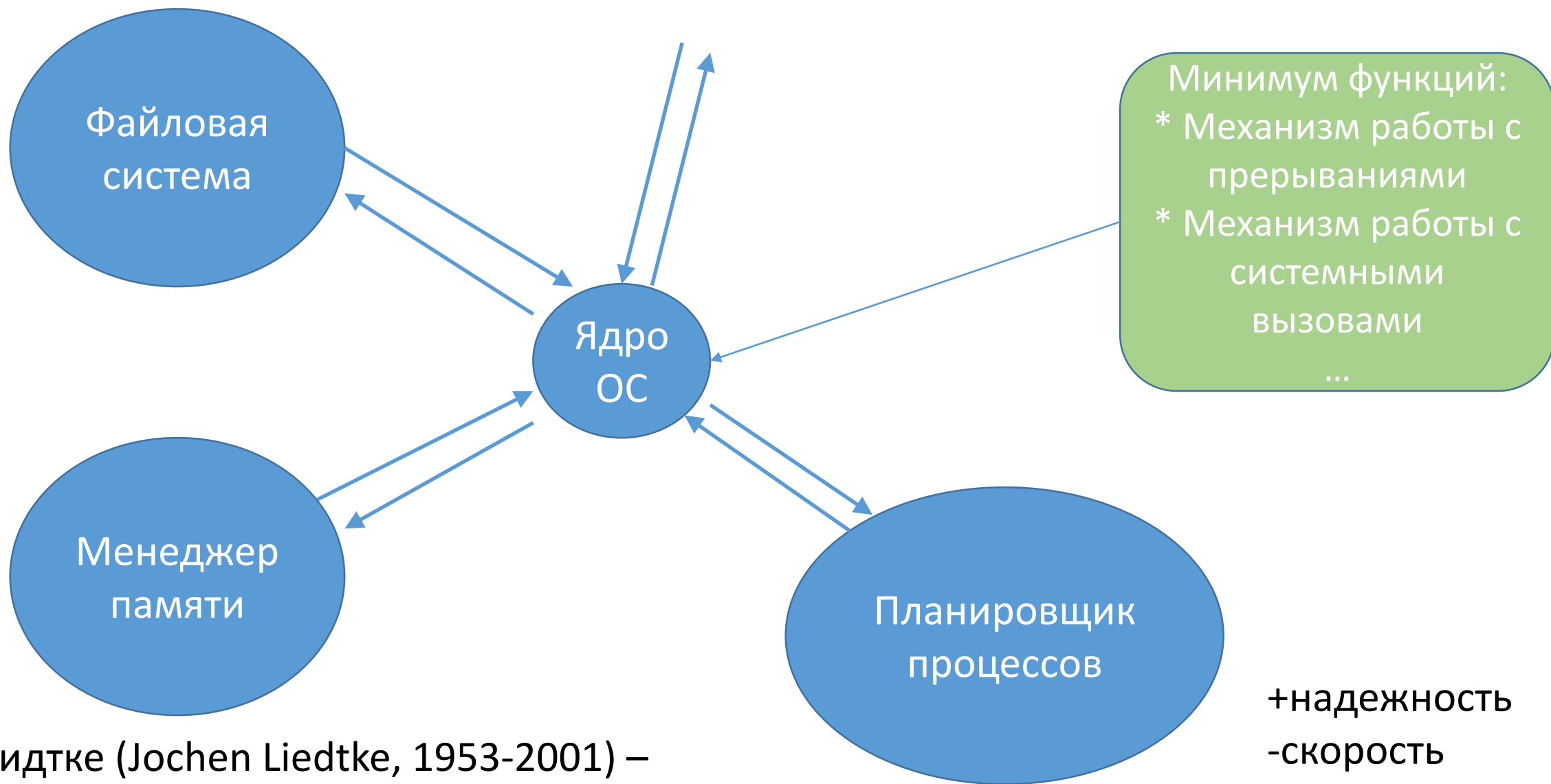
4. Ядро ОС – монолитные, микроядро, гибридные

# Ядра операционных систем: монолитные



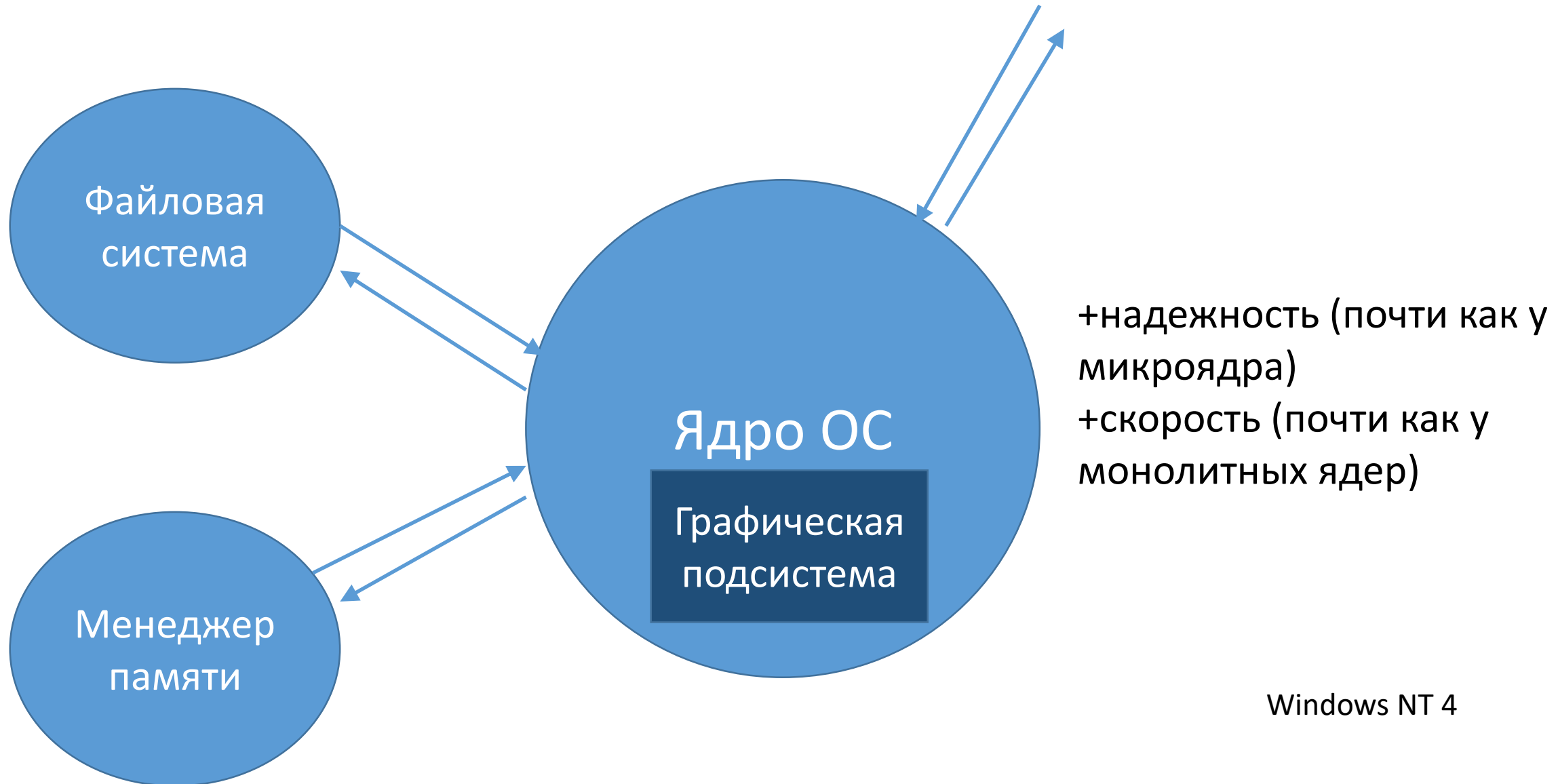


# Ядра операционных систем: микроядро



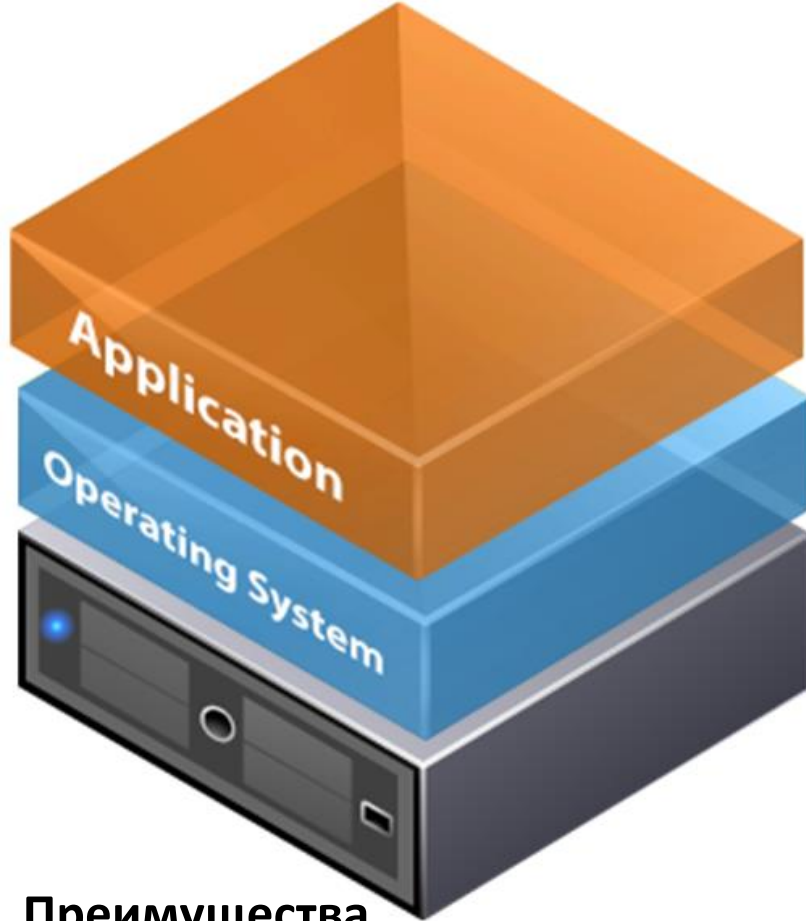
Йохен Лидтке (Jochen Liedtke, 1953-2001) –  
создал L4 - оптимизированное микроядро

# Ядра операционных систем: гибридное ядро ОС



# Виртуализация

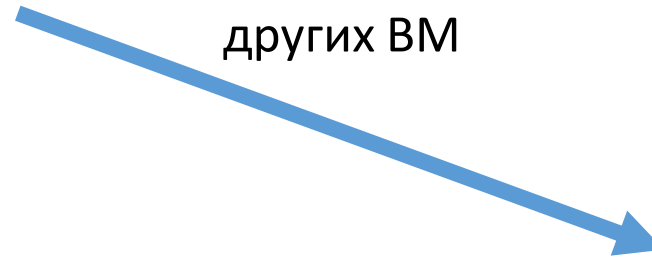
- позволяет повысить эффективность использования серверов



Типы гипервизоров

- Автономный гипервизор VmWare ESXi
- Гибридный Xen
- На основе базовой ОС MSVirtual PC, VmWare Player

ВМ надежно изолированы друг от основной ОС и других ВМ



## Преимущества

1. Повышение изоляции
2. Безопасность
3. Распределение ресурсов
4. Постоянная доступность
5. Повышение качества администрирования

**Минус** – на поддержку виртуализации тратятся ресурсы, поэтому для ресурсоемких приложений виртуализация нежелательна

# Модели управления доступом

1. Дискреционная(избирательная) модель
2. Ролевая модель
3. Мандатная модель



# Классификация ОС

1. Реализация многозадачности
  1. Однозадачные MSDOS
  2. Многозадачные WindowsNT...Win11..., Unix
2. Поддержка многопользовательского режима
  1. Однопользовательские Dos, win95
  2. Многопользовательские WinNt...Win11... , unix
3. Многопроцессорной обработки
4. Системы реального времени
  1. QNX