

# Краткая история ЭВМ

Первое поколение ЭВМ  
(1945-1955): электронные лампы и  
коммутационные панели

# ЭВМ

Опр. ЭВМ (электронная вычислительная машина, компьютер, computer) – это комплекс электронных устройств, предназначенных для хранения, преобразования и передачи данных.

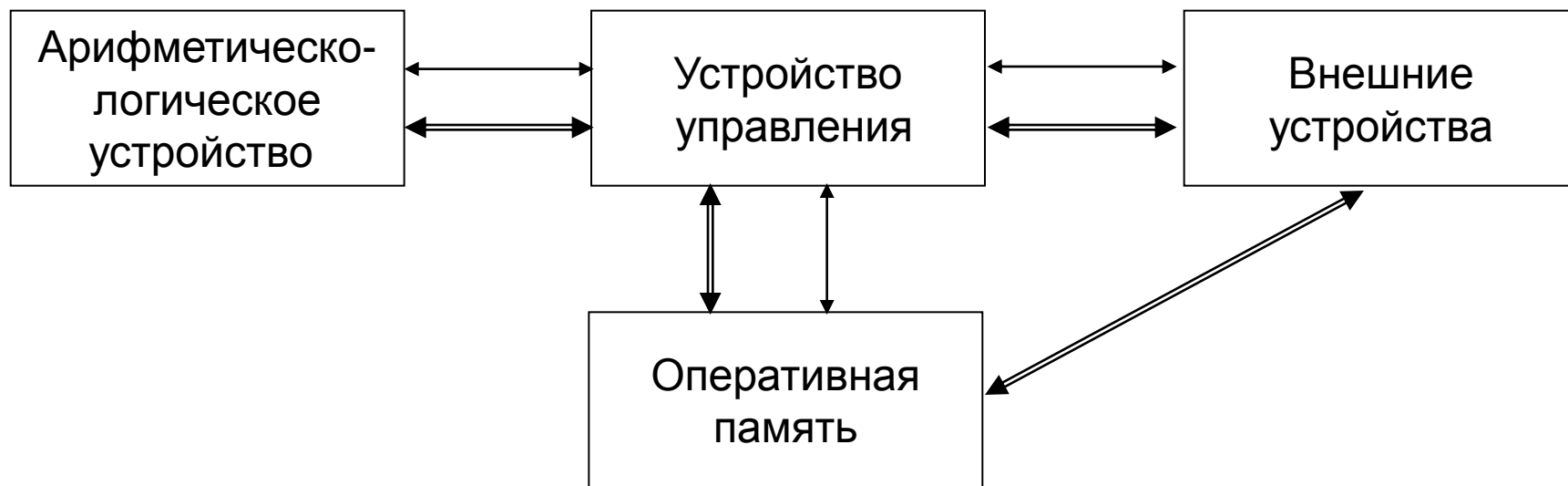
# Когда появилась первая ЭВМ?

- Первый компьютер на основе электронных ламп, в котором были воплощены принципы Джона фон Неймана (John von Neumann), был построен в 1948 году Морисом Уилксом (Morris Wilkes), Великобритания
- Во время второй мировой войны разрабатывались вычислительные машины на основе электромеханических реле
  - Конрад Цузе (Konrad Zuse), 1941 год, Германия
  - Говард Айкен (Howard Aiken), 1943 год, США

# Джон фон Нейман

- Родился в 1903 году в Будапеште
- В 1930 году приглашен на работу в научно-исследовательский Институт Перспективных Исследований (Institute for Advanced Study), Принстон, США
- Праотец современной архитектуры компьютеров и участник Манхэттенского проекта
- В 1957 году заболел раком кости и умер в тяжёлых мучениях

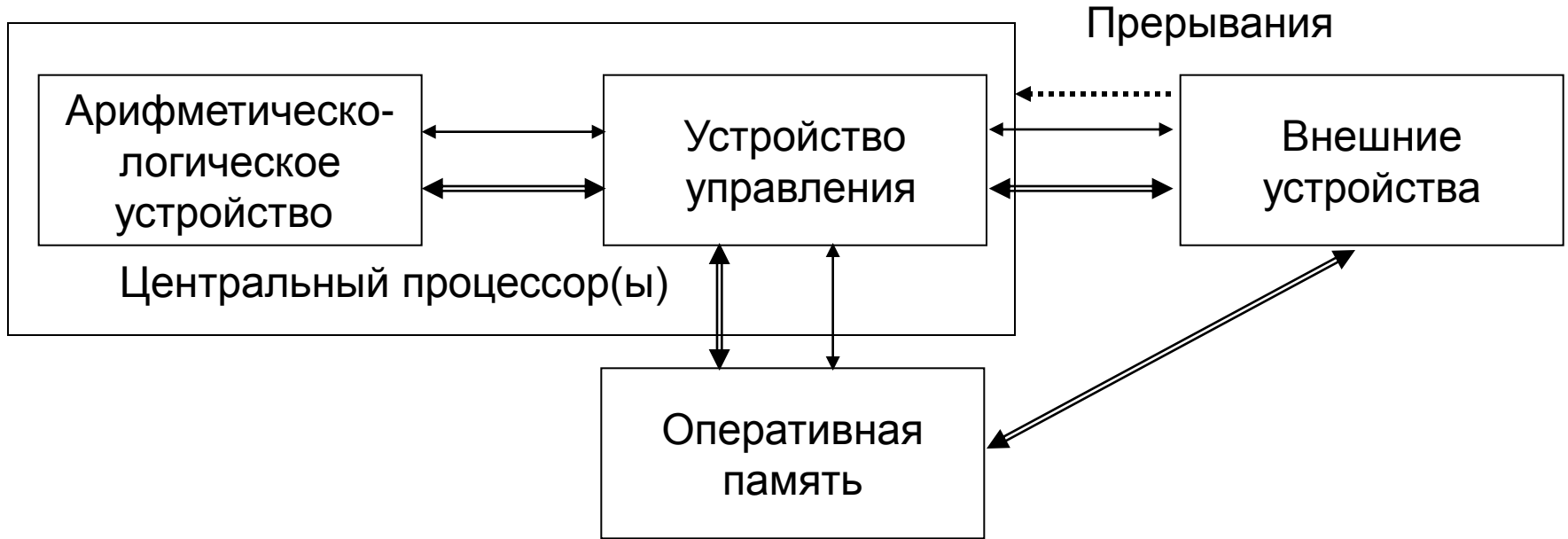
# Архитектура ЭВМ: принципы Джона фон Неймана



====> Поток данных

—> Управляющие связи

# Архитектура современных ЭВМ



====> Потоки данных

—> Управляющие связи

# Первое поколение ЭВМ

- Десятки тысяч электронных ламп (вероятность отказа)
- Программирование на абсолютном машинном языке
- Не было ОС, языков программирования (даже Ассемблера)
- Машину разрабатывала, программировала и эксплуатировала одна команда
- На смену коммутационным панелям в начале 50-х годов пришли перфокарты

# Вопрос для самопроверки

- Существовали ли ЭВМ во время второй мировой войны? (Да/Нет)



# Вопрос для самопроверки

- Существовали ли ЭВМ во время второй мировой войны? (Да/Нет)
- Нет. Вычислительные машины во время второй мировой войны работали на электромеханических реле.  
Электронные вычислительные машины на электронных лампах появились в конце 40-х годов.

# Вопрос для самопроверки

- Имели ли ЭВМ первого поколения операционные системы? (Да/Нет)

# Вопрос для самопроверки

- Имели ли ЭВМ первого поколения операционные системы? (Да/Нет)
- Нет. В первых ЭВМ операционных систем не было. Программисты вводили команды на машинном языке при помощи коммутационных панелей или механических переключателей.

# Краткая история ЭВМ

Второе поколение ЭВМ  
(1955-1965): транзисторы и  
системы пакетной обработки

# Транзистор

Опр. Транзистор (transistor) – миниатюрный полупроводниковый переключатель, который пропускает или не пропускает ток, позволяя процессорам выполнять операции а памяти ЭВМ хранить данные по битам. Изобретены в США в 1948 году.

# Второе поколение ЭВМ (1955-1965)

- Применение транзисторов сделало ЭВМ более надежными
- Однопоточные системы пакетной обработки
- Первые операционные системы и языки программирования

# Однопоточная система пакетной обработки данных

Опр. Однопоточная система пакетной обработки данных (single-stream batch-processing system) – разновидность ранних компьютерных систем, которая последовательно выполняла ряд неинтерактивных задач, по одной за раз.

Пример. Первая операционная система для компьютера IBM 701 (середина 1950-х)

# Задача

Опр. Задача (job) – совокупность работ, которая должна быть выполнена компьютером.



# Языки программирования

- Машинный язык
- Язык ассемблера
- Язык Фортран

# Машинный язык

Опр. Машинный язык (machine language)

– язык, который определяется структурой аппаратных средств компьютера и может быть непосредственно воспринят ими. Программа на машинном языке записывается в двоичных кодах.

# Язык ассемблера

Опр. Язык ассемблера (assembly language)

– язык низкого уровня, который представляет основные операции компьютера в виде английских сокращений названий команд.

Пр. LOAD     basePay  
      ADD     overTimePay  
      STORE  grossPay

# Ассемблер

Опр. Ассемблер (assembler) – программа транслятор, которая переводит программы с языка ассемблера на машинный язык. Обычно является подпрограммой операционной системы.

# Язык высокого уровня

Опр. Язык программирования высокого уровня (high-level language) – язык программирования, использующий английские идентификаторы и простой синтаксис для написания программы с применением меньшего количества операторов, чем при программировании на языках ассемблера.

Пр.  $grossPay = basePay + overTimePay$

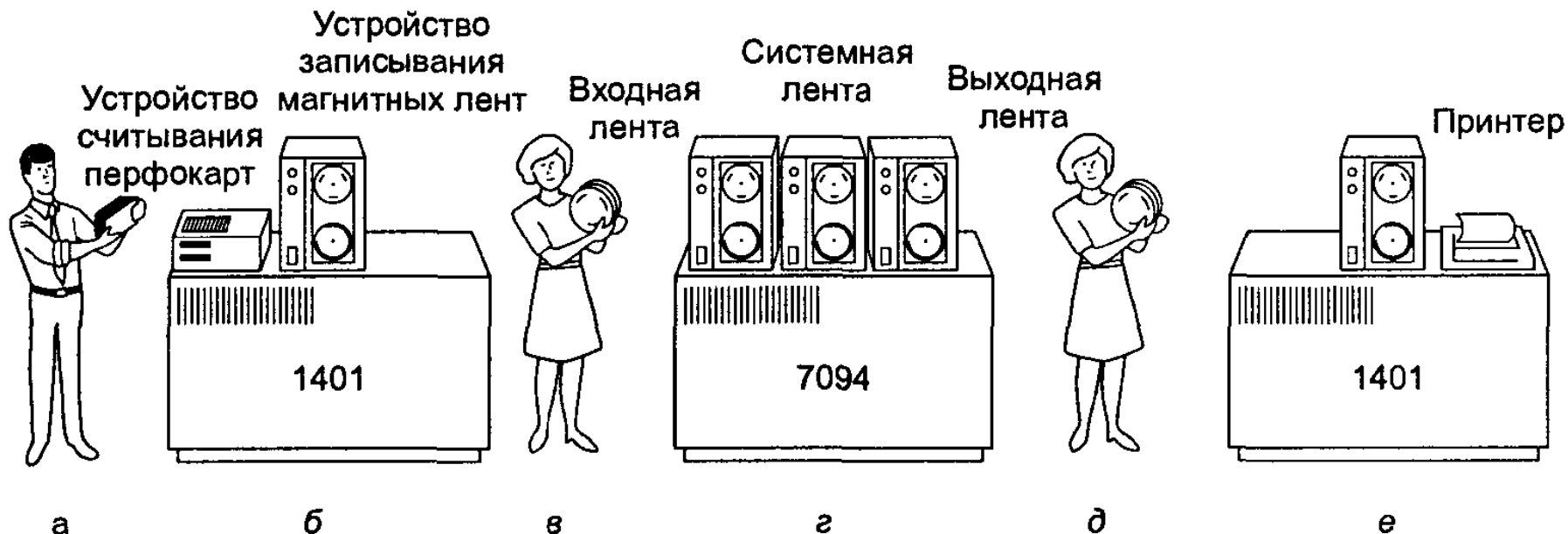
# Компилятор

Опр. Компилятор (compiler) – приложение, которое транслирует исходный код на языке высокого уровня в машинный код.

# Язык Фортран

Опр. Язык Фортран (Fortran) – язык программирования высокого уровня, разработанный компанией IBM в середине 1950-х годов для научных приложений, которые требовали сложных математических вычислений.

# Ранняя система пакетной обработки



а) Программист приносит карты для IBM 1401

б) IBM 1401 записывает пакет заданий на магнитную ленту

в) Оператор приносит входные данные на ленте к IBM 7094

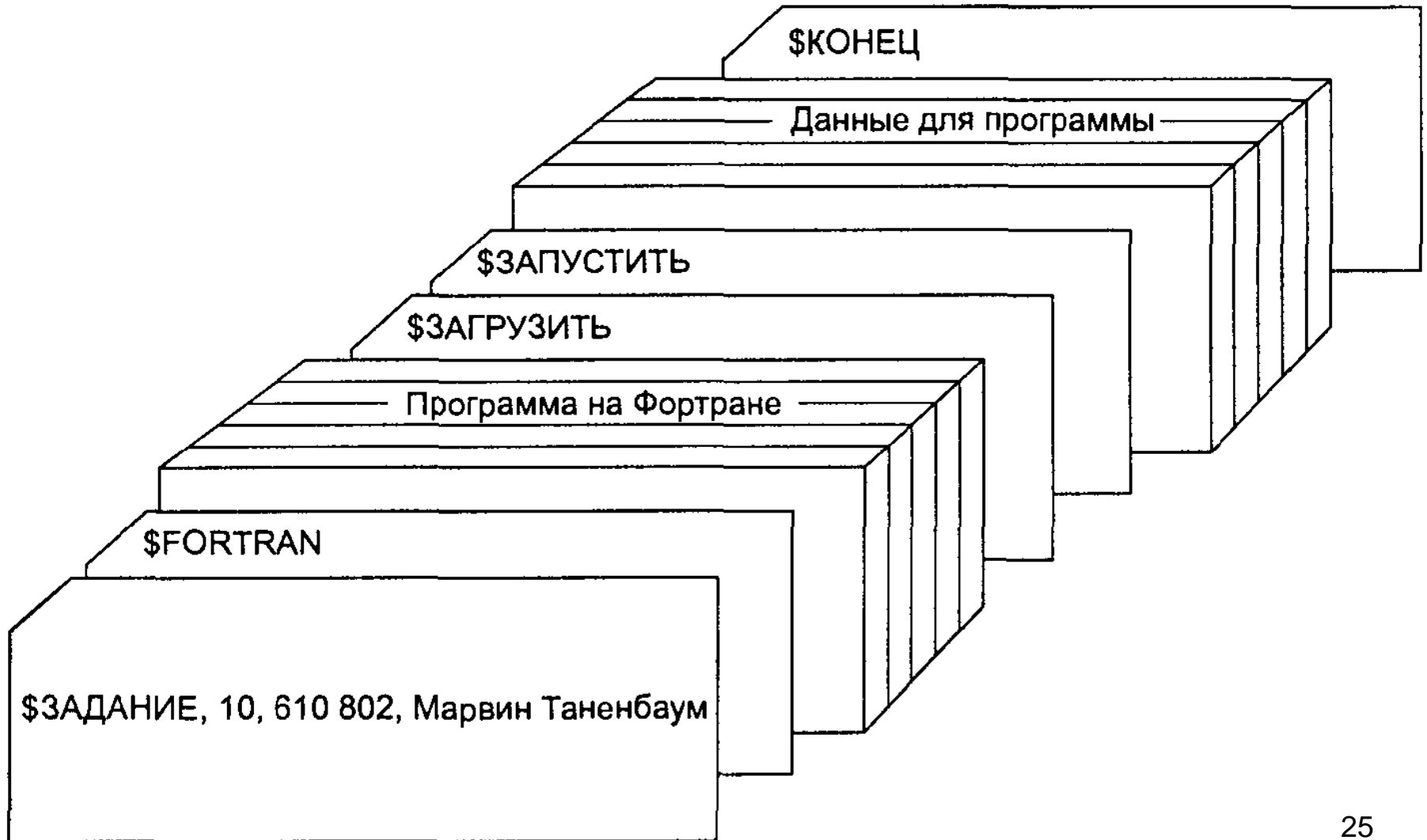
г) IBM 7094 загружает компилятор с системной ленты, выполняет вычисления и записывает результат на выходную ленту

д) Оператор переносит ленту с выходными данными на IBM 1401

е) IBM 1401 печатает выходные данные



# Структура типичного задания FMS (Операционная система Fortran Monitor System)



# Вопрос для самопроверки

- Верно ли, что компьютеры выполняют программу непосредственно на языке ассемблера? (Да/Нет)

# Вопрос для самопроверки

- Верно ли, что компьютеры выполняют программу непосредственно на языке ассемблера? (Да/Нет)
- Нет. Программы-трансляторы осуществляют перевод программы с языка ассемблера на машинный язык, прежде чем ее можно будет выполнить.

# Вопрос для самопроверки

- Можно ли писать программное обеспечение на переносимом машинном языке? (Да/Нет)

# Вопрос для самопроверки

- Можно ли писать программное обеспечение на переносимом машинном языке? (Да/Нет)
- Нет. Машинные языки являются машинно-зависимыми, таким образом, программное обеспечение, написанное на определенном машинном языке, воспринимается только компьютерами одного типа.

# Краткая история ЭВМ

Третье поколение ЭВМ  
(1965-1980): интегральные  
схемы и многозадачность

# Интегральная схема

Опр. Интегральная схема (integral circuit)  
– электронное устройство, состоящее  
из множества транзисторов.

# IBM/360

(позднее 370, 390, zSeries)

- Первая линия компьютеров на интегральных схемах, 1964 год
- Серия программно совместимых машин разной производительности
- OS/360 состояла из миллиона строк, написанных на ассемблере тысячами программистов и содержала тысячи ошибок



# Многозадачность

Опр. Многозадачность (мультипрограммность, multiprogramming) – возможность одновременного хранения в памяти большого количества программ, так что они могут выполняться одновременно.

# Многозадачность ОС третьего поколения

- В разных разделах оперативной памяти одновременно находятся ОС и несколько задач, одни из которых ориентированы на вычисления, а другие – на ввод/вывод
- Пока одна задача ожидает завершения операции ввода/вывода, другая использует центральный процессор

# Ориентированный на ввод/вывод

Опр. Ориентированный на ввод/вывод (i/o-bound) процесс (или задача), как правило, использует процессор в течение короткого промежутка времени, чтобы сформировать запрос на ввод/вывод, а затем освобождает процессор.

# Ориентированный на вычисления

Опр. Ориентированный на вычисления (processor-bound) процесс (или задача), в полной мере использует свой квант процессорного времени при исполнении. Такие процессы, как правило, выполняют интенсивные вычисления и выдают мало запросов на ввод/вывод.

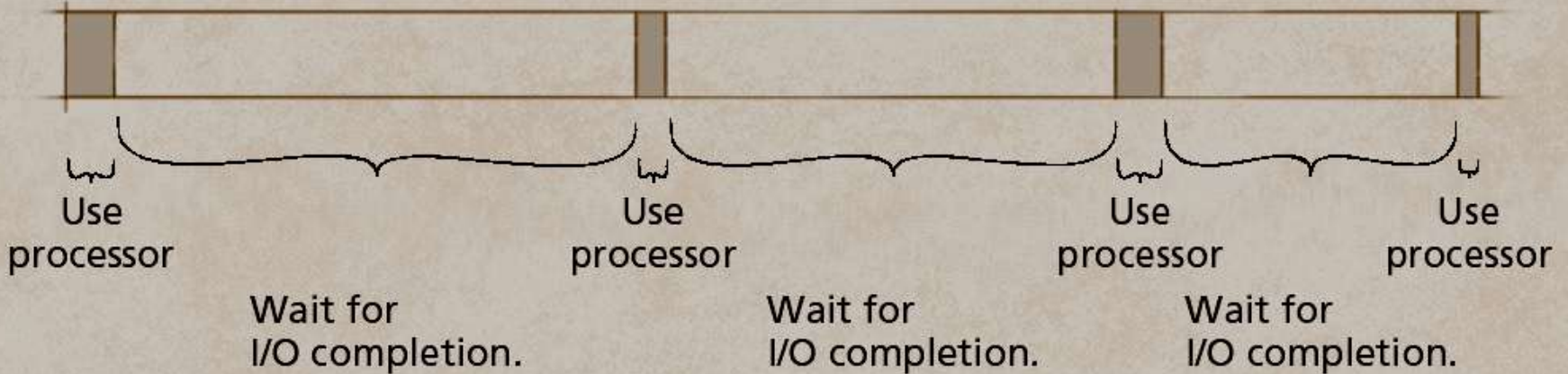
# Использование процессора в однозадачной системе

For a process doing intensive calculation:



Shaded area indicates "Processor in use."

For a process doing regular input/output:



# Спулинг

Опр. Спулинг (Simultaneous Peripheral Operation On Line, совместная периферийная операция в интерактивном режиме) – подкачка данных в оперативную память на фоне вычислений процессора.

# Спулинг ОС третьего поколения

- ОС считывает задания с перфокарт на диск в фоновом режиме
- Когда текущее задание заканчивается ОС загружает новое задание с диска в освободившийся раздел оперативной памяти

# Интерактивные пользователи

Опр. Интерактивные пользователи (interactive user) – пользователи, которые находятся непосредственно возле машины, пока та решает их задачи. Интерактивные пользователи взаимодействуют с задачами в процессе их решения с помощью диалоговых терминалов.



# Система с разделением времени

Опр. Система с разделением времени (timesharing system) – операционная система, которая делает возможным одновременную работу многочисленных интерактивных пользователей.

# CTSS

- CTSS (Compatible TimeSharing System) – совместимая система разделения времени
- Первая система с режимом разделения времени
- Разработана в 1960-х годах в Массачусетском технологическом институте (MIT) на специально переделанном компьютере IBM 7094

# Виртуальная память

Опр. Виртуальная память (virtual memory) – способность операционной системы предоставлять программам доступ к большему пространству адресов, чем фактически существует в оперативной памяти.

# Процесс

Опр. Процесс (process) – выполняемая операционной системой программа.

# MULTICS

- MULTICS (MULTiplexed Information and Computing Service) – мультиплексная информационная и вычислительная служба
- Совместная разработка МТИ, исследовательской лаборатории Bell Labs и корпорации General Electric на мейнфрейме GE-645 (вторая половина 1960-х)
- Компьютерное предприятие общественного пользования в районе Бостона

# MULTICS

- Первая крупная ОС, написанная на языке высокого уровня (PL/1)
- Одна из первых ОС с виртуальной памятью
- В системе MULTICS впервые использован термин процесс для описания программы в ходе ее выполнения

# Миникомпьютеры

- PDP-1 выпущен корпорацией DEC в 1961 году (позднее PDP-2,... PDP-11)
- PDP-1 продавался по цене 120 000 \$ (5 % цены IBM 7094)
- Кен Томпсон написал UNIX –  
однопользовательскую версию MULTICS для PDP-7

# UNIX

- В 1969 году Кен Томпсон (Bell Labs) создал OS UNICS (UNIX) в противовес MULTICS
- Через несколько лет система была переписана на интерпретируемом языке Томпсона B, а вскоре на компилируемом языке C (Денис Ритчи)
- В 1987 году Эндрю Таненбаум (университет Врейте, Амстердам) создал операционную систему Minix – урезанную версию UNIX для обучения основам проектирования операционных систем
- В 1991 году Линус Торвальдс, финский студент, использовал Minix в качестве основы при создании Linux



# Вопрос для самопроверки

- Повысило ли производительность труда программистов применение интерактивного режима работы? (Да/Нет)

# Вопрос для самопроверки

- Повысило ли производительность труда программистов применение интерактивного режима работы? (Да/Нет)
- Да. Время между постановкой задачи и возвратом результатов ее выполнения было сокращено от часов до минут. Это позволило программистам вводить, компилировать, редактировать, тестировать и отлаживать свои программы в интерактивном режиме, вплоть до устранения всех ошибок.

# Вопрос для самопроверки

- Была ли реализована в системе MULTICS концепция виртуальной памяти? (Да/Нет)

# Вопрос для самопроверки

- Была ли реализована в системе MULTICS концепция виртуальной памяти? (Да/Нет)
- Да. Эта система включала виртуальную память, которая позволяет приложениям обращаться к большему пространству памяти, чем физически существует в системе. Это предоставило программистам возможность разработки более мощных приложений и избавило от необходимости решать сложные задачи управления памятью.

# Краткая история ЭВМ

Четвертое поколение ЭВМ  
(1980-н.в.): микросхемы и  
персональные компьютеры

# Микросхема

Опр. Микросхема (большая интегральная схема, LSI, Large Scale Integration) – кремниевое микроминиатюрное электронное устройство, содержащее тысячи транзисторов на одном квадратном сантиметре.

Пример. Intel 8080 – первый универсальный 8-разрядный центральный процессор на микросхеме, 1974 год

# CP/M

- CP/M (Control Program for Microcomputers) – программа управления для микрокомпьютеров
- Первая операционная система для персональных компьютеров с Intel 8080
- Разработана в 1977 году Гэри Килдэллом (компания Digital Research)
- Лидировала среди ОС до 1983 года

# MS-DOS

- MS-DOS (Microsoft Disk Operation System) – дисковая операционная система фирмы Microsoft
- Первая операционная система для персональных компьютеров IBM PC
- Разработана в 1984 году Тимом Патерсоном (компания Microsoft )
- Работала в текстовом режиме



# GUI

Опр. Графический интерфейс пользователя (GUI, Graphical User Interface) – удобное для пользователя средство доступа к операционной системе, включающее в себя графические элементы, такие как окна, значки и меню, предназначенные для упрощения работы с программами и файлами.

# Дуглас Энгельбарт

- В 1968 году создал компьютерную систему NLS (oNLine System, диалоговая система), ее отличительными элементами были мышь и графический интерфейс с окнами
- Целью создания NLS было повышение уровня человеческого интеллекта и помощь людям быстрее решать задачи
- Основал Институт Совершенствования (Bootstrap Institute) с целью развития методов повышения уровня человеческого интеллекта

# GUI

- Apple Macintosh, 1984 год
  - Первая операционная система для персонального компьютера с GUI, поддерживающая мышь и окна
  - Разработана Стивом Джобсом
- Windows 1.0, 1985 год
  - Графическая оболочка для MS-DOS корпорации Microsoft
  - Развивалась до 1995 года

# Вопрос для самопроверки

- Верно ли, что Macintosh была первой операционной системой с графическим интерфейсом пользователя? (Да/Нет)

# Вопрос для самопроверки

- Верно ли, что Macintosh была первой операционной системой с графическим интерфейсом пользователя? (Да/Нет)
- Нет. Macintosh была первой операционной системой с GUI для персонального компьютера. Впервые GUI был реализован в 1968 году в системе NLS Дугласа Энгельбарта.

# Вопрос для самопроверки

- Верно ли, что MS-DOS разработал Бил Гейтс? (Да/Нет)

# Вопрос для самопроверки

- Верно ли, что MS-DOS разработал Бил Гейтс? (Да/Нет)
- Нет. MS-DOS разработал в 1984 году Тим Патерсон.

# Краткая история ЭВМ

История сети Интернет и  
всемирной паутины



# Сеть ARPA net

- Прародитель сети Интернет
- Разработана в 1969 году ARPA (Advanced Research Projects Agency, управление перспективных исследовательских программ Министерства обороны США)
- Соединяла около дюжины компьютерных систем университетов и научно-исследовательских организаций, финансируемых ARPA

# Сеть ARPA net

- Выделенные линии связи со скоростью 56 Кбит/сек.
- Передача информации посредством электронной почты (e-mail)
- Сеть ARPA net была децентрализована, перенаправляла трафик в обход поврежденных узлов
- Для ARPA net было разработано семейство протоколов TCP/IP

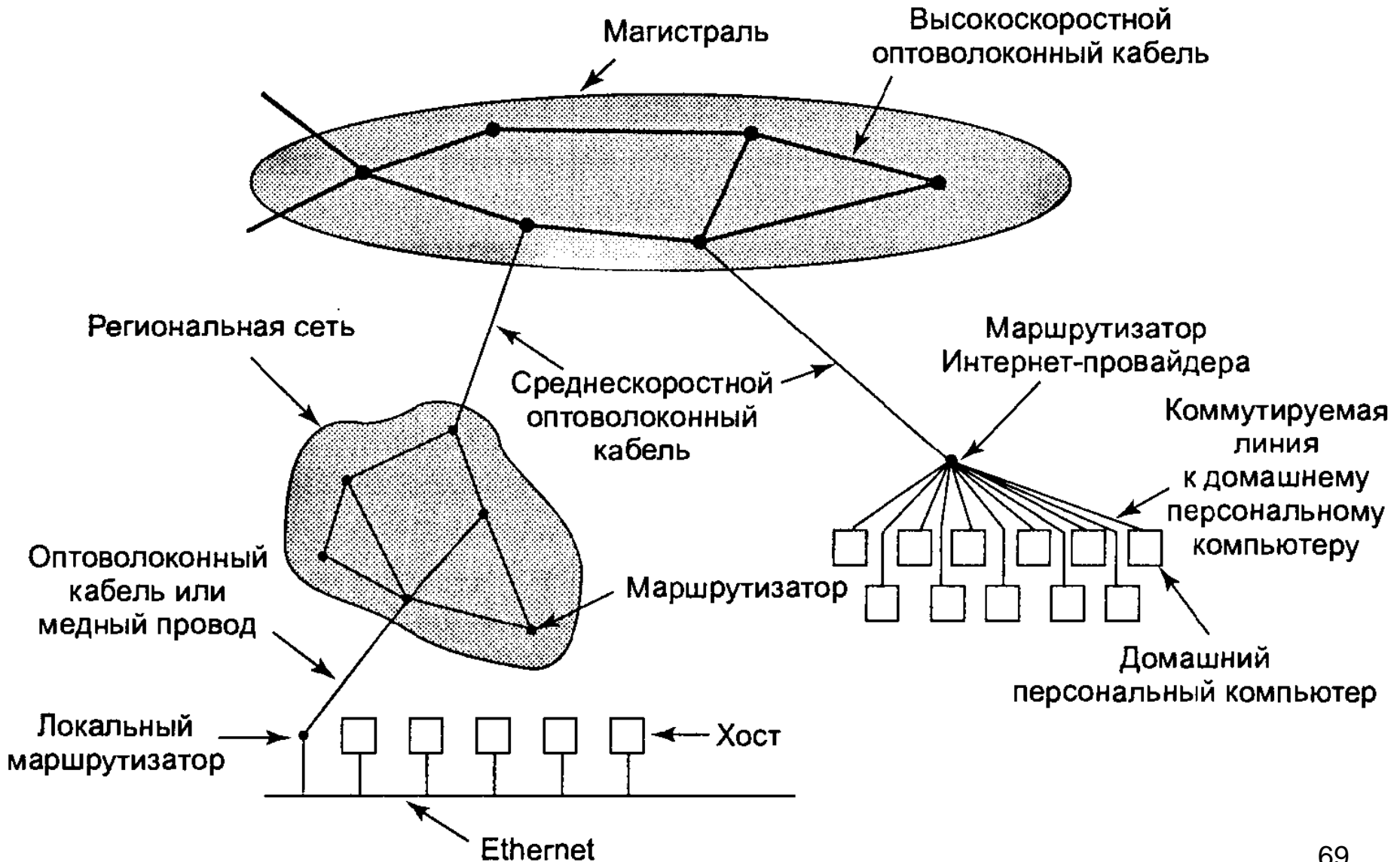
# TCP/IP

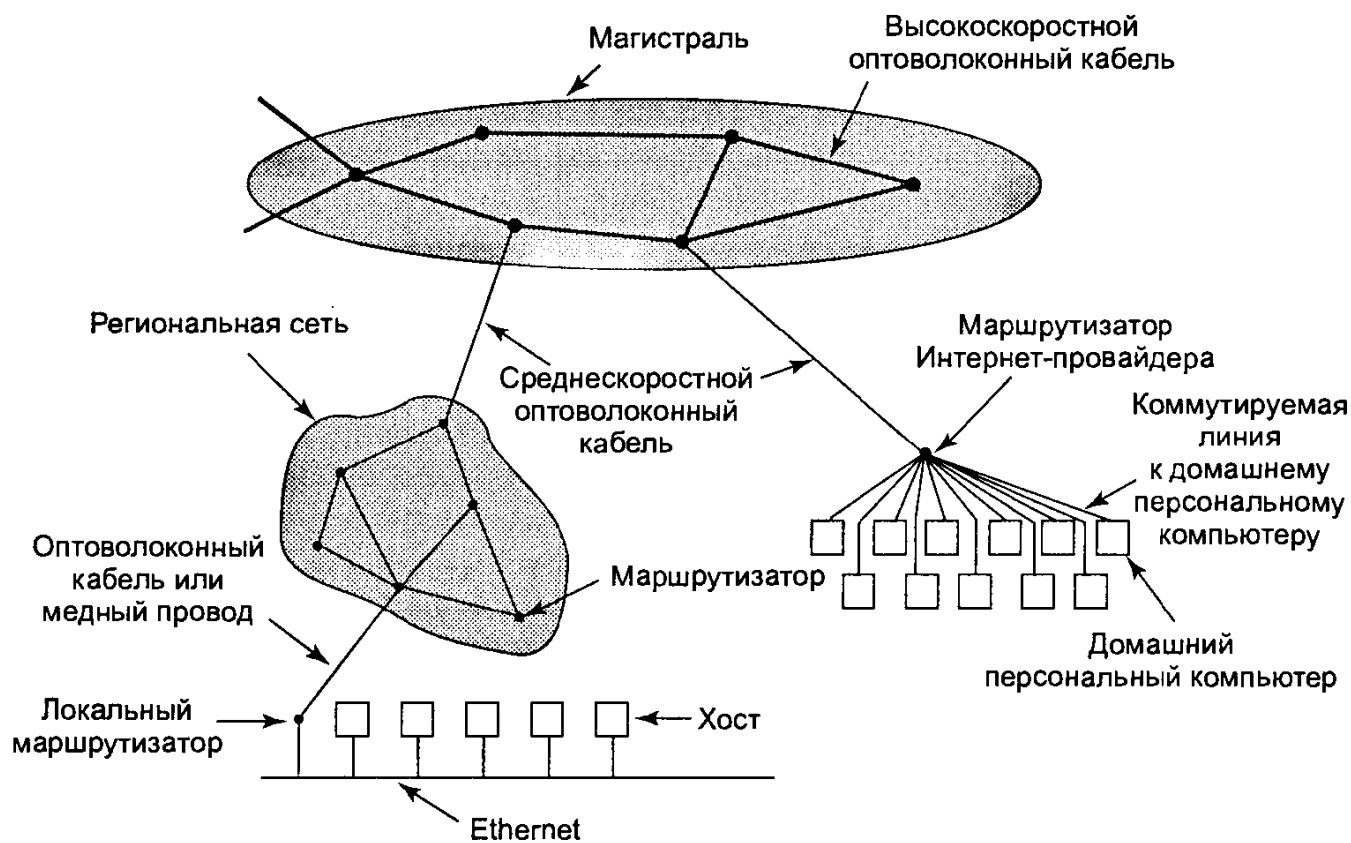
Опр. Протокол управления передачей / протокол Интернет (Transmission Control Protocol, TCP/IP) – семейство протоколов, которые определяют способы обмена информацией в сети Интернет.

# Интернет

Опр. Интернет (Internet) – сеть каналов связи, которая является технологической основой для телекоммуникаций и всемирной паутины. Каждый компьютер сети Интернет определяет, какими услугами сети он пользуется, и какие услуги он предоставляет другим компьютерам, подключенным к Интернету.

# Схема части Интернета





Ethernet – локальная сеть

Хост – клиентский компьютер

Маршрутизатор – коммутирующий компьютер, принимающий пакеты с одной или нескольких линий и отправляющий их дальше по одной или нескольким линиям

# Всемирная паутина

Опр. Всемирная паутина (World Wide Web, WWW) – совокупность гипертекстовых документов, доступных через сеть Интернет при помощи протокола передачи гипертекстовых файлов (HTTP). Веб-документы, как правило, пишутся на таких языках, как HTML.

# HTTP

Опр. Протокол передачи гипертекстовых файлов (HyperText Transfer Protocol, HTTP) – сетевой протокол, используемый для обмена документами на языке HTML и данными других форматов между клиентом и сервером. Это основной протокол всемирной паутины.



# HTML

Опр. Гипертекстовый язык разметки (HyperText Markup Language, HTML) – язык, который описывает содержимое и порядок отображения содержимого веб-страницы, а также предоставляет гиперссылки для доступа к другим страницам.

# Тим Бернерс-Ли

- Окончил Королевский колледж при Оксфордском университете в 1976 году
- Работая Европейской организации ядерных исследований (CERN), в 1989 году создал протокол HTTP, язык HTML, а также первый сервер и навигатор для Всемирной паутины
- В 1994 году учредител Консорциум Всемирной Паутины (World Wide Web Consortium – W3C, [www.w3c.org](http://www.w3c.org)) для создания стандартов веб-технологий и контроля над ними
- Будучи директором этой организации, он активно пропагандирует принципы свободного доступа к информации

# Вопрос для самопроверки

- Имела ли сеть ARPANET центральный управляющий компьютер? (Да/Нет)

# Вопрос для самопроверки

- Имела ли сеть ARPANET центральный управляющий компьютер? (Да/Нет)
- Нет. Сеть ARPANET была целиком децентрализована, таким образом, она не теряла способность передавать информацию в случае выхода из строя некоторых ее узлов.

# Вопрос для самопроверки

- Использовала ли сет ARPA net протокол HTTP? (Да/Нет)

# Вопрос для самопроверки

- Использовала ли сет ARPA net протокол HTTP? (Да/Нет)
- Нет. Для ARPA net было разработано семейство протоколов TCP/IP. Протокол HTTP создал Тим Бернерс-Ли в 1989 году.

# Краткая история ЭВМ

Девяностые годы

# Девяностые годы

- Экспоненциальный рост производительности аппаратных средств
- Распределенные вычисления стали обычным явлением среди домашних ПК, подключенных к Интернет
- MS Windows 95, 98, NT заняли лидирующее место на рынке ПО для ПК



# Объектно-ориентированные языки программирования

- C++ – расширение языка C, разработанное Бьярне Страуструпом (Bell Labs) в начале 1980-х годов
- Java – разработан компанией Sun Microsystems в 1995 году. Упрощает переносимость, обладая возможностью исполнения на виртуальной машине
- C# (Си-шарп) – разработан корпорацией Microsoft в 2000 году. Обеспечивает доступ к библиотекам .NET

# Объектно-ориентированное программирование

Опр. Объектно-ориентированное программирование (object-oriented programming, OOP) – концепция программирования, позволяющая разработчикам программного обеспечения быстро создавать сложные программные системы, используя компоненты многоразового использования – объекты, созданные на основе шаблонов – классов.

# Объекты и классы

Опр. Объект (object) – программный компонент многообразного использования, который может моделировать элементы реального мира посредством своих свойств и действий.

Опр. Класс (class) – тип объекта. Определяет атрибуты и методы объекта.

# Объектно-ориентированная операционная система

Опр. Объектно-ориентированная операционная система (object-oriented operating system, OOOS) – операционная система, в которой ресурсы и компоненты представлены в виде объектов. Наследование и интерфейсы помогают создавать модульные операционные системы, которые являются более простыми в обслуживании и расширении. Многие ОС используют объекты, но не многие были полностью написаны на объектно-ориентированных языках.

# Открытое программное обеспечение

Опр. Открытое программное обеспечение (open-source software) – программное обеспечение, к которому прилагается исходный код, и которое обычно распространяется согласно открытому лицензионному соглашению (General Public License, GPL). Такое программное обеспечение, как правило, разрабатывается группами независимых программистов по всему миру.

# Открытое программное обеспечение

- Может быть просмотрено и модифицировано любым членом сообщества
- Существует большая вероятность того, что ошибки будут выявлены и удалены
- ПО может быть модифицировано под специфические потребности отдельных организаций
- Производители могут взыскивать плату за предоставление ПО с GPL-лицензией, но не могут препятствовать дальнейшему распространению открытого ПО

# Открытое программное обеспечение

- GNU – проект свободного распространения программного обеспечения ([www.gnu.org](http://www.gnu.org))
  - Запущен Ричардом Столлменом в 1980-х годах
  - Был ориентирован на создание открытой операционной системы с возможностями и инструментарием UNIX
- OSI – инициативная группа по распространению открытого программного обеспечения (Open Source Initiative) – организация, которая занимается поддержкой и популяризацией открытого ПО ([www.opensource.com](http://www.opensource.com))

# Линус Торвальдс

- Родился в 1969 году в Хельсинки.
- В 1991 году завершил первую версию основного ядра Linux, которое работало на процессоре Intel 80386.
- Он бесплатно использовал многие из средств, опубликованных в рамках проекта свободного распространения программного обеспечения (GNU)
- В свою очередь он сам распространял Linux согласно открытому лицензионному соглашению (GPL) в качестве открытого кода
- К 1994 году система Linux набрала достаточное количество приложений для того, чтобы стать полнофункциональной и доступной для использования операционной системой, - была выпущена версия 1.0.



# Вопрос для самопроверки

- Верно ли, что многие современные операционные системы являются объектно-ориентированными? (Да/Нет)

# Вопрос для самопроверки

- Верно ли, что многие современные операционные системы являются объектно-ориентированными? (Да/Нет)
- Да. Разработчики ОС могут повторно использовать объекты при разработке новых компонентов. Увеличение модульности благодаря применению объектно-ориентированной технологии способствует поддержке новых архитектур.

# Вопрос для самопроверки

- Верно ли, что открытое лицензионное соглашение позволяет свободно просматривать, модифицировать и распространять открытое программное обеспечение? (Да/Нет)

# Вопрос для самопроверки

- Верно ли, что открытое лицензионное соглашение позволяет свободно просматривать, модифицировать и распространять открытое программное обеспечение? (Да/Нет)
- Да. Конечные пользователи могут свободно модифицировать и распространять любое ПО, удовлетворяющее GPL. ПО, распространяемое согласно этой лицензии, должно содержать полный исходный код, сведения о всех его модификациях и сопровождаться общедоступной лицензией.

# Краткая история ЭВМ

2000-ные годы

# 2000-ные годы

- Производительность процессоров отстает от экспоненциального роста числа транзисторов в них
- Многопроцессорные системы
  - Мультипроцессоры (специальные ОС)
  - Многомашинные системы (несколько одинаковых ОС)
  - Распределенные системы (разные ОС на разных машинах)
- Обработка данных на мобильных устройствах

# Стандартизация интерфейсов пользователей и приложений

- Windows XP объединила потребительскую и профессиональные линии OS Windows (Windows Me и Windows 2000)
- Linux и другие открытые ОС используются шире и применяют стандартные программные интерфейсы приложений (напр., POSIX) для повышения совместимости с другими ОС на базе UNIX

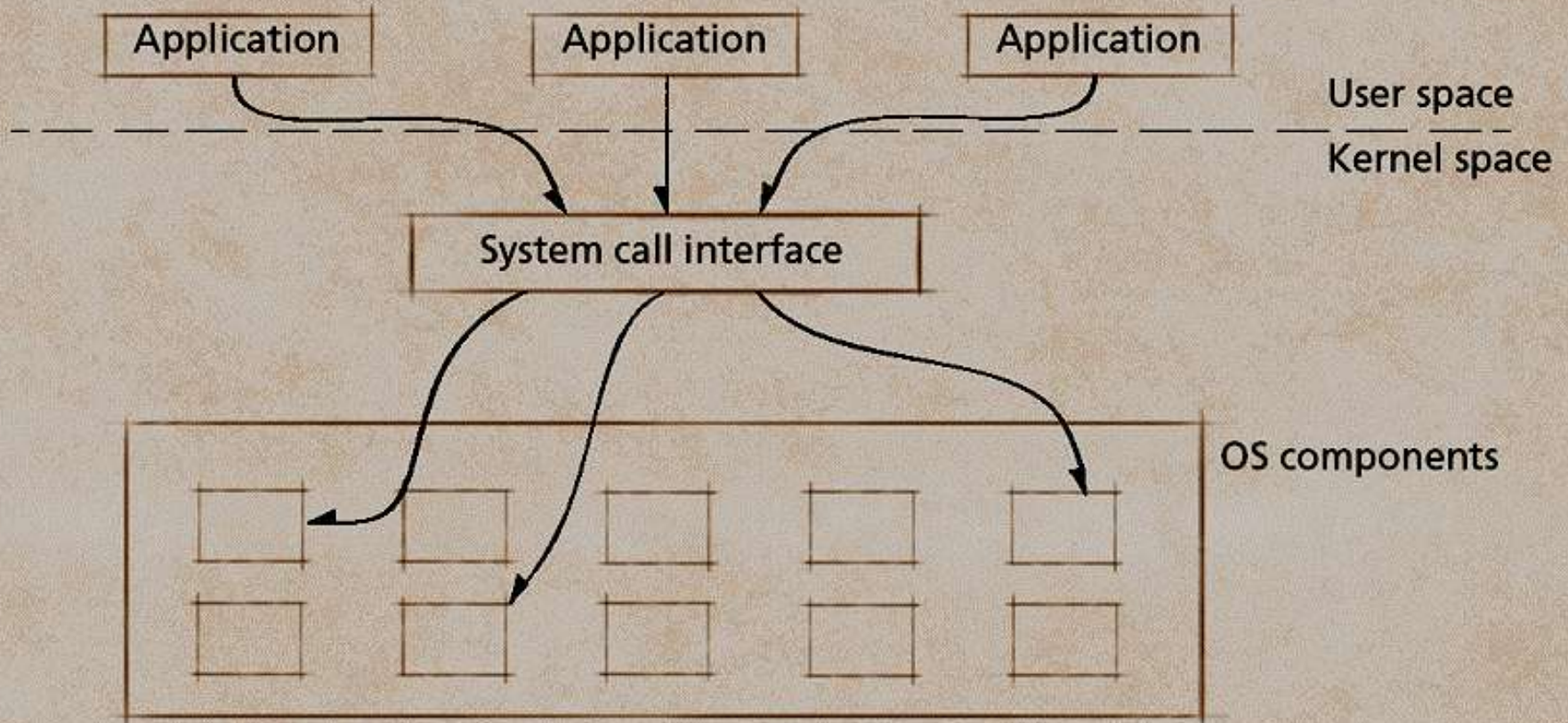
# Программный интерфейс приложений

Опр. Программный интерфейс приложений (Application Programming Interface, API) – спецификация, которая позволяет приложениям запрашивать услуги у ядра посредством системных вызовов (System call interface).

Пр. Интерфейс переносимых операционных систем (Portable Operating System Interface, POSIX) – программный интерфейс приложений, первоначально разработанный для ранней ОС Unix. Применяется для совместимости ОС, основанных на Unix.



# Взаимодействие приложений с операционной системой



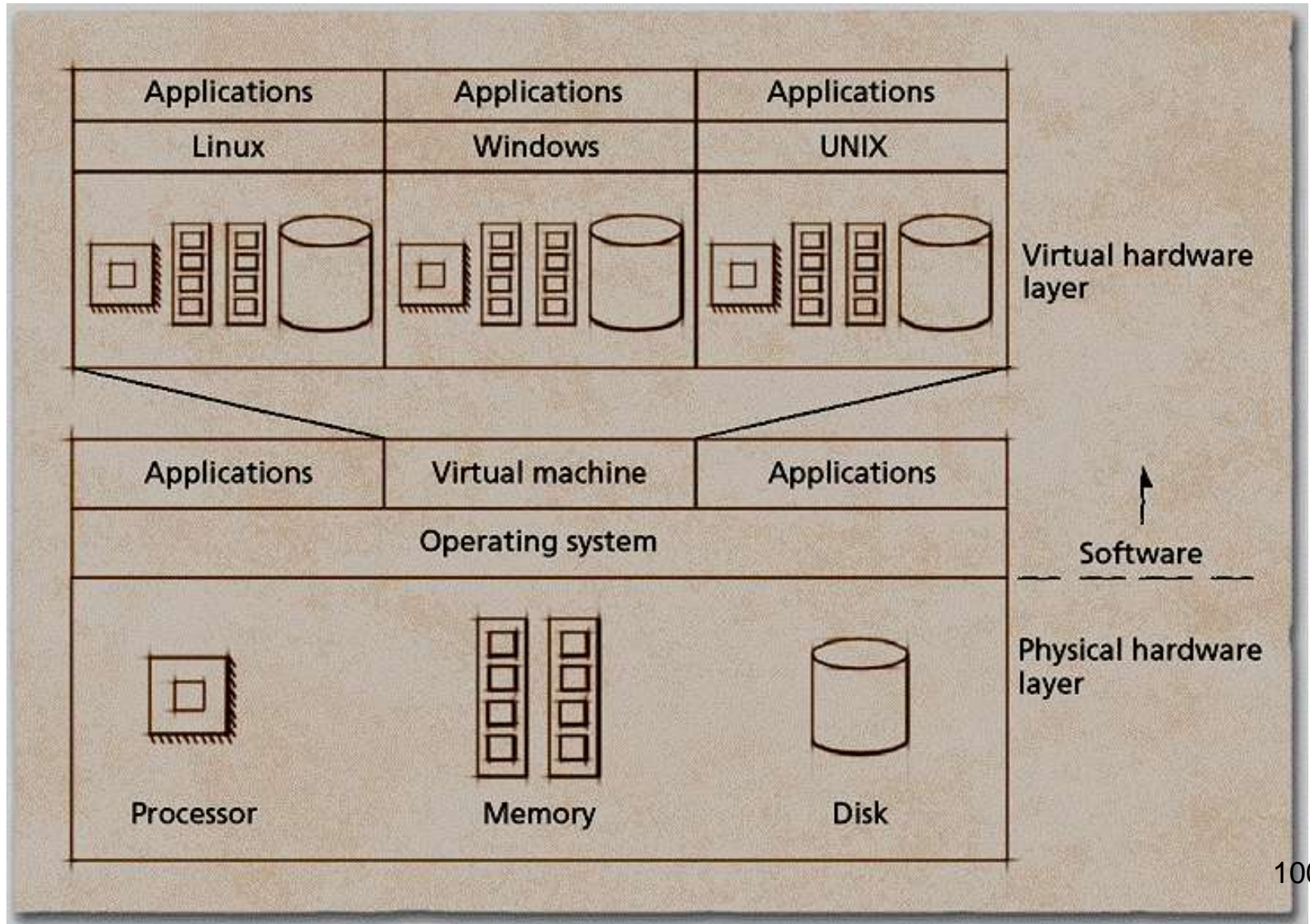
# Пространство пользователя и пространство ядра

- В пространстве пользователя (User space) показаны компоненты программного обеспечения, не обладающие непосредственным доступом к физическим ресурсам системы
- В пространстве ядра (Kernel space) показаны компоненты ОС, обладающие неограниченным доступом к системным ресурсам

# Виртуальная машина

Опр. Виртуальная машина (virtual machine) – приложение, эмулирующее функциональные возможности компьютерной системы.

# Схема виртуальной машины





# Виртуальные машины

- Дают многочисленным пользователям возможность работы с аппаратными средствами в режиме коллективного пользования при иллюзии непосредственного доступа к ресурсам базового компьютера
- Эта концепция впервые была реализована в системе VM/370 на IBM 370 и до сих пор широко используется на еще оставшихся мэйнфреймах

# Виртуальные машины

- Могут выполнять приложения, не полностью совместимые с физической системой, на которой функционирует виртуальная машина
- Например, виртуальные машины DOS в Windows
- Повышают мобильность программного обеспечения, т.е. возможность функционировать на различных платформах
- Например, виртуальные машины Java

# Виртуальная машина Java

Опр. Виртуальная машина Java (Java Virtual Machine, JVM) – виртуальная машина, которая позволяет выполнять программы, написанные на языке Java, на различных платформах, не компилируя заново эти программы на машинный язык компьютеров, где они выполняются.

# Виртуальная машина Java

- Повышает мобильность приложений, т.е. возможность программного обеспечения функционировать на различных платформах
- Упрощает программирование, освобождая программиста от необходимости изучать архитектурные спецификации различных платформ



# Вопрос для самопроверки

- Верно ли, что виртуальные машины в большинстве случаев обладают более низкой производительностью, чем реальные машины?  
(Да/Нет)

# Вопрос для самопроверки

- Верно ли, что виртуальные машины в большинстве случаев обладают более низкой производительностью, чем реальные машины? (Да/Нет)
- Да. Виртуальные машины, как правило, не обладают столь высокой производительностью, как реальные машины, поскольку им приходится выполнять программные действия, эмулирующие работу аппаратных средств. Зато они способствуют повышению мобильности программ, позволяя им функционировать на различных платформах.

# Вопрос для самопроверки

- Оказывает ли распределенная обработка данных помощь вычислениям, выполняемым на мобильных устройствах? (Да/Нет)

# Вопрос для самопроверки

- Оказывает ли распределенная обработка данных помощь вычислениям, выполняемым на мобильных устройствах? (Да/Нет)
- Да. Распределенная обработка данных позволяет мобильным устройствам передавать задачи другим машинам, обладающим большим запасом ресурсов. Мобильное устройство, имея ограниченные ресурсы и срок службы аккумуляторной батареи, может запрашивать данные и вычислительную мощность по сети у более мощного компьютера.