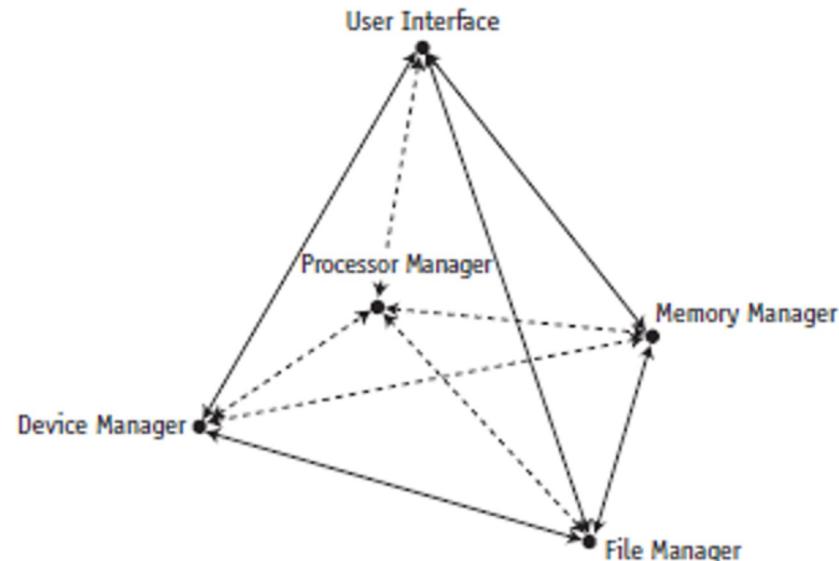


Системная инженерия программного обеспечения

Левицкий Михаил Сергеевич

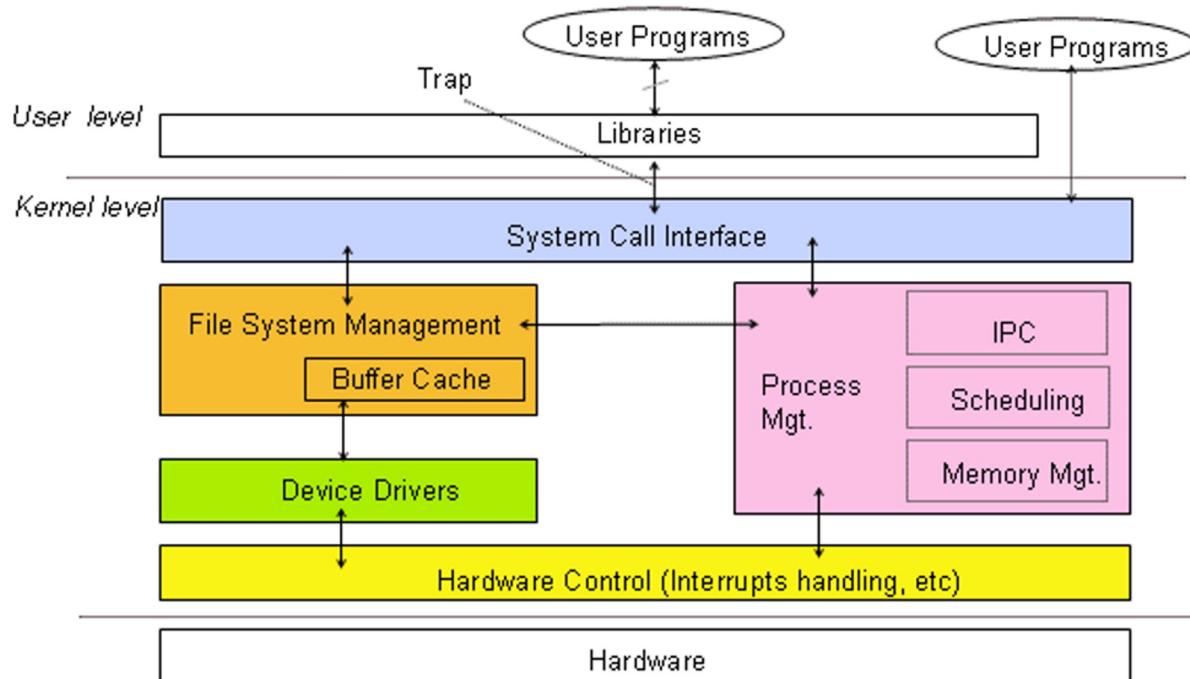
Операционная система

Комплекс взаимосвязанных программ, предназначенные для управления ресурсами компьютера и организации взаимодействия с пользователем.



Ресурсы

- процессор
- память
- storage
- network
- informational(auth)
- другие(мышь, клавиатура)



Объекты операционных систем

- Устройства
- Процессы
- Потоки
- Файлы
- Терминалы
- Объекты синхронизации
- IPC

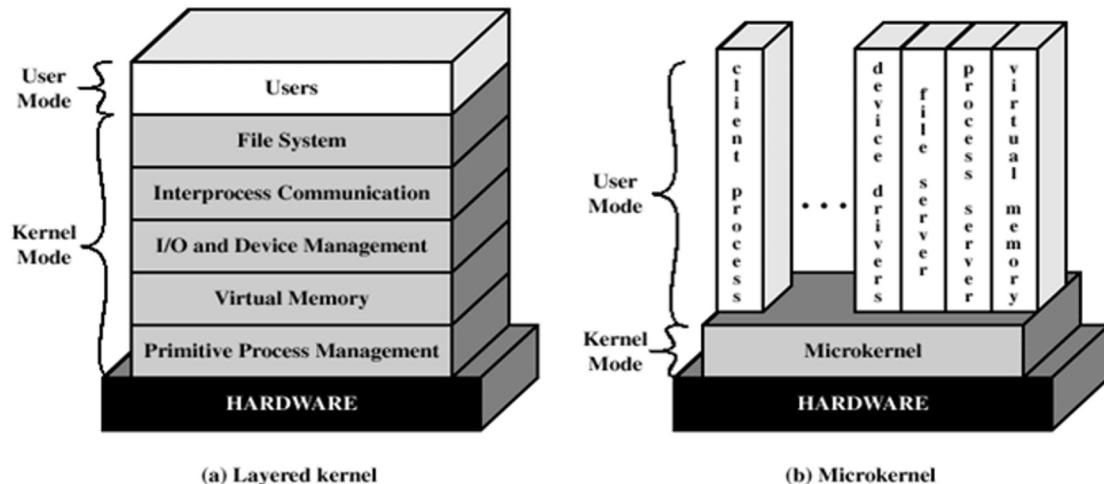
Основные задачи ОС

- Scheduling
- Виртуальная память и подкачка
- Загрузка программ и библиотек
- Разграничение доступа
- Борьба с блокировками доступа к ресурсам
- Поддержка IPC
- Планирование и оптимизация ввода/вывода
- Поддержка файловых систем
- Предоставление API пользовательским приложениям

Основные ОС

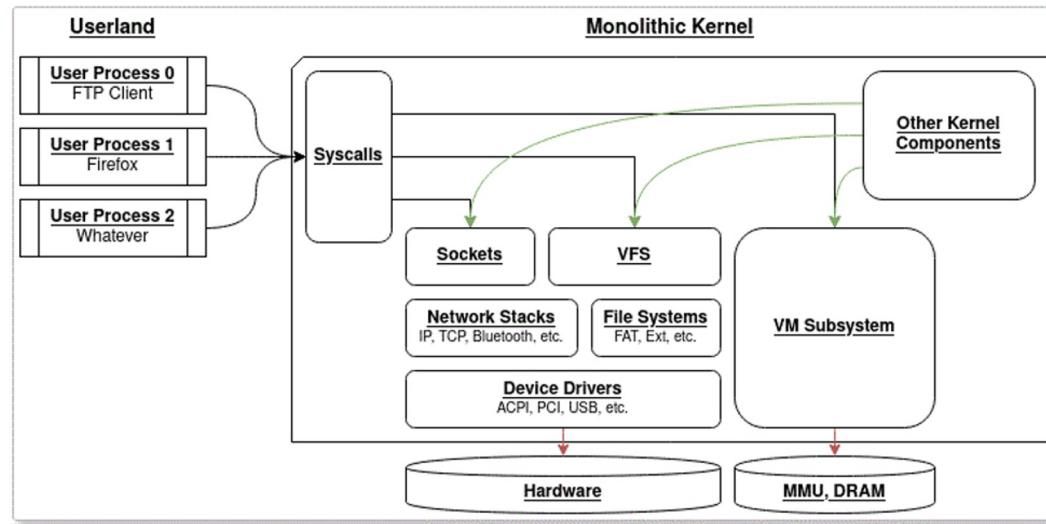
Microkernel (μ -kernel)

- Асинхронные взаимодействия
- Нити в ядре вытесняются другими
- Как правило разные адресные пространства для компонентов ядра



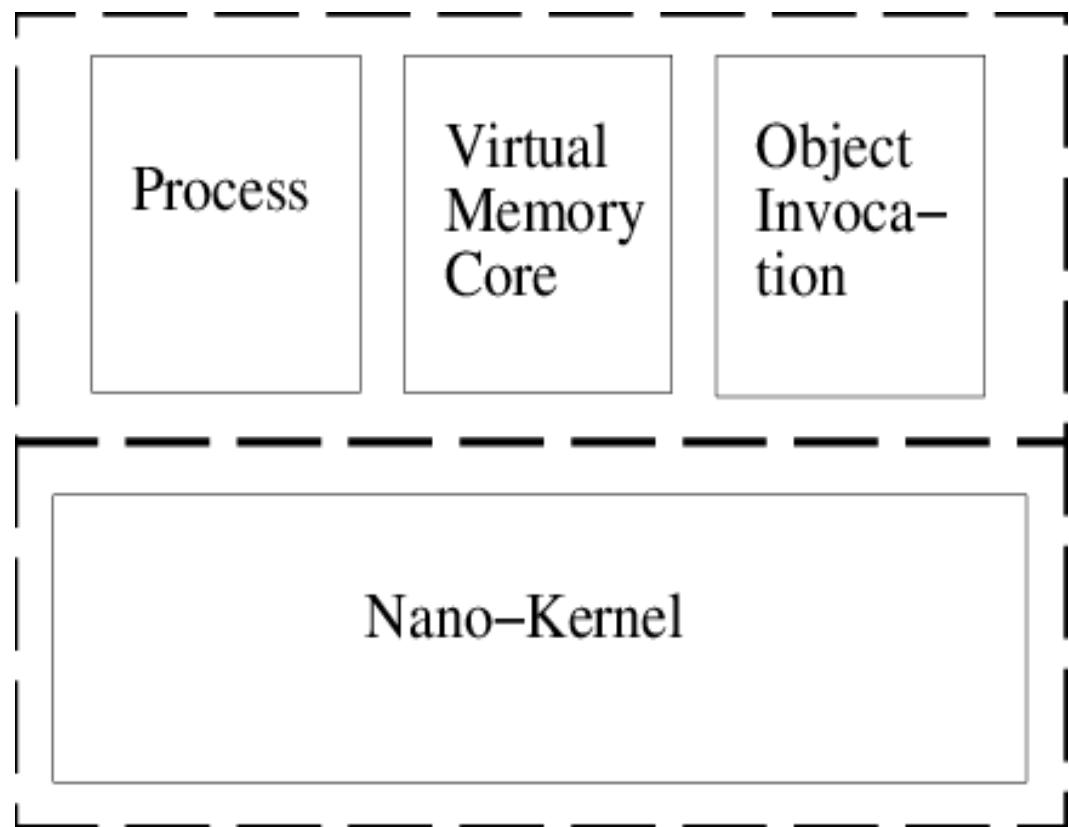
Monolithic kernel

- Синхронные взаимодействия
- Нити в ядре не вытесняются другими
- В микроядре только примитивы
- Одно адресное пространство для компонентов ядра



Nanokernel

Обработка аппаратных прерываний.

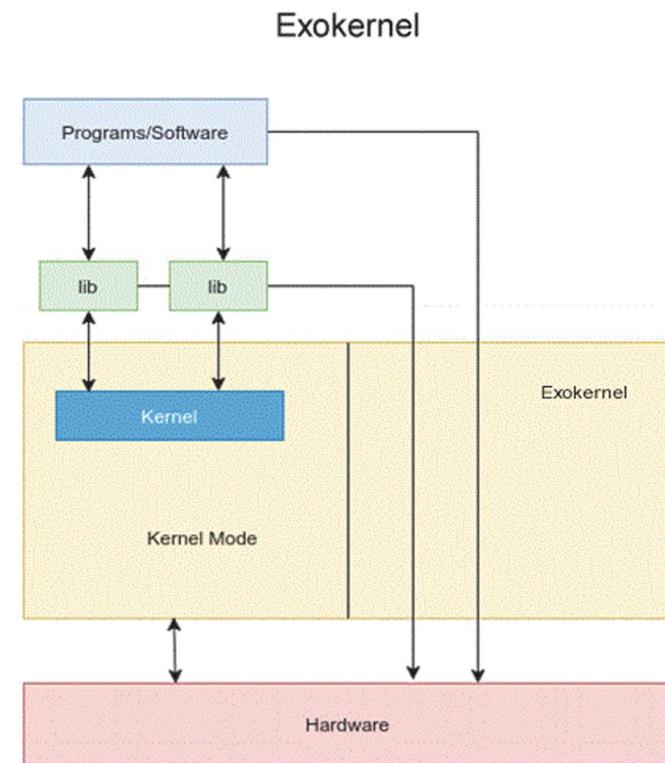


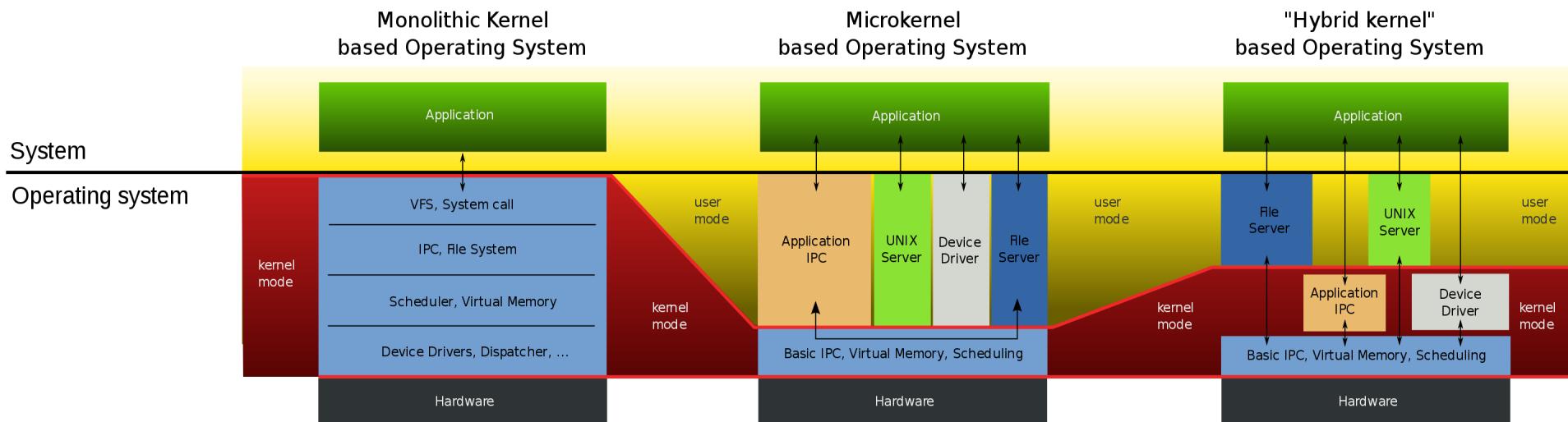
Hybrid kernel

Модифицированные микроядра, позволяющие для ускорения работы запускать модули ОС в пространстве ядра.

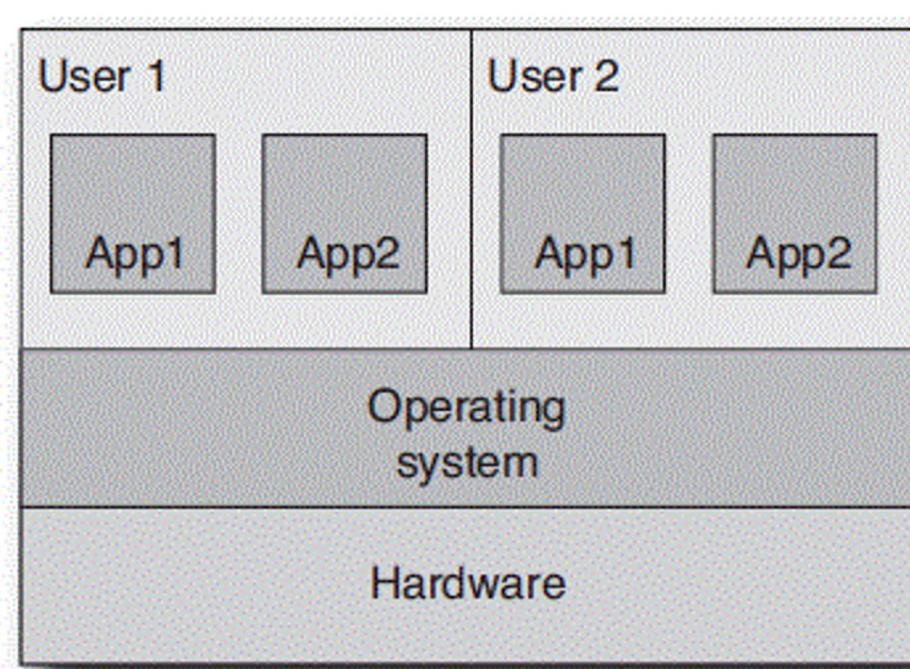
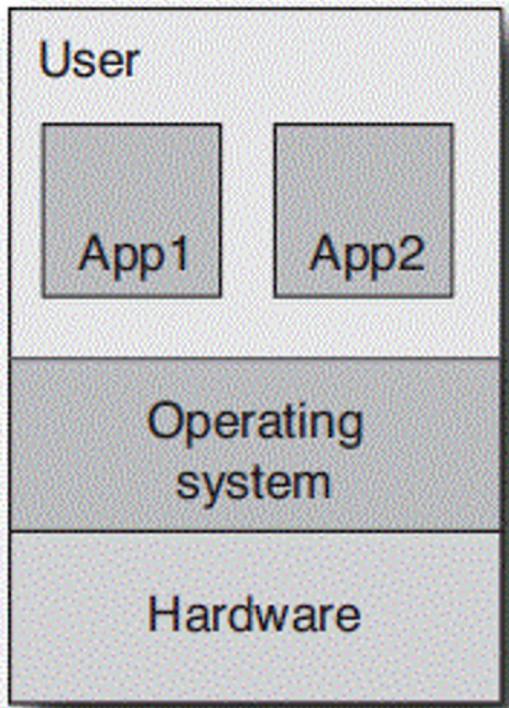
Exokernel

- Нет переключения между процессами при обращении к оборудованию
- Предоставляет набор служб для взаимодействия между приложениями,
- Необходимый минимум функций, связанных с защитой: выделение и высвобождение ресурсов, контроль прав доступа

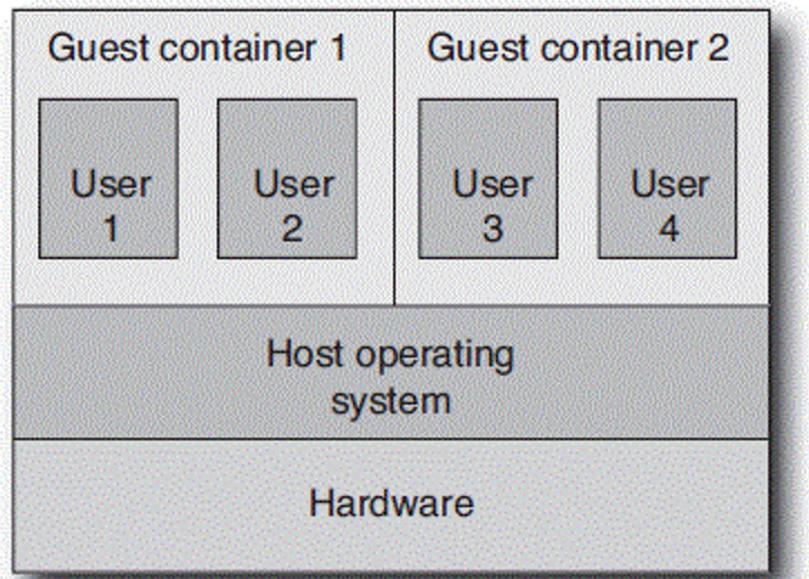




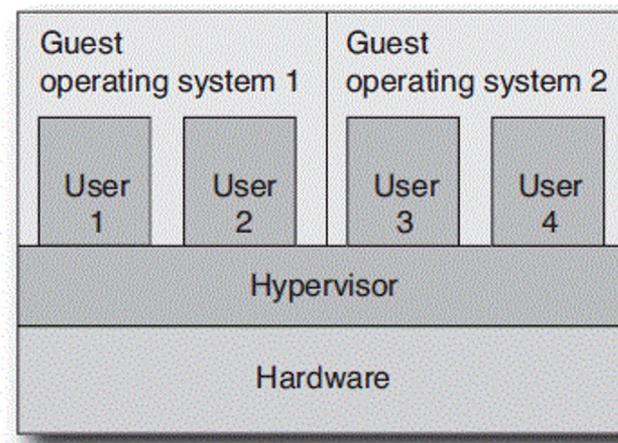
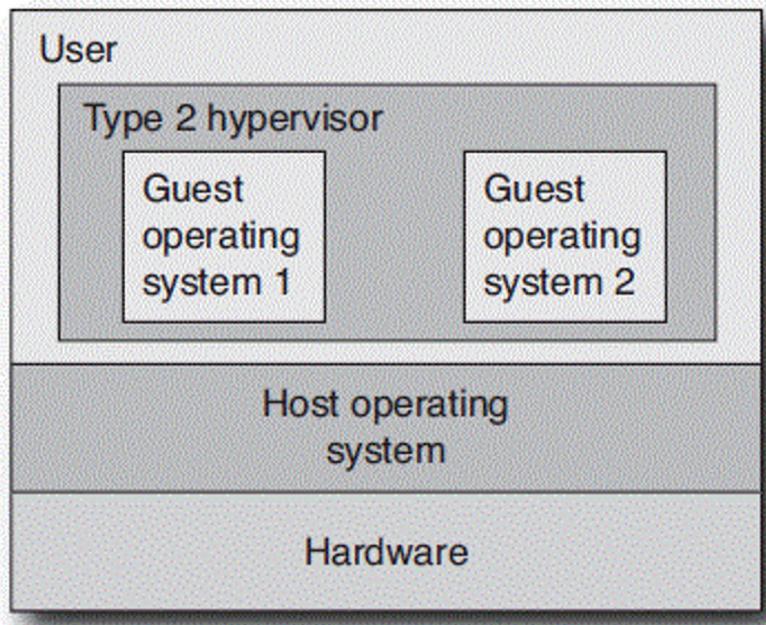
Однопользовательская, многопользовательская



Виртуализация, паравиртуализация



Виртуализация, паравиртуализация

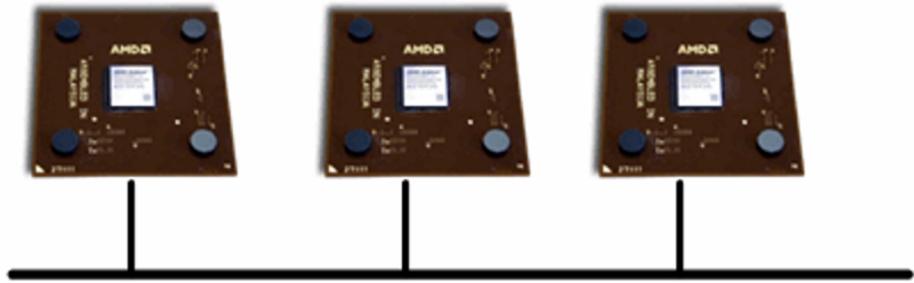


Процессорный ресурс

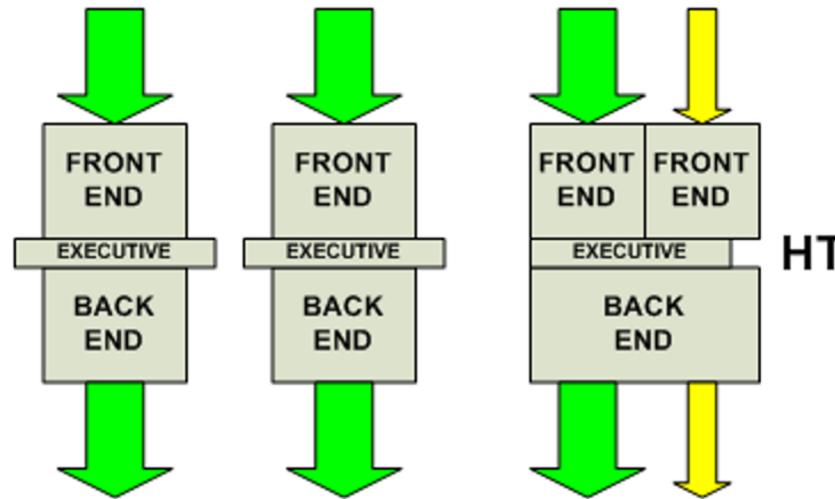
- Один из объектов, управляемых ОС
- Выполняет поток команд(ядро, нить)

Мультипроцессор

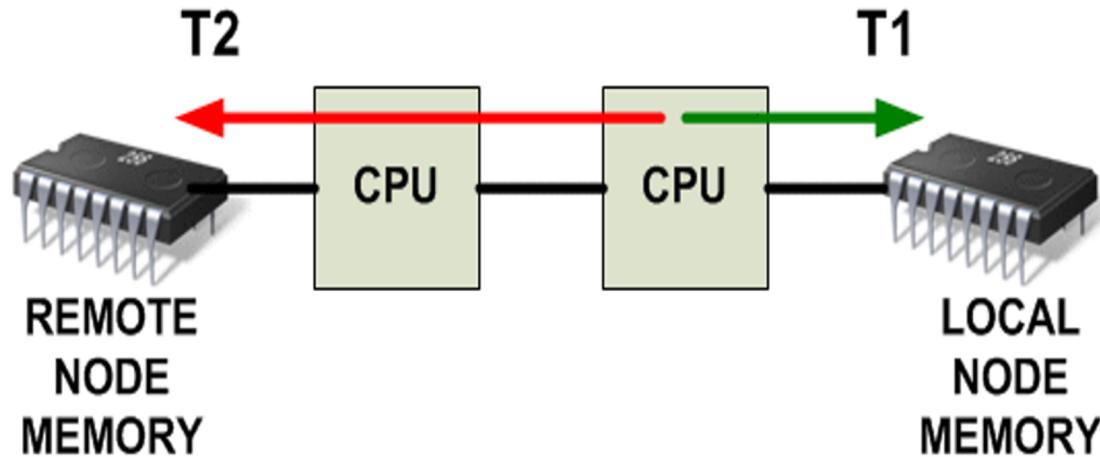
- Одинаковые процессоры сравнимой производительности.
- Общая память



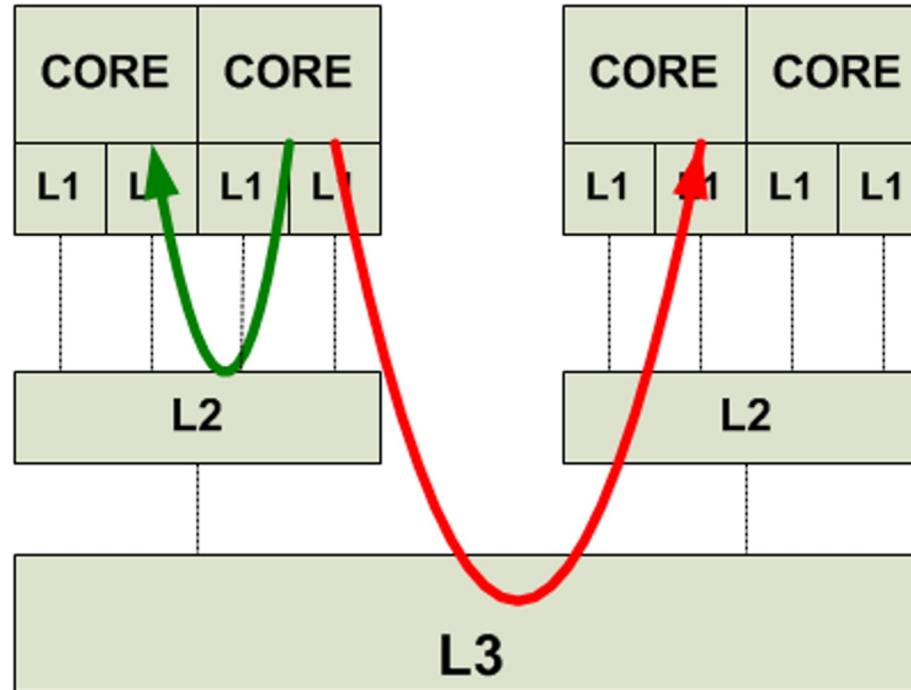
Hyper-threading technology



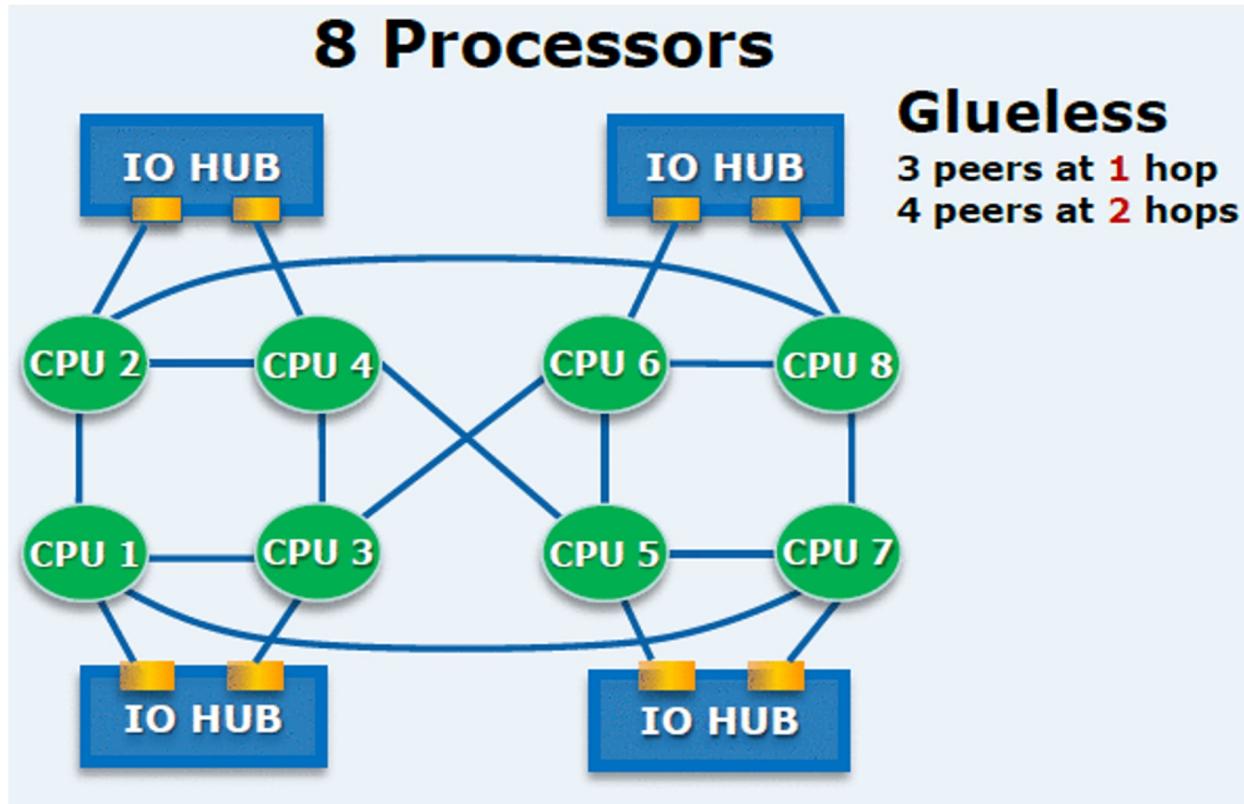
Топология по памяти



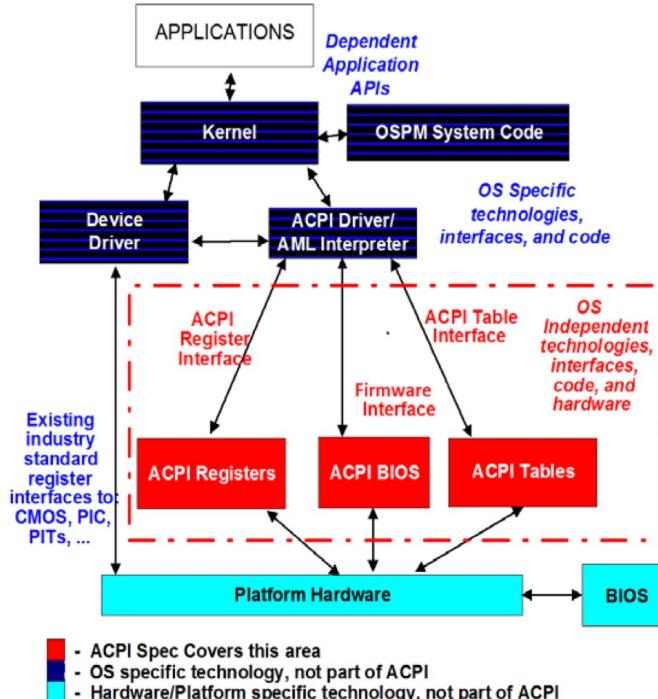
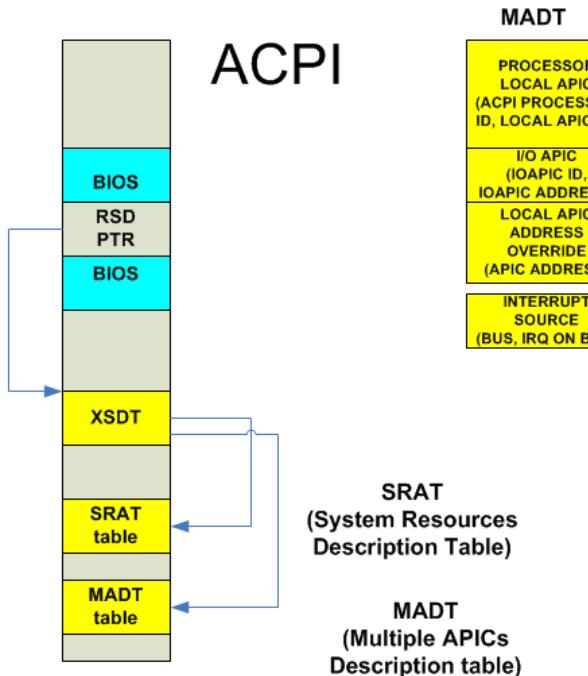
Топология по cache



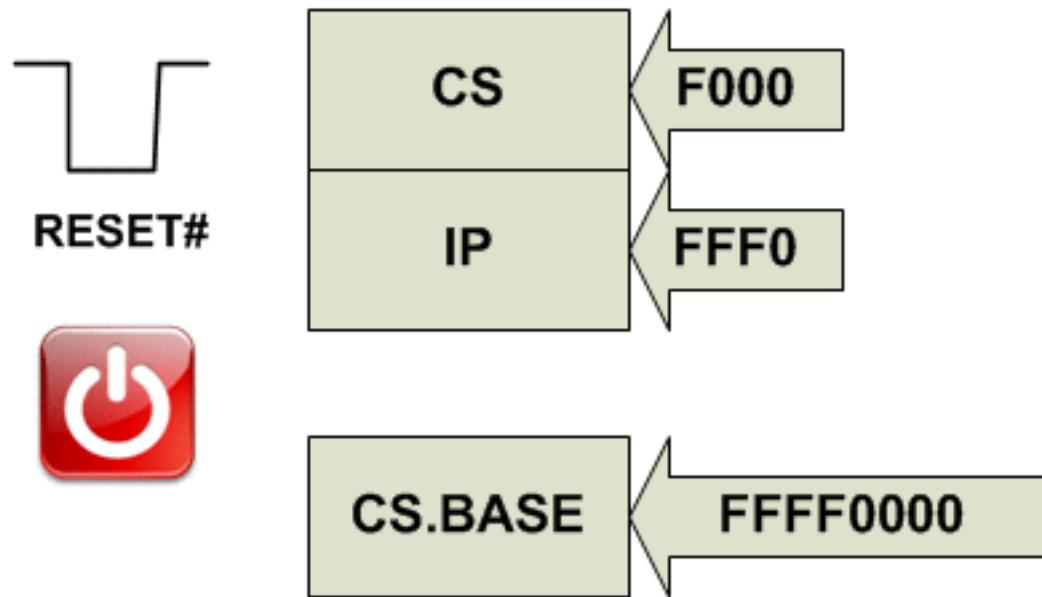
Топология по межпроцессорному соединению



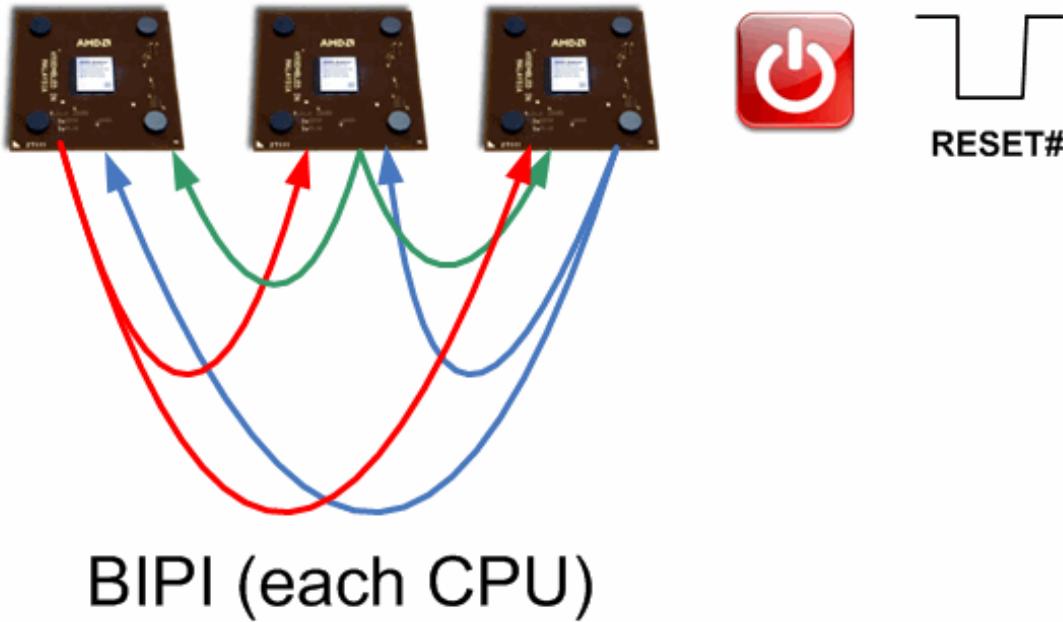
Получение информации о топологии ACPI таблицы



Инициализация процессорных ресурсов(x86-64) - BIST

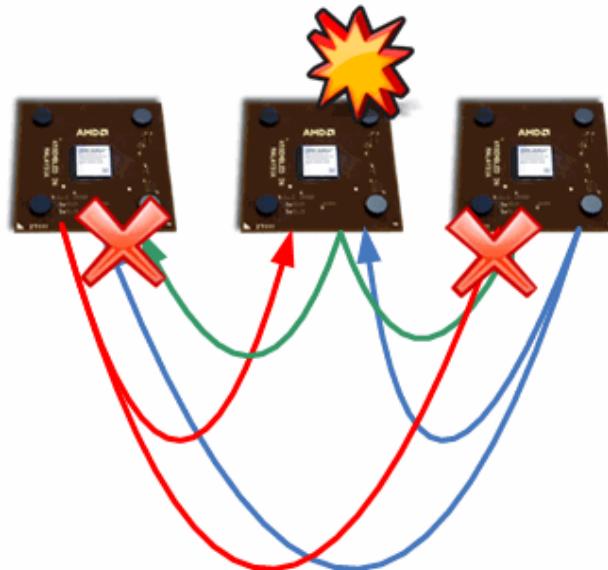


Инициализация процессорных ресурсов (x86-64) - BIPI



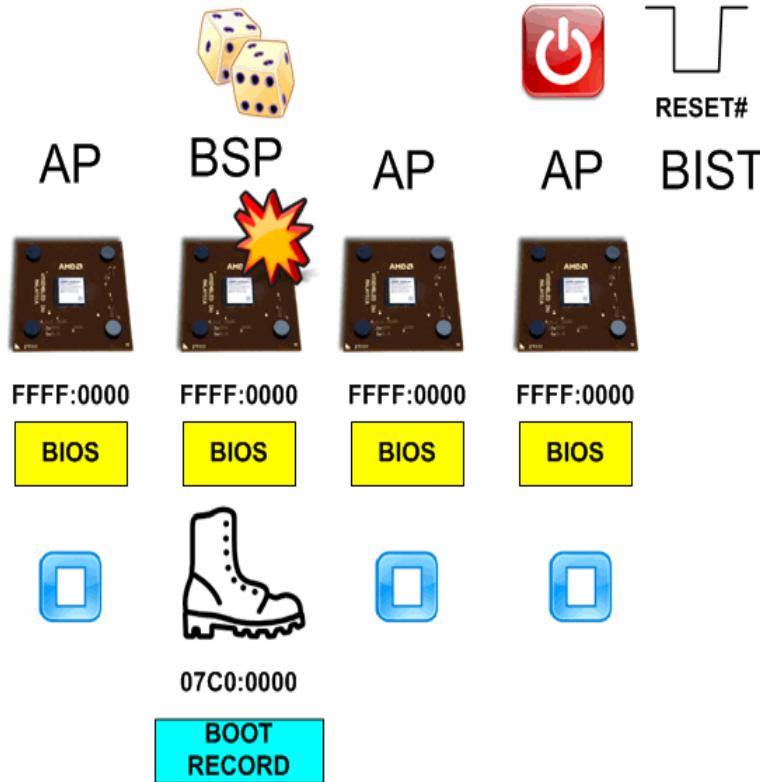
Инициализация процессорных ресурсов (x86-64) -FPII

AP BSP AP

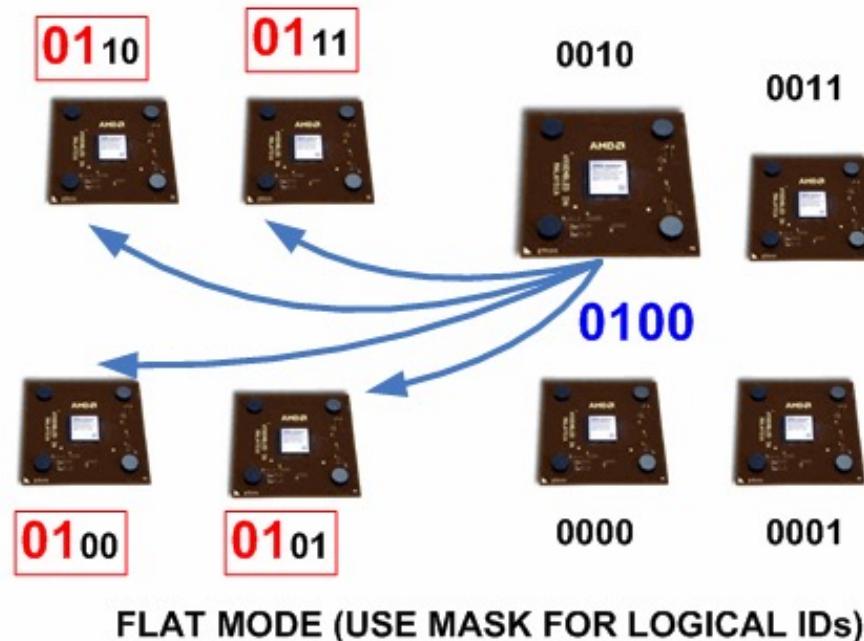


FPII (send each CPU,
receive only BSP)

Инициализация процессорных ресурсов (x86-64) - SIPI



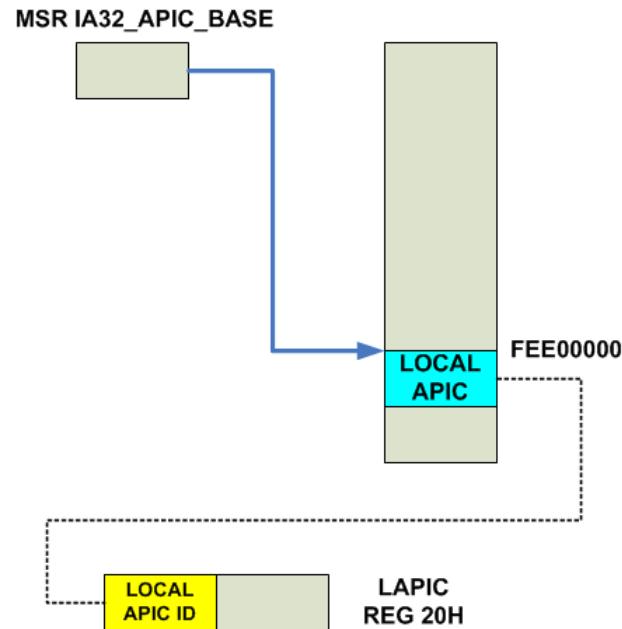
Interprocess interrupt



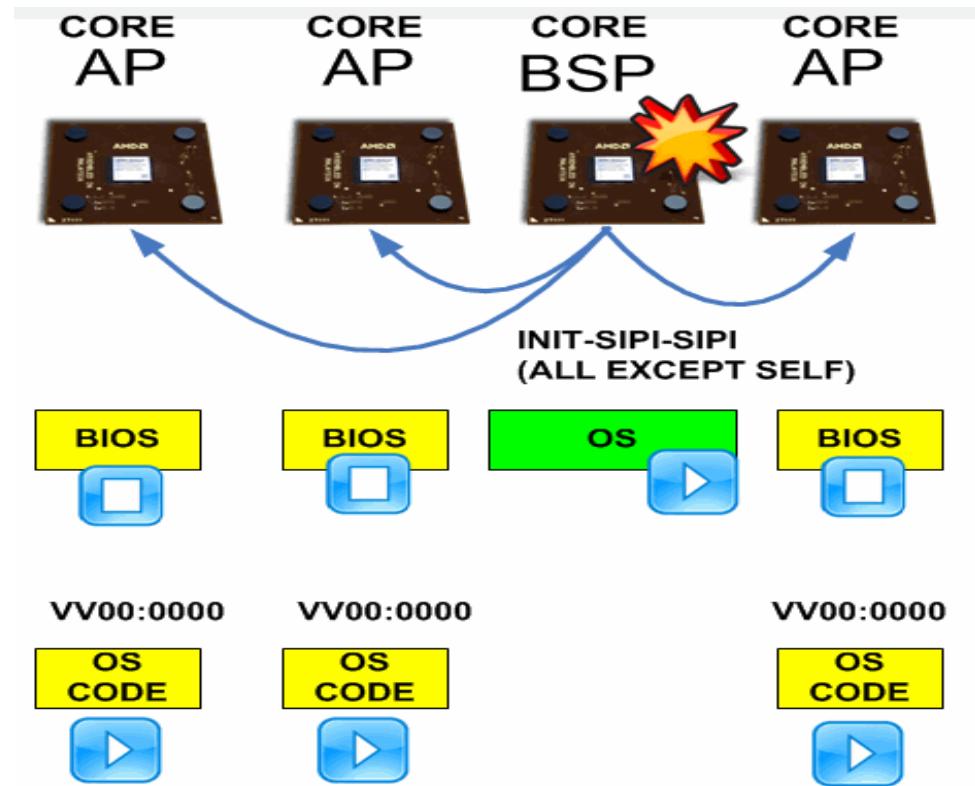
**LOGICAL
GROUP**

FLAT MODE (USE MASK FOR LOGICAL IDs)

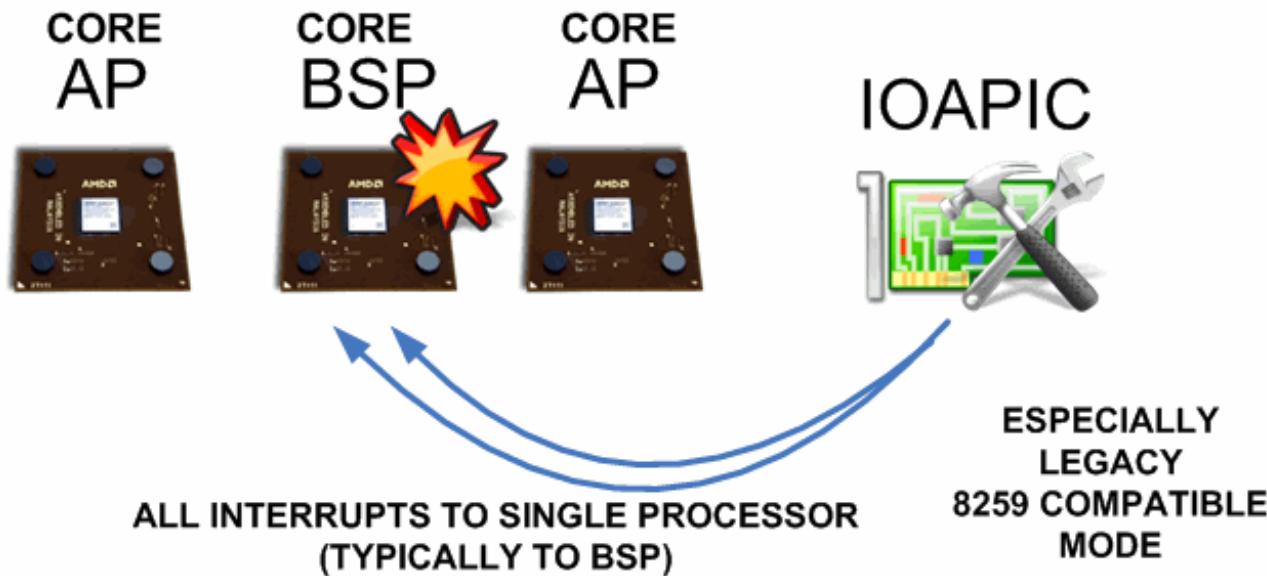
APIC



Запуск Application Processors

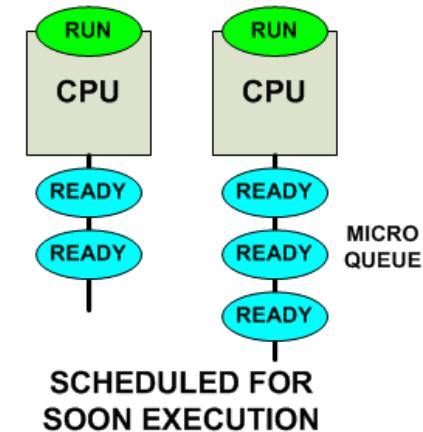
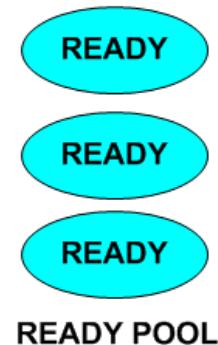


IOAPIC и системные прерывания

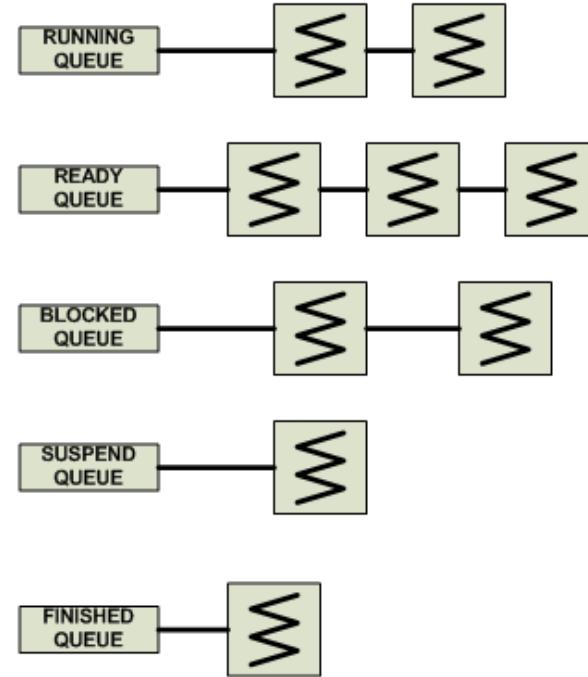


Scheduling

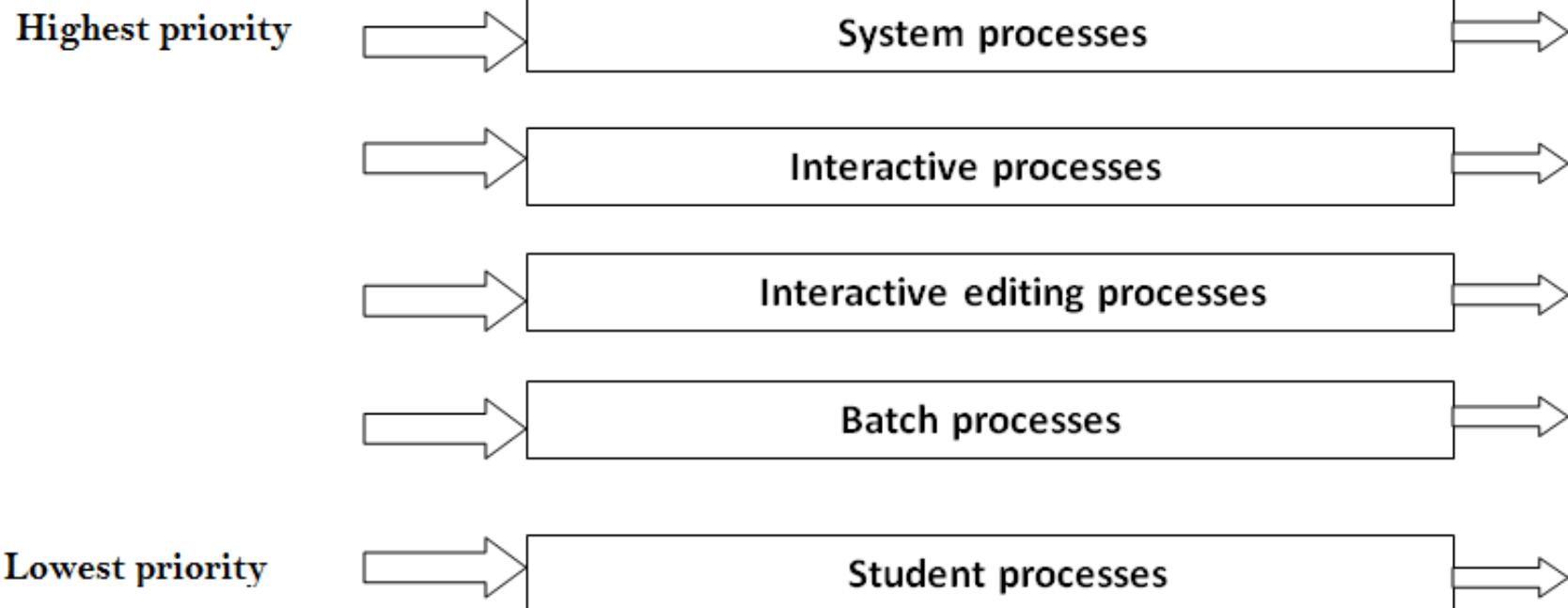
Контекст выполнения(набор
регистров)



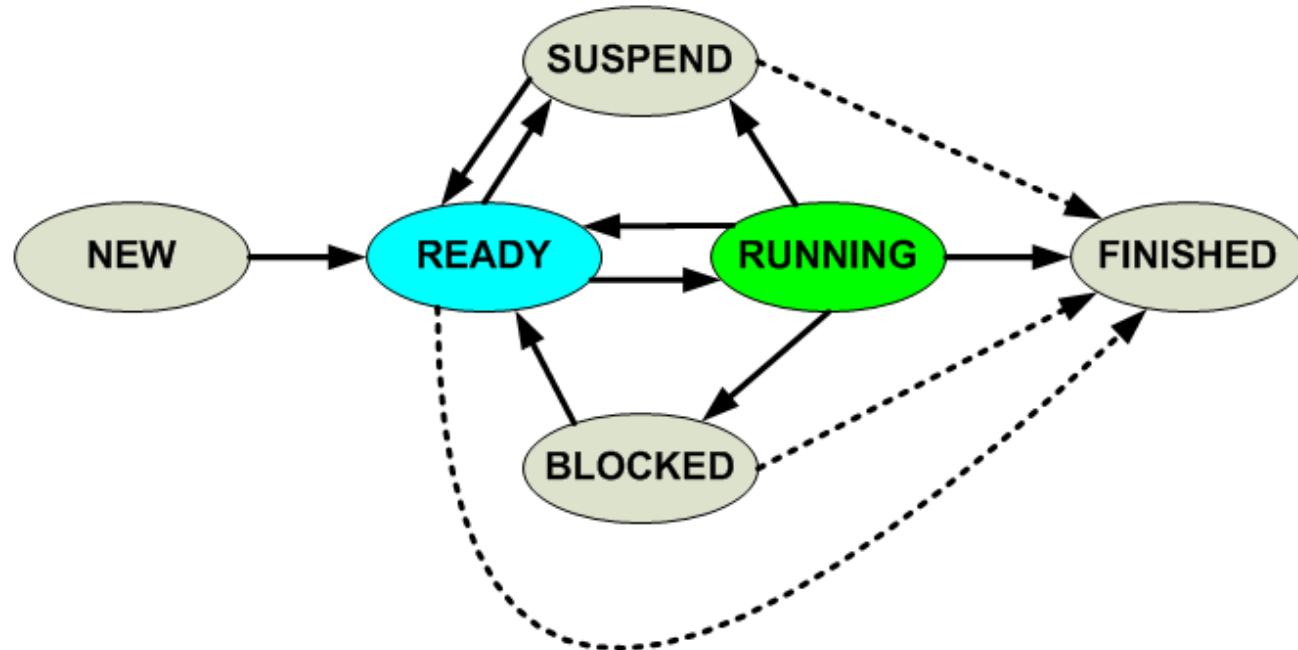
Микроочереди



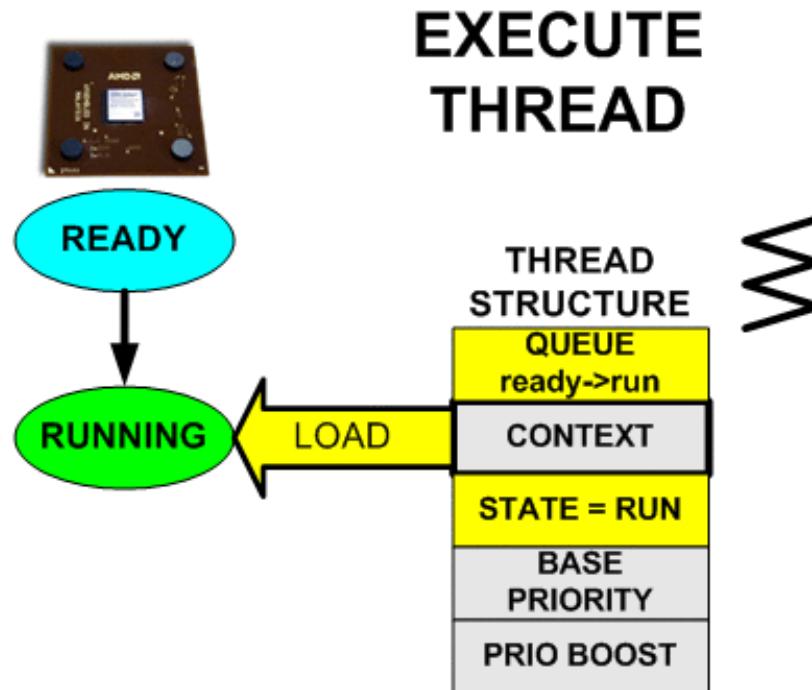
Системные очереди



Состояние нити и переходы

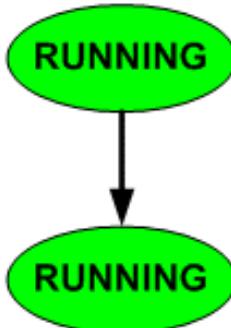


Запуск нити



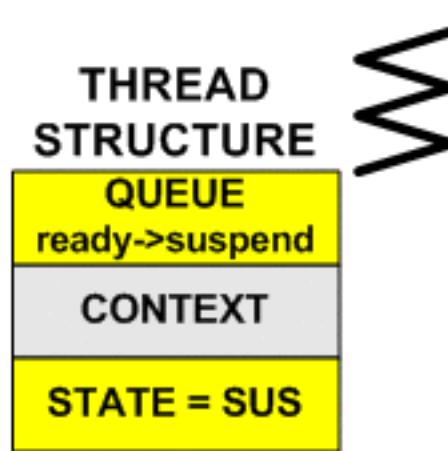
Шедулирование нити

SchedulerTick



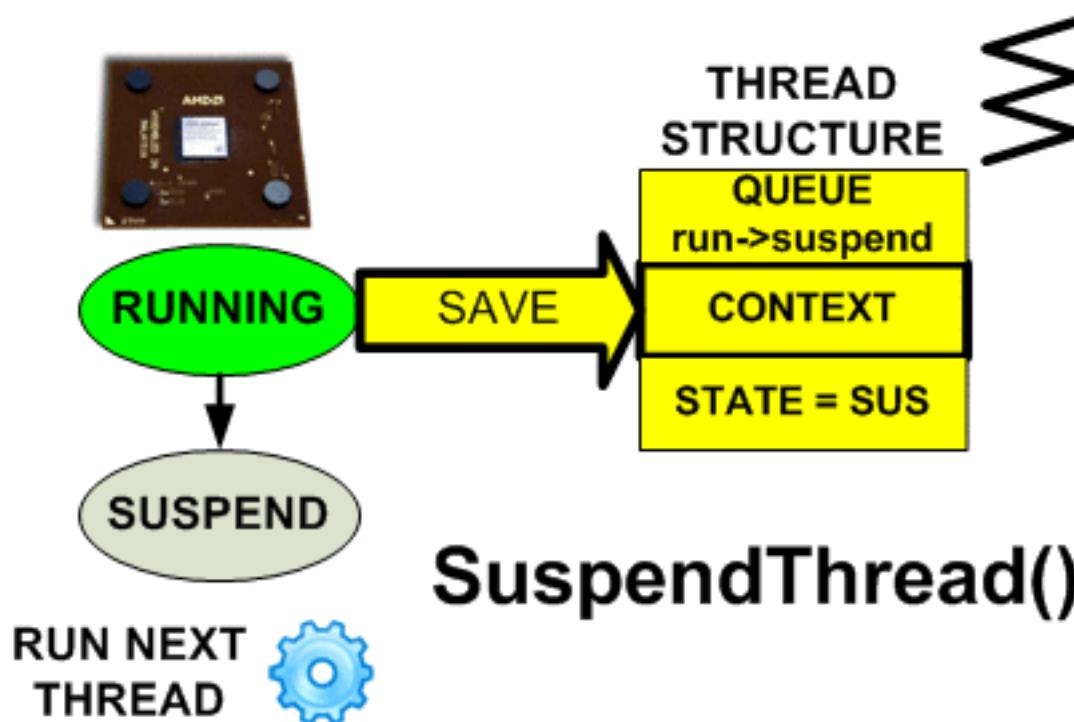
THREAD STRUCTURE
QUEUE run
CONTEXT
STATE = RUN
PRIORITY
PRIO BOOST
CURRENT QUANT--

Suspend нити

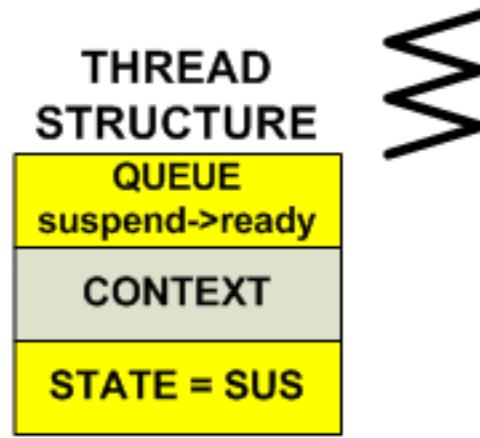


SuspendThread()

Suspend нити



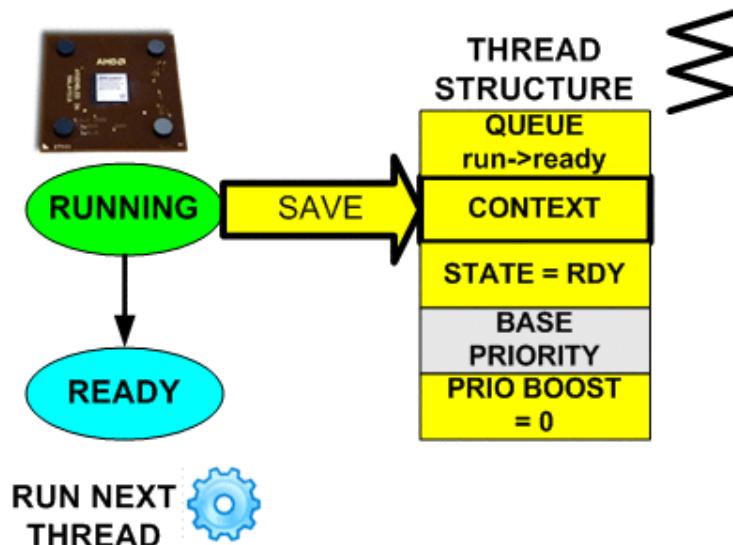
Resume нити



ResumeThread()

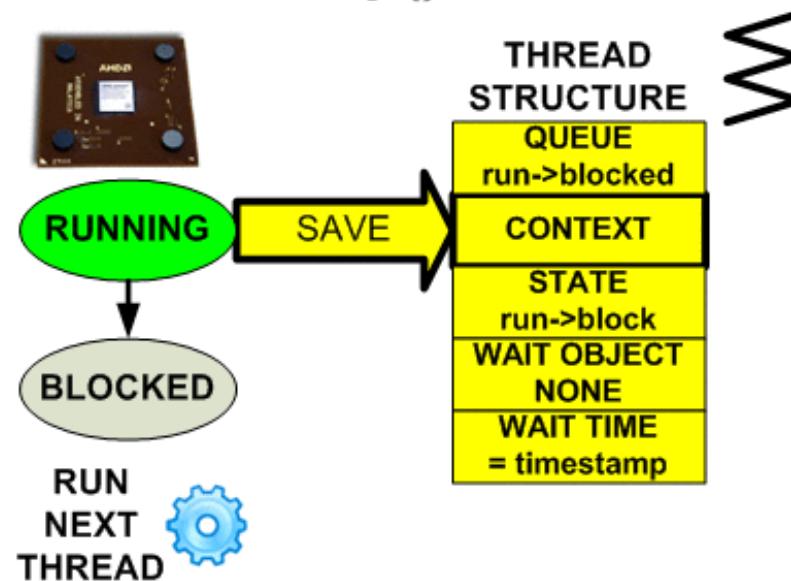
Yield нити

YieldThread()



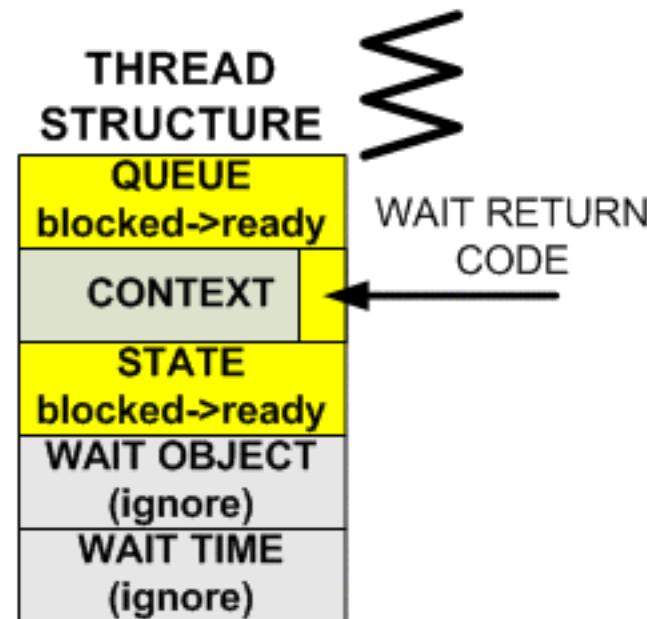
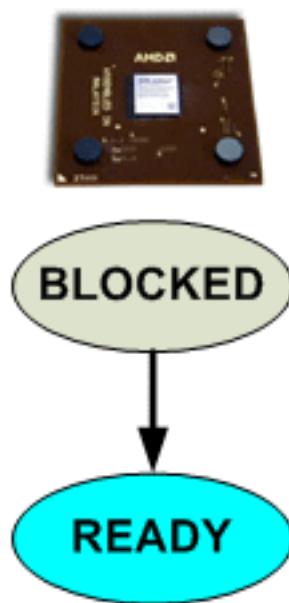
Sleep нити

Sleep()



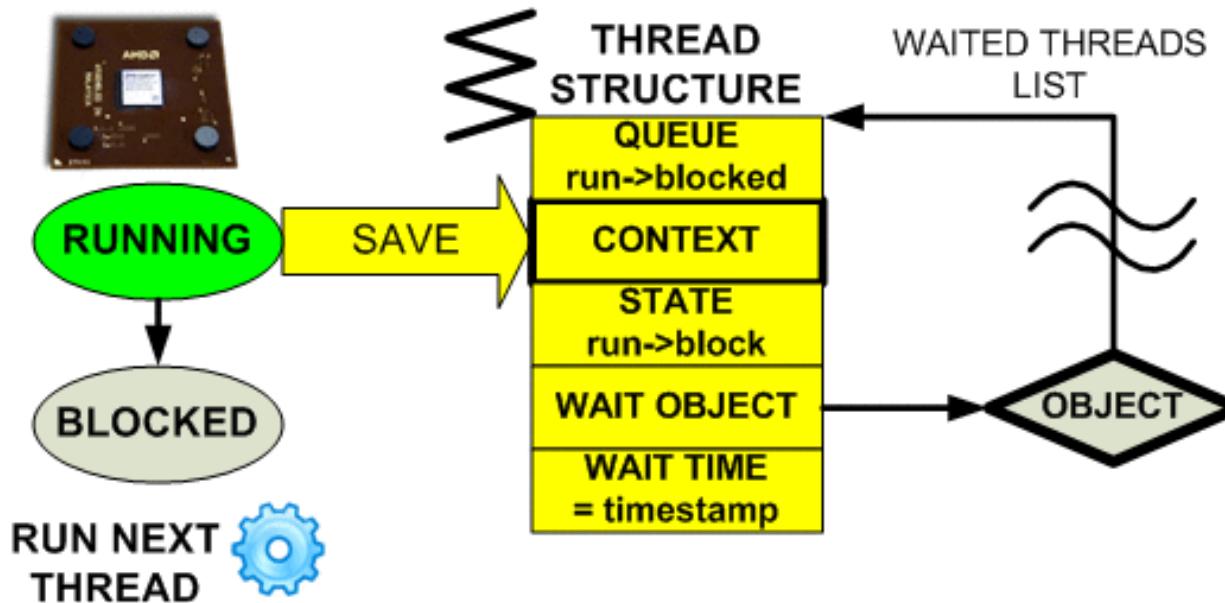
Sleep нити

Sleep() finished



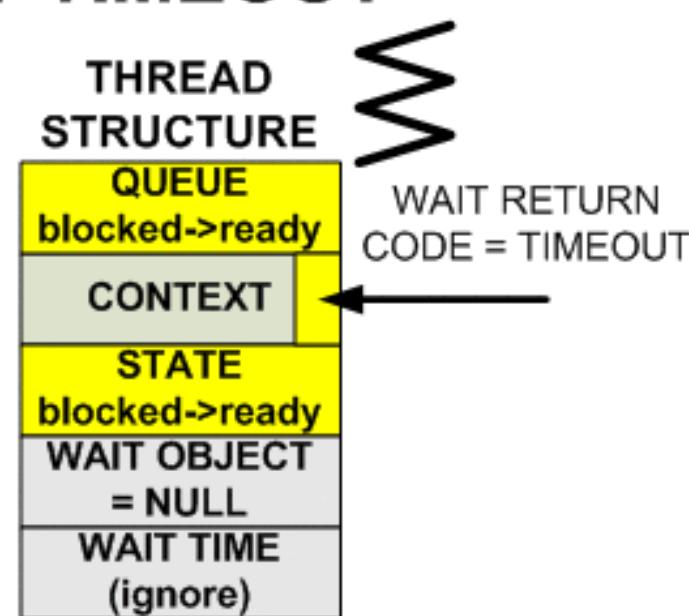
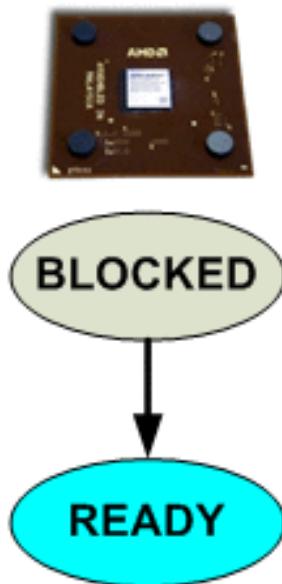
Wait нити

WaitForSingleObject()



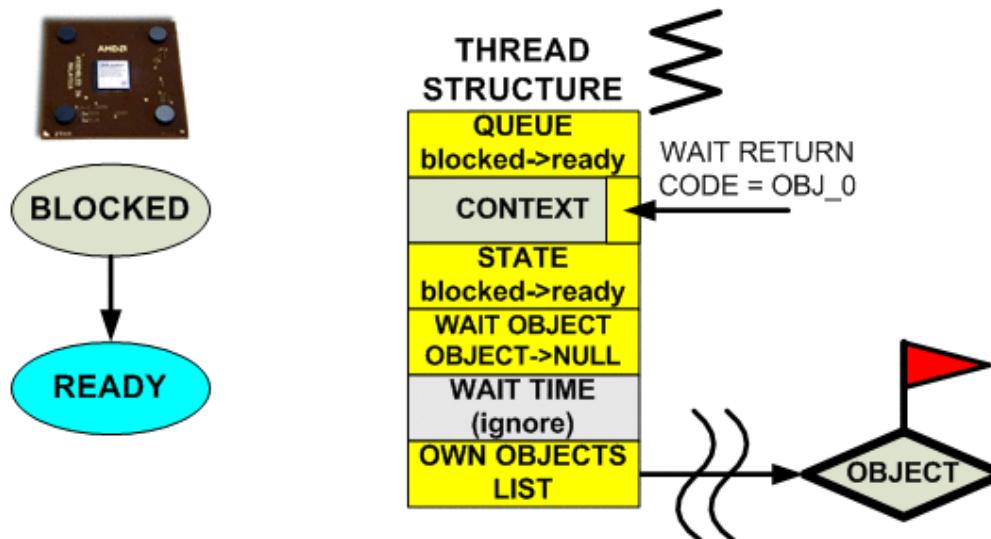
Wait нити

WAIT TIMEOUT



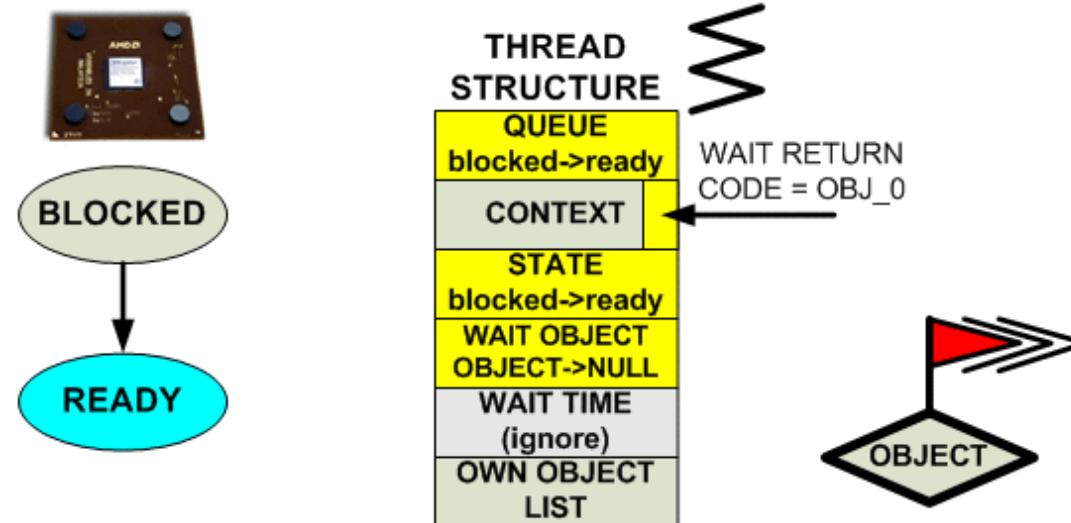
Wait нити

WAIT SUCCESS (SYNCHRONIZATION OBJECT)



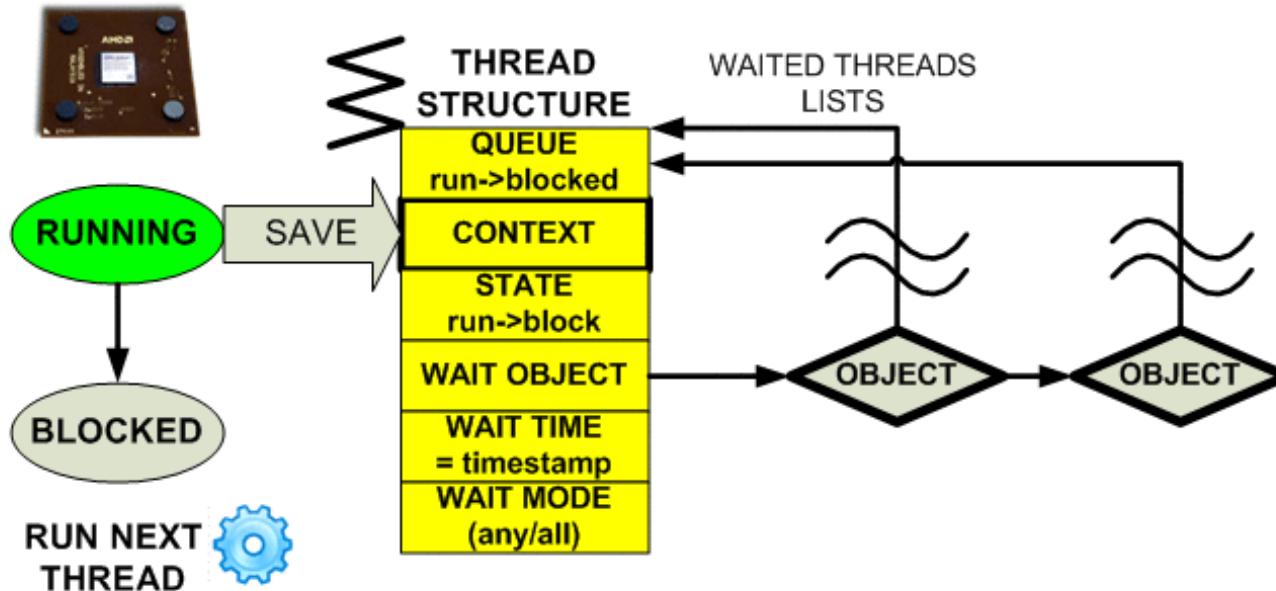
Wait нити

WAIT SUCCESS (NOTIFICATION OBJECT)



Wait нити

WaitForMultipleObject()



Boost priority нити

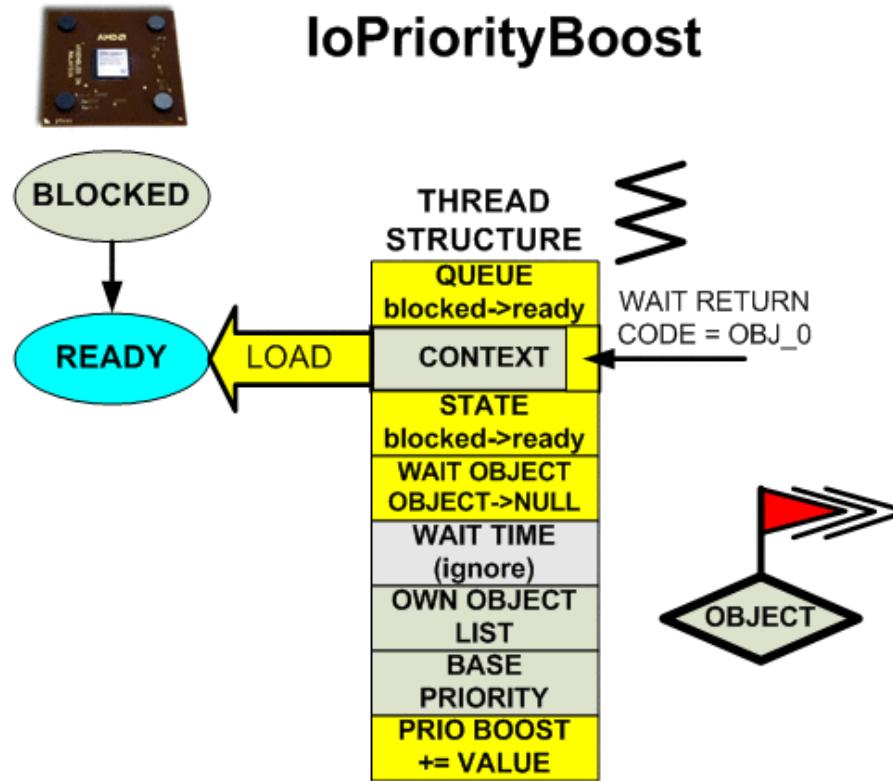
TimedBoost



THREAD STRUCTURE
QUEUE ready
CONTEXT
STATE = RDY
BASE PRIORITY
PRIOR BOOST += 1

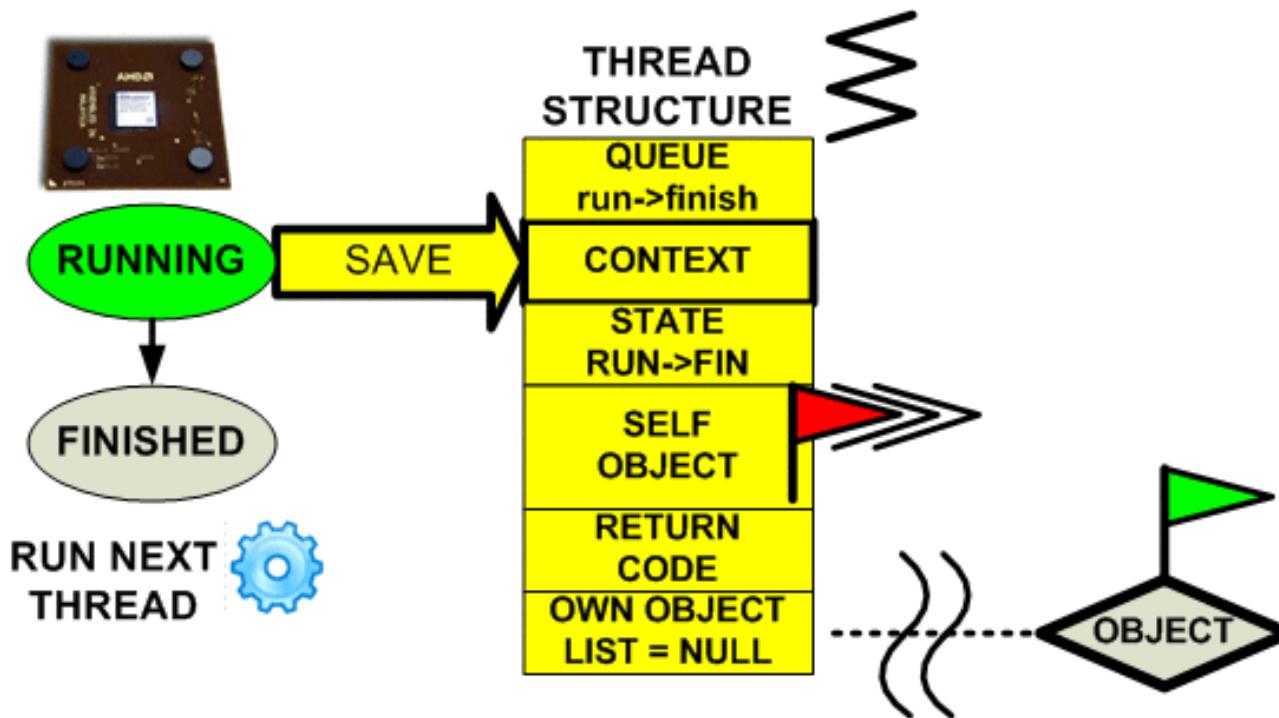


I/O priority boost нити



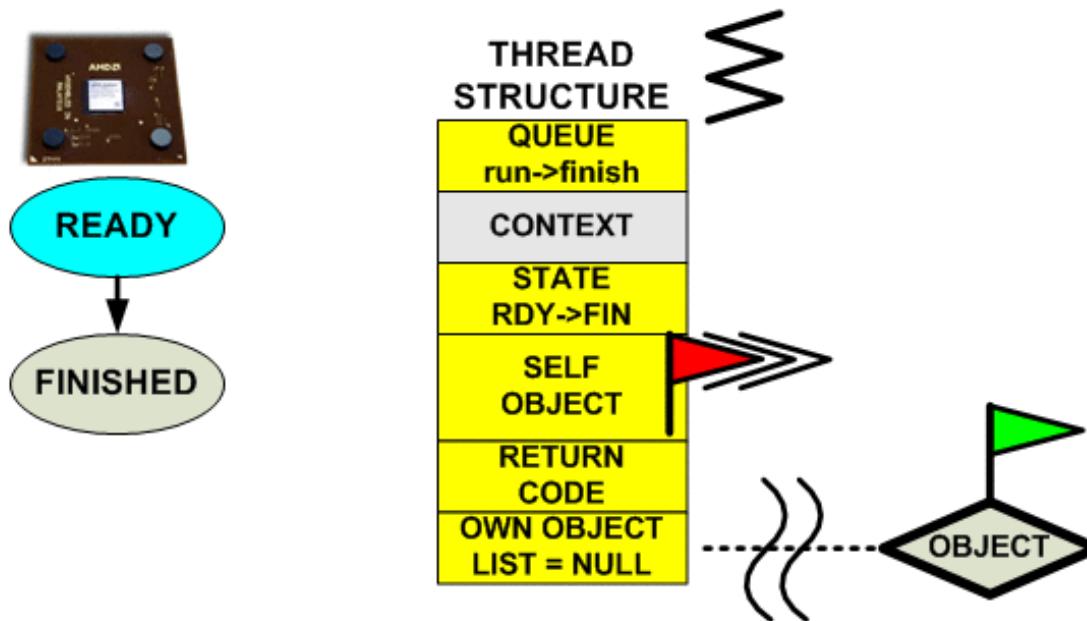
Terminate нити

TerminateThread()



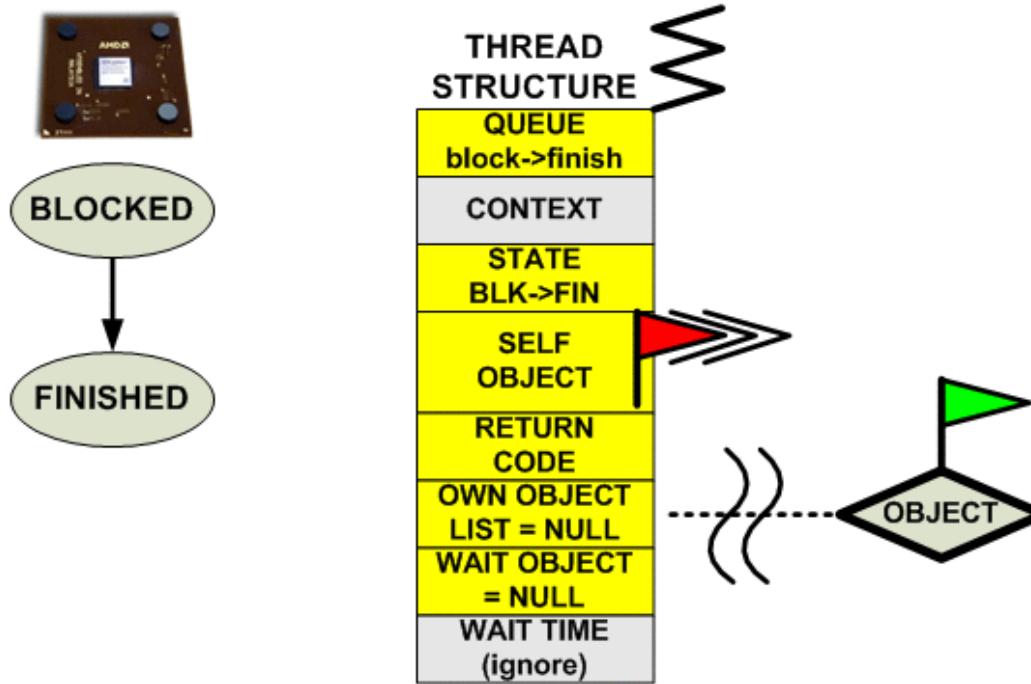
Terminate нити

TerminateThread()



Terminate нити

TerminateThread()



Deadlock нити

DeadLockDetection

