**Тема 2. Архитектура операционной системы**

**Лекция 6 «Микроядерная архитектура (модель клиент-сервер)».**

В развитии современных операционных систем наблюдается тенденция в сторону дальнейшего переноса задач из ядра в уровень пользовательских процессов, оставляя минимальное микроядро.

В этой модели вводятся два понятия:

1. серверный процесс (который обрабатывает запросы);

2. клиентский процесс (который посылает запросы).

В задачу ядра входит только управление связью между клиентами и серверами.

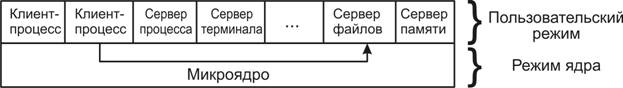


Рис.1. Модель клиент-сервер

**Преимущества:**

- малый код ядра и отдельных подсистем, и как следствие меньшее содержание ошибок.

- ядро лучше защищено от вспомогательных процессов.

- легко адаптируется к использованию в распределенной системе.

**Недостатки:**

- уменьшение производительности.

Современная тенденция в разработке операционных систем - это перенесение значительной части системного кода на уровень пользователя и, следовательно, минимизация ядра. Такой подход к построению ядра называется микроядерной архитектурой операционной системы или моделью клиент-сервер. Суть микроядерной архитектуры - в привилегированном режиме остается работать только очень небольшая часть ОС, называемая микроядром. Микроядро защищено от остальных частей ОС и приложений. В состав микроядра обычно входят машинно-зависимые модули, а также модули, выполняющие базовые функции ядра по управлению процессами, обработке прерываний, управлению виртуальной памятью, пересылке сообщений и управлению устройствами ввода/вывода, которые практически невозможно выполнить в пользовательском режиме. Все остальные функции ядра оформляются в виде приложений, работающих в пользовательском режиме, которые теперь называются серверами ОС. Клиент, которым может быть либо другой компонент ОС, либо прикладная программа, запрашивает сервис, посылая сообщение на сервер. Микроядро, работающее в привилегированном режиме, доставляет сообщение нужному серверу, сервер выполняет операцию, после чего ядро возвращает результаты клиенту с помощью другого сообщения

Поддержка этого механизма является одной из главных задач микроядра.

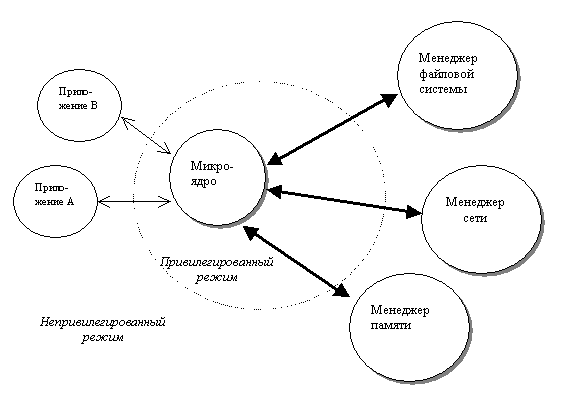


Рис. 2 Микроядерная архитектура операционной системы

Основное достоинство микроядерной архитектуры – высокая степень модульности ядра ОС. Это существенно упрощает добавление в него новых компонентов. В микроядерной операционной системе можно, не прерывая ее работы, загружать и выгружать новые драйверы, файловые системы и т. д. Существенно упрощается процесс отладки компонентов ядра, так как новая версия драйвера может загружаться без перезапуска всей операционной системы. Компоненты ядра операционной системы ничем принципиально не отличаются от пользовательских программ, поэтому для их отладки можно применять обычные средства. Микроядерная архитектура повышает надежность системы, поскольку ошибка на уровне непривилегированной программы менее опасна, чем отказ на уровне режима ядра. В то же время, микроядерная архитектура существенно снижает производительность операционной системы.

Одна из проблем, возникающих при разработке микроядерной ОС – какие функции включать в микроядро, а какие выносить в пользовательское пространство. В идеальном случае микроядро может состоять только из средств передачи сообщений и аппаратно-зависимых модулей (так называемая модель экзоядра). Для повышения производительности ОС в состав микроядра могут входить и другие часто используемые функции. В результате реализации ОС образуют некоторый спектр, на одном краю этого спектра находится операционные системы Workplace OS, 4.4BSD и MkLinux, разработанные на основе микроядра Mach и придерживающиеся чистой микроядерной доктрины. Микроядро обеспечивает управление виртуальной памятью и работу низкоуровневых драйверов. Все остальные функции, в том числе взаимодействие с прикладными программами, осуществляется монолитным ядром. Этот подход возник в результате попыток использовать преимущества микроядерной архитектуры, сохраняя по возможности хорошо отлаженный код монолитного ядра.

На другом краю спектра - Windows NT, в составе которой имеется исполняющая система (NT executive), работающая в режиме ядра и выполняющая функции обеспечения безопасности, ввода-вывода и другие. Компоненты ядра Windows NT располагаются в вытесняемой памяти и взаимодействуют друг с другом путем передачи сообщений, как и положено в микроядерных операционных системах. В тоже время все компоненты ядра работают в одном адресном пространстве и активно используют общие структуры данных, что свойственно операционным системам с монолитным ядром. Кроме того, в Windows NT существует разделение между режимом ядра и режимом пользователя - еще одна черта монолитного ядра.

В настоящее время именно операционные системы, построенные с использованием модели клиент-сервер и концепции микроядра, в наибольшей степени удовлетворяют требованиям, предъявляемым к современным ОС.

Схема смены режимов при выполнении системного вызова в ОС с микроядерной архитектурой выглядит, как показано на [рис.](https://intuit.ru/studies/courses/631/487/lecture/11048?page=4" \l "image.1.8) 3. Из рисунка ясно, что выполнение системного вызова сопровождается четырьмя переключениями режимов (4 t), в то время как в классической архитектуре – двумя. Следовательно, производительность ОС с микроядерной архитектурой при прочих равных условиях будет ниже, чем у ОС с классическим ядром.



**Рис. 3.** Обработка системного вызова в микроядерной архитектуре

В то же время признаны следующие достоинства микроядерной архитектуры:

* единообразные интерфейсы;
* простота расширяемости;
* высокая гибкость;
* возможность переносимости;
* высокая надежность;
* поддержка распределенных систем;
* поддержка объектно-ориентированных ОС.

По многим источникам вопрос масштабов потери производительности в микроядерных ОС является спорным. Многое зависит от размеров и функциональных возможностей микроядра. Избирательное увеличение функциональности микроядра приводит к снижению количества переключений между режимами системы, а также переключений адресных пространств процессов.

Может быть, это покажется парадоксальным, но есть и такой подход к микроядерной ОС, как уменьшение микроядра.

Для возможности представления о размерах микроядер операционных систем в ряде источников приводятся такие данные:

* типичное микроядро первого поколения – 300 Кбайт кода и 140 интерфейсов системных вызовов;
* микроядро ОС L4 (второе поколение) – 12 Кбайт кода и 7 интерфейсов системных вызовов.