**Лекция 8-1 Планирование процессов в Windows**

В разных ОС процессы реализуются по-разному. Эти различия заключаются в том, какими структурами данных представлены процессы, как они именуются, какими способами защищены друг от друга и какие отношения существуют между ними. Процессы Windows NT имеют следующие характерные свойства:

-        Процессы Windows NT реализованы в форме объектов, и доступ к ним осуществляется посредством службы объектов.

-        Процесс Windows NT имеет многонитевую организацию.

-        Как объекты-процессы, так и объекты-нити имеют встроенные средства синхронизации.

-        Менеджер процессов Windows NT не поддерживает между процессами отношений типа "родитель-потомок".

В любой системе понятие "процесс" включает следующее:

-        исполняемый код,

-        собственное адресное пространство, которое представляет собой совокупность виртуальных адресов, которые может использовать процесс,

-        ресурсы системы, такие как файлы, семафоры и т.п., которые назначены процессу операционной системой.

-        хотя бы одну выполняемую нить.

Адресное пространство каждого процесса защищено от вмешательства в него любого другого процесса. Это обеспечивается механизмами виртуальной памяти. Операционная система, конечно, тоже защищена от прикладных процессов. Чтобы выполнить какую-либо процедуру ОС или прочитать что-либо из ее области памяти, нить должна выполняться в режиме ядра. Пользовательские процессы получают доступ к функциям ядра посредством системных вызовов. В пользовательском режиме выполняются не только прикладные программы, но и защищенные подсистемы Windows NT.

В Windows NT процесс - это просто объект, создаваемый и уничтожаемый менеджером объектов. Объект-процесс, как и другие объекты, содержит заголовок, который создает и инициализирует менеджер объектов. Менеджер процессов определяет атрибуты, хранимые в теле объекта-процесса, а также обеспечивает системный сервис, который восстанавливает и изменяет эти атрибуты.

В число атрибутов тела объекта-процесса входят:

-        Идентификатор процесса - уникальное значение, которое идентифицирует процесс в рамках операционной системы.

-        Токен доступа - исполняемый объект, содержащий информацию о безопасности.

-        Базовый приоритет - основа для исполнительного приоритета нитей процесса.

-        Процессорная совместимость - набор процессоров, на которых могут выполняться нити процесса.

-        Предельные значения квот - максимальное количество страничной и нестраничной системной памяти, дискового пространства, предназначенного для выгрузки страниц, процессорного времени - которые могут быть использованы процессами пользователя.

-        Время исполнения - общее количество времени, в течение которого выполняются все нити процесса.

Объект-нить имеет следующие атрибуты тела:

-        Идентификатор клиента - уникальное значение, которое идентифицирует нить при ее обращении к серверу.

-        Контекст нити - информация, которая необходима ОС для того, чтобы продолжить выполнение прерванной нити. Контекст нити содержит текущее состояние регистров, стеков и индивидуальной области памяти, которая используется подсистемами и библиотеками.

-        Динамический приоритет - значение приоритета нити в данный момент.

-        Базовый приоритет - нижний предел динамического приоритета нити.

-        Процессорная совместимость нитей - перечень типов процессоров, на которых может выполняться нить.

-        Время выполнения нити - суммарное время выполнения нити в пользовательском режиме и в режиме ядра, накопленное за период существования нити.

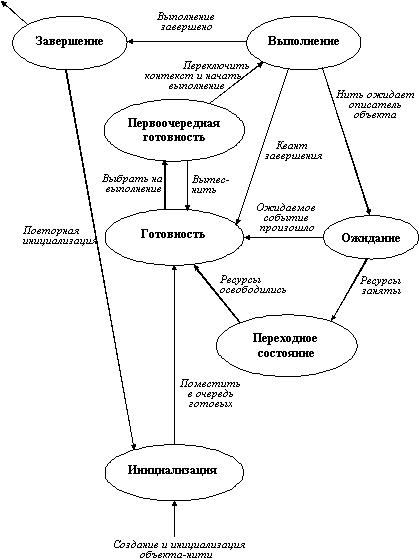
-        Состояние предупреждения - флаг, который показывает, что нить должна выполнять вызов асинхронной процедуры.

-        Счетчик приостановок - текущее количество приостановок выполнения нити.

Как видно из перечня, многие атрибуты объекта-нити аналогичны атрибутам объекта-процесса. Весьма сходны и сервисные функции, которые могут быть выполнены над объектами-процессами и объектами-нитями: создание, открытие, завершение, приостановка, запрос и установка информации, запрос и установка контекста и другие функции.

В Windows NT реализована вытесняющая многозадачность, при которой операционная система не ждет, когда нить сама захочет освободить процессор, а принудительно снимает ее с выполнения после того, как та израсходовала отведенное ей время (квант), или если в очереди готовых появилась нить с более высоким приоритетом. При такой организации разделения процессора ни одна нить не займет процессор на очень долгое время.

В ОС Windows NT нить в ходе своего существования может иметь одно из шести состояний (рисунок 1.3). Жизненный цикл нити начинается в тот момент, когда программа создает новую нить. Запрос передается NT executive, менеджер процессов выделяет память для объекта-нити и обращается к ядру, чтобы инициализировать объект-нить ядра. После инициализации нить проходит через следующие состояния:



*Рис. 1.3. Граф состояний нити*

-        *Готовность*. При поиске нити на выполнение диспетчер просматривает только нити, находящиеся в состоянии готовности, у которых есть все для выполнения, но не хватает только процессора.

-        *Первоочередная готовность* (standby). Для каждого процессора системы выбирается одна нить, которая будет выполняться следующей (самая первая нить в очереди). Когда условия позволяют, происходит переключение на контекст этой нити.

-        *Выполнение*. Как только происходит переключение контекстов, нить переходит в состояние выполнения и находится в нем до тех пор, пока либо ядро не вытеснит ее из-за того, что появилась более приоритетная нить или закончился квант времени, выделенный этой нити, либо нить завершится вообще, либо она по собственной инициативе перейдет в состояние ожидания.

-        *Ожидание*. Нить может входить в состояние ожидания несколькими способами: нить по своей инициативе ожидает некоторый объект для того, чтобы синхронизировать свое выполнение; операционная система (например, подсистема ввода-вывода) может ожидать в интересах нити; подсистема окружения может непосредственно заставить нить приостановить себя. Когда ожидание нити подойдет к концу, она возвращается в состояние готовности.

-        *Переходное состояние*. Нить входит в переходное состояние, если она готова к выполнению, но ресурсы, которые ей нужны, заняты. Например, страница, содержащая стек нити, может быть выгружена из оперативной памяти на диск. При освобождении ресурсов нить переходит в состояние готовности.

-        *Завершение*. Когда выполнение нити закончилось, она входит в состояние завершения. Находясь в этом состоянии, нить может быть либо удалена, либо не удалена. Это зависит от алгоритма работы менеджера объектов, в соответствии с которым он и решает, когда удалять объект. Если executive имеет указатель на объект-нить, то она может быть инициализирована и использована снова.

Диспетчер ядра использует для определения порядка выполнения нитей алгоритм, основанный на приоритетах, в соответствии с которым каждой нити присваивается число - приоритет, и нити с более высоким приоритетом выполняются раньше нитей с меньшим приоритетом. В самом начале нить получает приоритет от процесса, который создает ее. В свою очередь, процесс получает приоритет в тот момент, когда его создает подсистема той или иной прикладной среды. Значение базового приоритета присваивается процессу системой по умолчанию или системным администратором. Нить наследует этот базовый приоритет и может изменить его, немного увеличив или уменьшив. На основании получившегося в результате приоритета, называемого приоритетом планирования, начинается выполнение нити. В ходе выполнения приоритет планирования может меняться.

Windows NT поддерживает 32 уровня приоритетов, разделенных на два класса - класс реального времени и класс переменных приоритетов. Нити реального времени, приоритеты которых находятся в диапазоне от 16 до 31, являются более приоритетными процессами и используются для выполнения задач, критичных ко времени.

Каждый раз, когда необходимо выбрать нить для выполнения, диспетчер прежде всего просматривает очередь готовых нитей реального времени и обращается к другим нитям, только когда очередь нитей реального времени пуста. Большинство нитей в системе попадают в класс нитей с переменными приоритетами, диапазон приоритетов которых от 0 до 15. Этот класс имеет название "переменные приоритеты" потому, что диспетчер настраивает систему, выбирая (понижая или повышая) приоритеты нитей этого класса.

Алгоритм планирования нитей в Windows NT объединяет в себе обе базовых концепции - квантование и приоритеты. Как и во всех других алгоритмах, основанных на *квантовании*, каждой нити назначается квант, в течение которого она может выполняться. Нить освобождает процессор, если:  блокируется, уходя в состояние ожидания; завершается; исчерпан квант; в очереди готовых появляется более приоритетная нить.

Использование *динамических приоритетов*, изменяющихся во времени, позволяет реализовать адаптивное планирование, при котором не дискриминируются интерактивные задачи, часто выполняющие операции ввода-вывода и недоиспользующие выделенные им кванты. Если нить полностью исчерпала свой квант, то ее приоритет понижается на некоторую величину. В то же время приоритет нитей, которые перешли в состояние ожидания, не использовав полностью выделенный им квант, повышается. Приоритет не изменяется, если нить вытеснена более приоритетной нитью.

Для того, чтобы обеспечить хорошее время реакции системы, алгоритм планирования использует наряду с квантованием концепцию *абсолютных приоритетов*. В соответствии с этой концепцией при появлении в очереди готовых нитей такой, у которой приоритет выше, чем у выполняющейся в данный момент, происходит смена активной нити на нить с самым высоким приоритетом.

В многопроцессорных системах при диспетчеризации и планировании нитей играет роль их процессорная совместимость: после того, как ядро выбрало нить с наивысшим приоритетом, оно проверяет, какой процессор может выполнить данную нить и, если атрибут нити "процессорная совместимость" не позволяет нити выполняться ни на одном из свободных процессоров, то выбирается следующая в порядке приоритетов нить.