

Архитектура операционной системы





- Под архитектурой операционной системы понимают структурную и функциональную организацию ОС на основе некоторой совокупности программных модулей. В состав ОС входят исполняемые и объектные модули стандартных для данной ОС форматов, программные модули специального формата (например, загрузчик ОС, драйверы ввода-вывода), конфигурационные файлы, файлы документации, модули справочной системы и т.д.



- Большинство современных ОС представляют собой хорошо структурированные модульные системы, способные к развитию, расширению и переносу на новые платформы. Какой-либо единой унифицированной архитектуры ОС не существует, но известны универсальные подходы к структурированию ОС.

Принципиально важными универсальными подходами к разработке архитектуры ОС являются



- модульная организация;
- функциональная избыточность;
- функциональная избирательность;
- параметрическая универсальность;
- концепция многоуровневой иерархической вычислительной системы, по которой ОС представляется многослойной структурой;
- разделение модулей на две группы по функциям: ядро – модули, выполняющие основные функции ОС, и модули, выполняющие вспомогательные функции ОС;
- разделение модулей ОС на две группы по размещению в памяти вычислительной системы: резидентные, постоянно находящиеся в оперативной памяти, и транзитные, загружаемые в оперативную память только на время выполнения своих функций;
- реализация двух режимов работы вычислительной системы: привилегированного, или режима супервизора и пользовательского режима или режима задачи ;
- ограничение функций ядра (а следовательно, и количества модулей ядра) до минимального количества необходимых самых важных функций.

Операционная система состоит из четырех основных частей:



- Первая часть — ядро, низкоуровневая основа любой ОС, выполняемая аппаратурой в особом привилегированном режиме. Ядро загружается в память один раз и находится в памяти резидентно — постоянно, по одним и тем же адресам. Ядро — командный интерпретатор, «переводчик» с программного языка на «железный», язык машинных кодов.
- Вторая часть — Подсистема управления ресурсами — управляет вычислительными ресурсами компьютера — оперативной и внешней памятью, процессором
- Третья часть — Управляющая программа — управляет исполнением других программ и функционированием устройств ввода-вывода.(используются специализированные программы для управления различными устройствами, входящими в состав компьютера. Драйвера «системные библиотеки»)
- Четвертая часть — удобная оболочка, с которой общается пользователь — интерфейс.

Архитектура операционной системы — ЯДРО



- Ядро – самый ключевой, основной компонент операционной системы, именно в нем реализуется большая часть функциональности ОС. Появились различные подходы к проектированию и реализации ядра ОС, а именно
 - Монолитное ядро
 - Поуровневый подход
 - Микроядро
 - Модули ядра

Монолитное ядро



- Появилось исторически первым в ходе эволюции операционных систем. ОС используют большое монолитное ядро Монолит – все вместе, все библиотеки, сервисные функции в одном ядре.

Монолитное ядро содержит следующие базовые элементы



Планирование процессов

Управление файловой системой

Сетевое взаимодействие

Драйверы устройств

Управление памятью



❑ Преимущества:

- Производительность — в виду того, что количество переключений из контекста режима пользователя в режим ядра сведено к минимуму;

❑ Недостатки:

- Неустойчивость к сбоям — так как все базовые элементы и их работа выполняются в режиме ядра, и если хотя бы в одном модуле или блоке ядра произойдет какой-либо сбой, то ему будет подвержена вся ОС(все ядро), вариантов других нет, закончится все — перезапуском ОС.

Простой поуровневый подход



Разбиение на уровни помогает разбить программирование операционной системы, становится легче программировать.

Поуровневый подход используется оригинальным ядром Unix:

- Небольшое число монолитных слоев
- Нет инкапсуляции, функции и сервисы, предоставляемые на разных уровнях доступны всей системе
- По сути ядро – набор процедур, которые могут вызывать любые другие процедуры
- Огромное количество функциональности в ядре

МИКРОЯДРО



Микроядро – ядро, содержащее только самые необходимые функции.

Идея: минимизировать само ядро, вынести как можно функциональности в режим пользователя (т.е. исполнять эту функциональность в виде обычных процессов).

Многие сервисы становятся пользовательскими процессами:

- Драйверы устройств,
- Файловые системы,
- Менеджер виртуальной памяти,
- Оконные системы графического интерфейса пользователя,
- Службы безопасности

Модули ядра



- Модуль – нечто среднее между поуровневой архитектурой и микроядром.
- Удобно для разработки (каждый может разрабатывать свой модуль без необходимости трогать основную систему).
- Модули находятся в режиме ядра (уменьшаются затраты на взаимодействие модулей между собой)
- Компромисс ради производительности.
- Под разные задачи необходимо выбирать ОС