

# **Особенности ОС для различных классов компьютерных систем**

# Особенности ОС для персональных компьютеров (desktops, PCs)

- *Персональные компьютеры* – предназначены как правило, для одного пользователя , однако в ОС необходима поддержка мультипрограммирования и деления времени (для параллельного выполнения нескольких заданий и работы в локальной сети).
- *Устройства ввода-вывода* – клавиатура, мышь, монитор, принтер, сканер, внешние накопители (flash, ZIV и др.), CD-ROM/CD-RW/DVD-ROM/DVD-RW/DVD-RAM/BluRay; цифровые камеры и др.
- Удобство для пользователя, дружелюбность к пользователю (пользовательский интерфейс типа multi-touch, Tablet PC, речевой ввод-вывод и др.)
- Могут использовать технологии, применяемые в “больших” ОС; пользователь имеет персональный доступ к компьютеру и часто не нуждается в оптимизации работы процессора или улучшенных средствах защиты
- На одном и том же ПК могут использоваться несколько ОС (Windows, MacOS, UNIX, Linux) – *double bootable systems*
- ОС для ПК имеет сетевые возможности для соединения в сеть нескольких ПК, а также соединения ПК с серверами и с мобильными устройствами

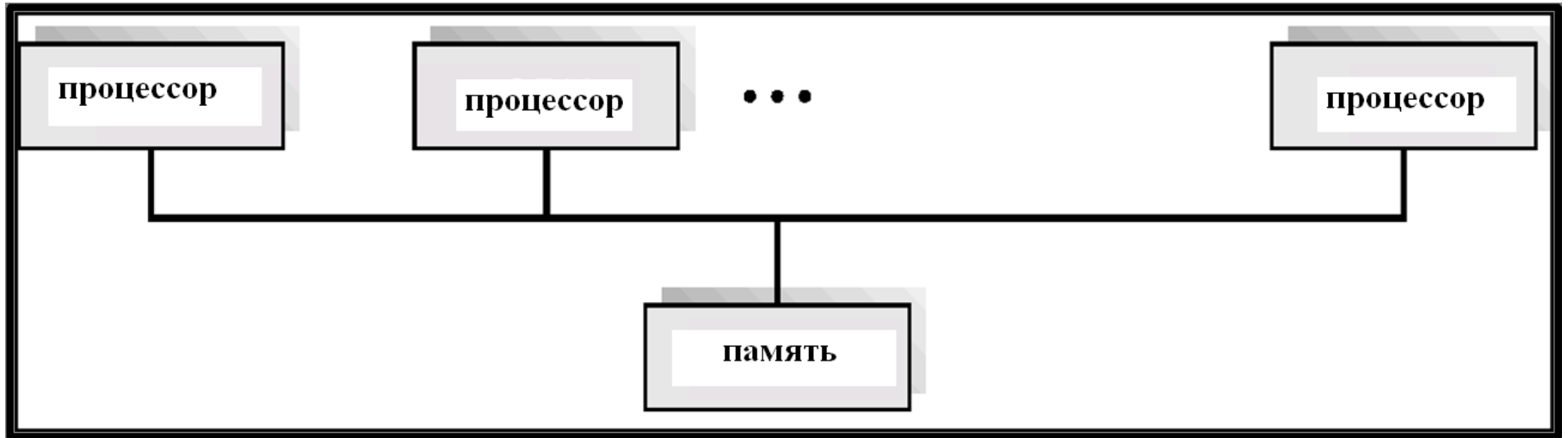
# Параллельные компьютерные системы и особенности их ОС

- Мультипроцессорные системы с несколькими непосредственно взаимодействующими процессорами (CRAY, Эльбрус, позднее – мультипроцессорные рабочие станции и др.)
- *Тесно связанные (tightly coupled) системы* – процессоры разделяют общую память и таймер (такты); взаимодействие происходит через общую память.
- Very Long Instruction Word (VLIW), Explicit Parallelism Instruction Computer (EPIC) системы (см. лекцию 2)
- Multi-core computers ~ еще более тесно связанные процессоры; находятся в одном кристалле, разделяют кэш второго уровня, работают на общей памяти
- Преимущества параллельной компьютерной системы:
  - Улучшенная производительность (*throughput*)
  - Экономичность
  - Повышенная надежность:
    - “дружественное” к пользователю снижение производительности (*graceful degradation*)
    - Устойчивость к ошибкам (*fail-soft systems*)

# Симметричные и асимметричные мультипроцессорные системы

- *Симметричные мультипроцессорные системы - symmetric multiprocessing (SMP)*
  - Все процессоры используют одну и ту же копию ОС, которая может выполняться на *любом* процессоре
  - Любому свободному процессору может быть распределено любое задание
  - Используется общая память и общие дисковые ресурсы
  - Несколько процессов (или потоков) могут исполняться сразу без существенного нарушения производительности
  - Большинство современных ОС поддерживают SMP
- *Асимметричные мультипроцессорные системы (asymmetric multiprocessing)*
  - Каждому процессору дается специфическое задание; главный процессор (master processor) планирует работу подчиненных процессов (slave processors). *За ОС, как правило, закреплен один определенный центральный процессор*
  - Более типично для очень больших систем
  - Пример (Эльбрус): 1-10 ЦП; 1-4 ПВВ (процессоры ввода-вывода); 1-4 ППД (процессоры передачи данных)

# SMP-архитектура



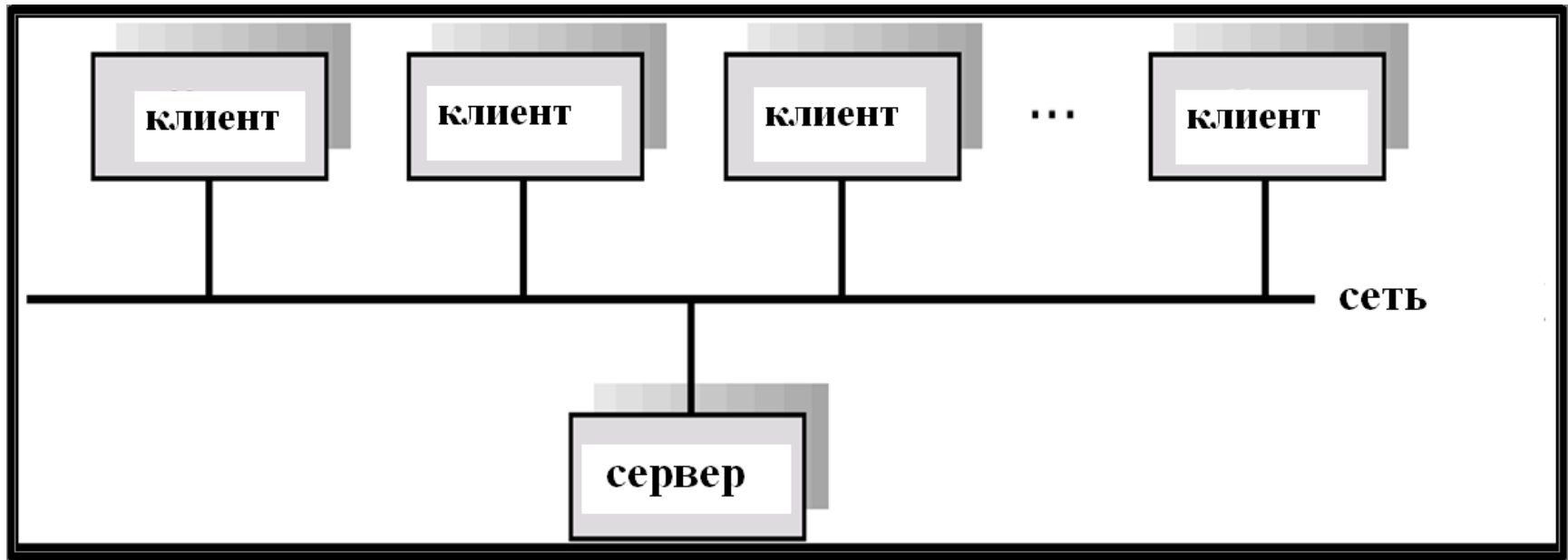
# Распределенные компьютерные системы (distributed systems)

- Распределяют вычисления между несколькими физическими процессорами
- *Слабо связанная система (loosely coupled system)* – каждый процессор имеет свою локальную память; процессоры взаимодействуют между собой через *линии связи* – высокоскоростные *шины, телефонные линии, беспроводную связь (Wi-Fi, EVDO, WWAN/Cingular и др.)*
- Преимущества распределенных систем
  - Разделение ресурсов
  - Совместная загрузка (load sharing)
  - Надежность
  - Связь

# Распределенные компьютерные системы

- Связаны в сетевую инфраструктуру
- Локальные сети (local area networks - LAN) или глобальные сети (wide area networks - WAN)
- Могут быть клиент-серверными (client-server) или одноранговыми (peer-to-peer) системами

# Структура клиент-серверной системы





# Виды серверов в клиент-серверных компьютерных системах

- *Файл-серверы (file servers)* – компьютеры + ПО, предоставляющие доступ к подмножеству своих файловых систем, расположенных на дисках, другим компьютерам локальной сети (LAN). *Пример: SAMBA (SMB – от Server Message Block)* – серверное ПО для ОС типа UNIX (Linux, FreeBSD, Solaris , etc.), обеспечивающее доступ с Windows-компьютеров LAN к файловым системам UNIX-машины. Samba также реализована для платформы Macintosh/MacOS
- *Серверы приложений (application servers)* – компьютеры + ПО, обеспечивающие вычислительные ресурсы для (удаленного) исполнения определенных классов (больших) приложений с других компьютеров LAN. *Примеры: WebSphere (IBM), WebLogic (BEA)* – наилучшие из известных application-серверов для приложений J2EE
- *Серверы баз данных (database servers)* – компьютеры + ПО (Microsoft SQL Server, Oracle, etc.), обеспечивающие доступ другим компьютерам сети к базам данных, расположенным на этих компьютерах
- *Web-серверы (Web servers)* – компьютеры + ПО, обеспечивающие доступ через WWW к Web-страницам, расположенным на этих серверах. *Примеры: Apache; Microsoft.NET Web Servers; Java Web Servers*
- *Proxy-серверы* – компьютеры + ПО, обеспечивающие более эффективное выполнение обращений к Интернету, фильтрацию трафика, защиту от атак
- *Email-серверы* – компьютеры + ПО, обеспечивающие отправку, получение и “раскладку” электронной почты для некоторой локальной сети. Могут обеспечивать также *криптование* почты (email encryption)
- *(Server) back-end* – группа (pool) связанных в LAN компьютеров (вместо одного сервера), обеспечивающая серверные функции

# Кластерные вычислительные системы (clustered systems)

- Компьютеры в кластере, как правило, связаны между собой через быструю локальную сеть
- Кластеризация позволяет двум или более системам использовать общую память
- Обеспечивают высокую надежность
- *Асимметричная кластеризация (asymmetric clustering)*: один сервер выполняет приложение, остальные серверы простаивают
- *Симметричная кластеризация (symmetric clustering)*: все N машин (hosts) исполняют одно приложение
- *Кластеры с высокоскоростным доступом (high-availability clusters, HA)* – обеспечивают оптимальный доступ к ресурсам, предоставляемым компьютерами кластера, например, к базам данных
- *Кластеры с балансировкой загрузки (load-balancing clusters)* – имеют несколько входных балансирующих запросы front-ends, которые распределяют задания между server back-ends (server farm)
- Часто используются в университетах

# Системы реального времени (real-time systems)

- Часто используются как управляющие устройства для специальных приложений, - например, для научных экспериментов; в медицинских системах, связанных с изображениями; системах управления в промышленности; системах отображения (display); системах управления космическими полетами, АЭС и др.
- Четко определенные временные ограничения (время реакции – response time; время наработки на отказ и др.)
- Системы реального времени могут быть *hard* или *soft* real-time
- *Hard real-time systems*: При нарушении временных ограничений может возникнуть критическая ошибка (отказ). *Примеры*: система управления двигателем автомобиля; система управления кардиостимулятором
- *Soft real-time systems*: Нарушение временных ограничений не приводит к отказу. Это системы управления несколькими взаимосвязанными системами для управления совокупностью изменяющихся ситуаций. *Пример*: система планирования рейсов на коммерческих авиалиниях

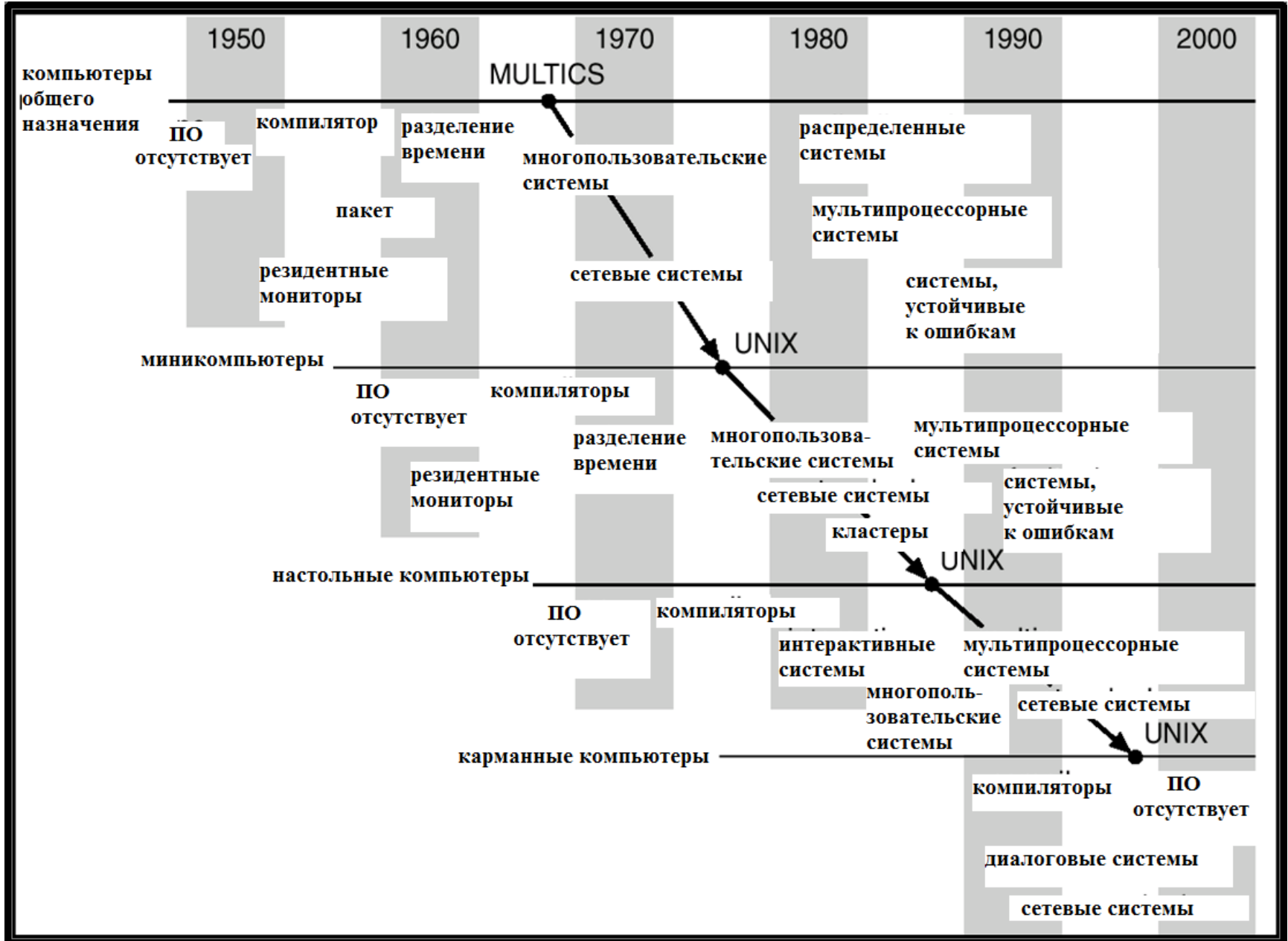
# Системы реального времени: hard real-time и soft real-time

- Hard real-time:
  - Вторичная память ограничена или отсутствует; данные хранятся в оперативной памяти (RAM) или ПЗУ (ROM)
  - Конфликты с системами разделения времени, не имеющие места для ОС общего назначения.
- Soft real-time
  - Ограниченная полезность для промышленных систем управления или в роботике
  - Полезны в приложениях (мультимедиа, виртуальная реальность), требующих развитых возможностей ОС

# Карманные вычислительные системы (handheld systems)

- Карманные персональные компьютеры - Personal Digital Assistants (PDAs)
- Мобильные телефоны – Cellular/mobile phones
- Особенности и проблемы:
  - Ограниченный объем памяти
  - Медленные процессоры (ожидание выполнения простейшей команды в течение нескольких секунд)
  - Маленькие экраны мониторов (дисплеев), отсюда – необходимость в специализированном ПО для поддержки GUI (JME: *javax.microelectronics... /cdui*), не совместимом с обычным (JSE)
  - Невысокая скорость связи через Интернет: GPRS-модем мобильного телефона обеспечивает связь примерно со скоростью dial-up – 3-5 Kbps; обычный мобильный Интернет – 9600 bps
  - Связь для передачи данных – через Bluetooth или IrDA (последний часто отсутствует); имеются не все необходимые порты (нет USB => нельзя использовать flash-память); сменный диск – SmartMedia (как для цифровых фотоаппаратов)
  - Тем не менее, во многие современные коммуникаторы встраивается Wi-Fi

# Развитие концепций и возможностей ОС



# Вычислительные среды

- Традиционные (Traditional computing)
- Ориентированные на WWW (Web-Based Computing)
- Встроенные (Embedded Computing)

# ОС для облачных вычислений

- *Cloud computing* - “облачные” вычисления
- “Облако” (*cloud*) –давно используемая метафора для изображения сервисов, предоставляемых через Интернет или другую коммуникационную сеть (например, через АТМ-сеть)
- *Облачные вычисления* – модель вычислений, основанная на *динамически масштабируемых (scalable)* и *виртуализованных* ресурсах (данных, приложениях, ОС и др.), которые доступны и используются как *сервисы* через Интернет и реализуются с помощью мощных *центров обработки данных (data centers)*
- С точки зрения пользователей, существуют “облака” (общедоступные, частные и т.д.) , предоставляемые различными компаниями, для использования мощных вычислительных ресурсов, которых нет у индивидуального пользователя
- *Недостаток*: пользователь полностью зависит от “облака” и не может управлять даже резервным копированием своих данных и программ
- Наиболее популярная “облачная” платформа – Microsoft Windows Azure (облачная ОС) и Microsoft Azure Services Platform (реализованная на основе Microsoft.NET)
- В настоящее время все крупные компании (Microsoft, IBM, HP, Dell, Sun и др.) разрабатывают свои системы облачных вычислений; имеется тенденция к интеграции этих корпоративных систем в единое доступное пользователю “облако”



**Вопросы?**