Планирование процессов

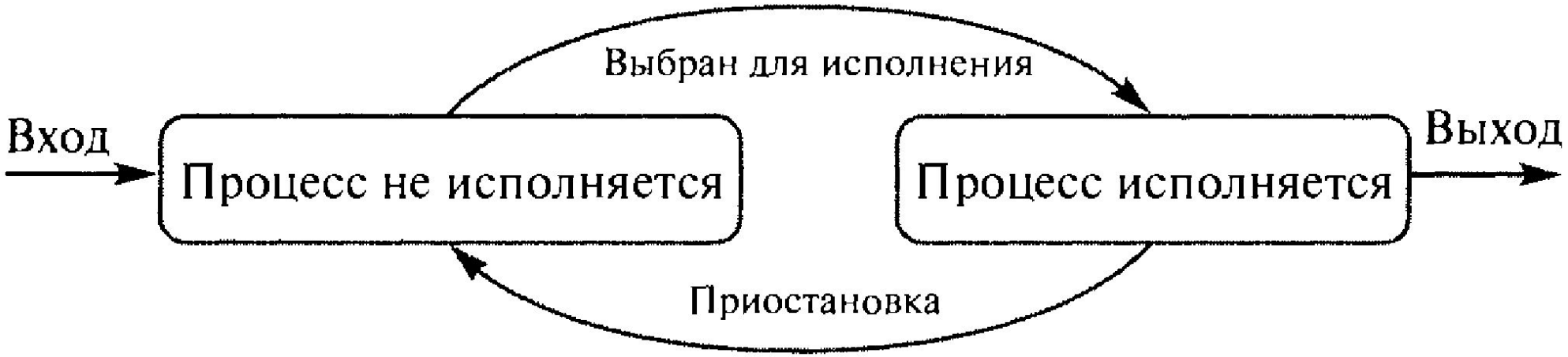
Под ***процессом*** обычно понимается последовательность операций при выполнении *программы* или ее части в совокупности с используемыми данными. В общем случае процесс и программа представляют собой разные понятия.

Планирование процессов

с каждым процессом связан **определенный набор ресурсов**,
Процесс — это контейнер, в котором содержится вся информация, необходимая для работы программы.

Планирование процессов

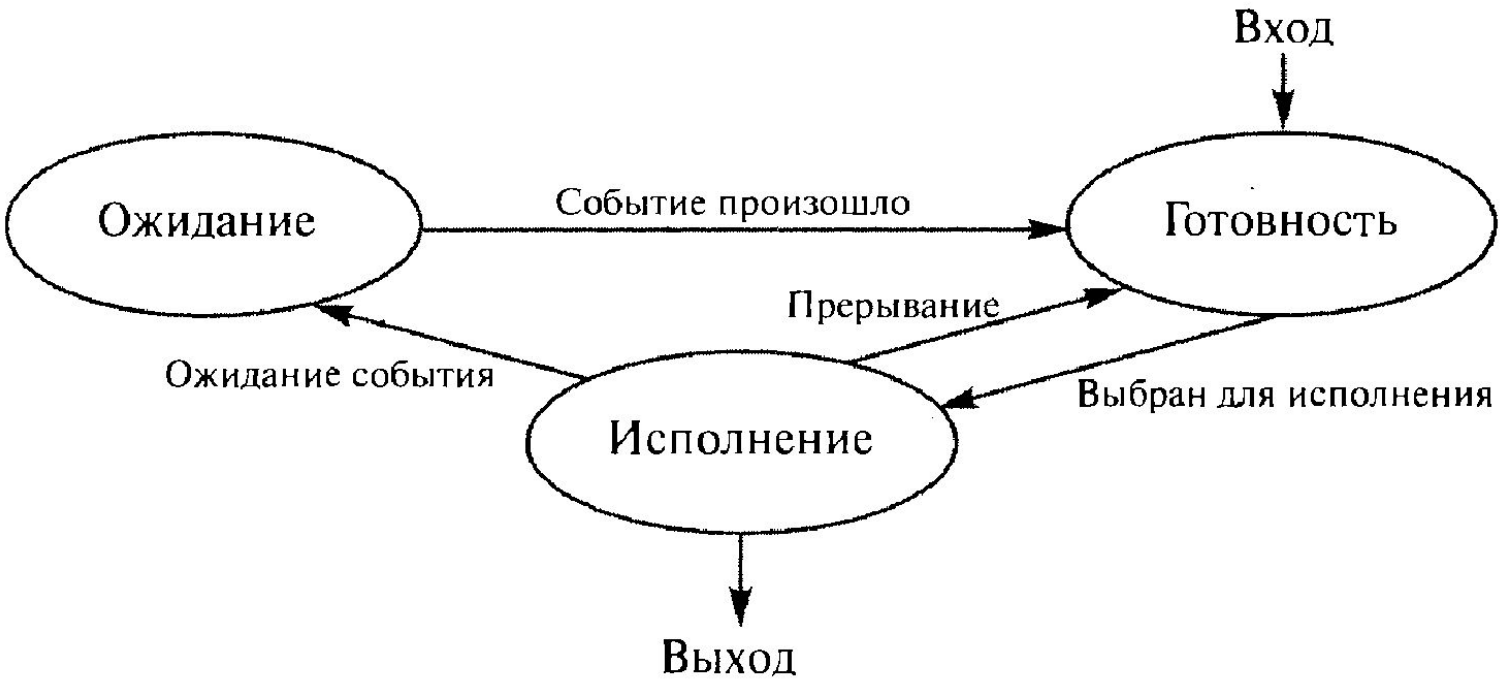
каждый процесс может находиться как минимум в 2-х состояниях: ***процесс исполняется*** и ***процесс не исполняется***.



модель не учитывает, в частности, то, что процесс, выбранный для исполнения, может все еще ждать события, из-за которого он был приостановлен, и реально к выполнению не готов

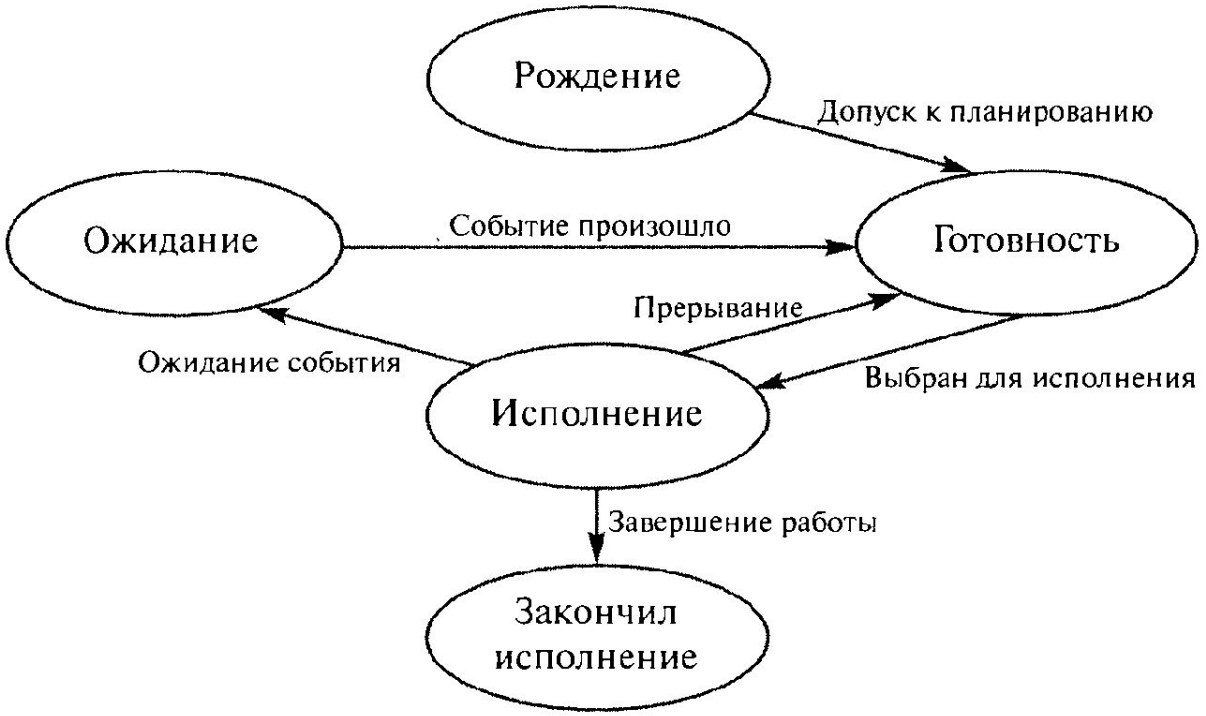
Планирование процессов

разобьем состояние **процесс не исполняется** на два новых состояния: ***готовность*** и ***ожидание*** .



новая модель хорошо описывает поведение процессов во время их существования, но она не акцентирует внимания на появлении процесса в системе и его исчезновении.

Планирование процессов



При рождении процесс получает в свое распоряжение ***адресное пространство***, в которое загружается программный код процесса; ему выделяются стек и системные ресурсы; устанавливается начальное значение программного счетчика этого процесса и т. д.

Планирование процессов

В конкретных операционных системах состояния процесса могут быть еще более детализированы, могут появиться некоторые новые варианты переходов из одного состояния в другое. Так, например, модель состояний процессов для операционной системы Windows NT содержит 7 различных состояний, а для операционной системы Unix — 9. Тем не менее, все операционные системы подчиняются модели из 5 состояний.

Планирование процессов

- Состояния процесса
-
- Рождение.
- Готовность.
- Ожидание.
- Исполнение.
- Закончил исполнение..

Планирование процессов

Важнейшей частью операционной системы, непосредственно влияющей на функционирование вычислительной машины, является ***подсистема управления процессами***.

Для операционной системы процесс представляет собой единицу работы, заявку на потребление системных ресурсов.

Планирование процессов

Изменяя состояния процессов, ОС выполняет следующие операции:

- **создание процесса**;
- **завершение процесса**;
- **приостановка процесса** (перевод из состояния исполнение в состояние готовность);
- **запуск процесса** (перевод из состояния готовность в состояние исполнение);
- **блокирование процесса** (перевод из состояния исполнение в состояние ожидание);
- **разблокирование процесса** (перевод из состояния ожидание в состояние готовность);
- **изменение приоритета процесса**.

Планирование процессов

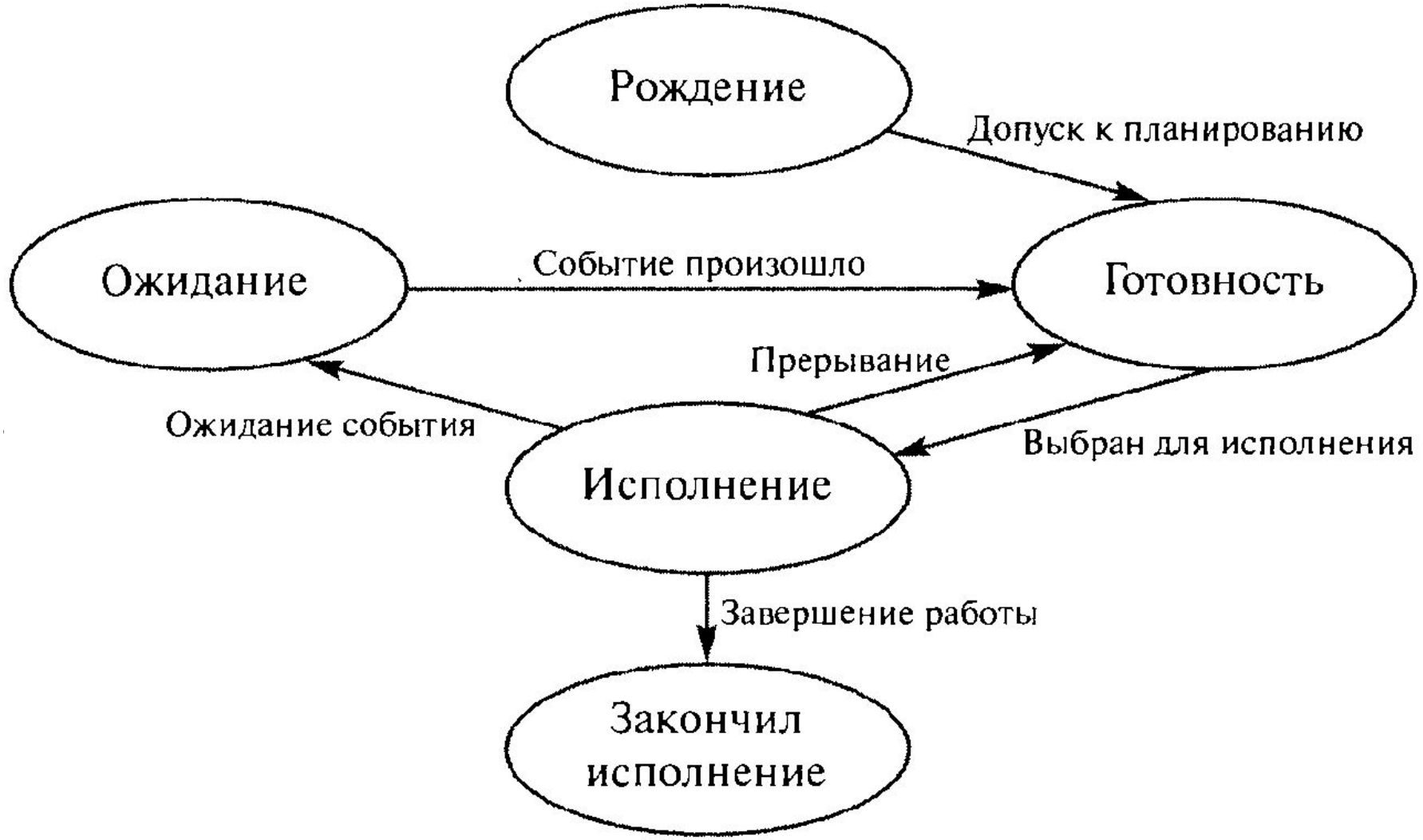
Планирование - обеспечение поочередного доступа процессов к одному процессору. Та часть операционной системы, на которую возложено *планирование*, называется **планировщиком**, а алгоритм, который ею используется, называется **алгоритмом планирования**.

Планировщик может принимать решения о выборе для исполнения нового процесса из числа находящихся в состоянии готовность

Планирование процессов

- Когда процесс переводится из состояния ***исполнение*** в состояние ***закончил исполнение***. Процесс больше не может выполняться (поскольку он уже не существует), поэтому нужно выбрать какой-нибудь процесс из числа готовых к выполнению. Если готовые к выполнению процессы отсутствуют, обычно запускается предоставляемый системой холостой процесс.
- Когда процесс переводится из состояния ***исполнение*** в состояние ***ожидание***. Процесс больше не может выполняться (т. к. ждет какое-то событие), поэтому надо выбрать на выполнение некоторый готовый процесс.
- Когда процесс переводится из состояния ***исполнение*** в состояние ***готовность*** (например, после прерывания от таймера). Планировщик должен решить, какой процесс ему запускать: тот, что только что перешел в состояние готовности, тот, который был запущен за время прерывания, или какой-нибудь третий процесс.
- Когда процесс переводится из состояния ***ожидание*** в состояние ***готовность*** (завершилась операция ввода-вывода или произошло другое событие). Планировщик должен решить, какой процесс ему запускать: тот, что только что перешел в состояние готовности, тот, который был запущен за время ожидания, или какой-нибудь третий процесс.

Планирование процессов



Планирование процессов

В случаях 1 и 2 процесс, находившийся в состоянии ***исполнение***, не может дальше исполняться, и операционная система вынуждена осуществлять планирование, выбирая новый процесс для выполнения. В случаях 3 и 4 планирование может как проводиться, так и не проводиться, планировщик не вынужден обязательно принимать решение о выборе процесса для выполнения, процесс, находившийся в состоянии ***исполнение*** может просто продолжить свою работу.

Планирование процессов

Если в операционной системе планирование осуществляется только в вынужденных ситуациях, говорят, что имеет место ***невытесняющее планирование (неприоритетный алгоритм)***. При таком режиме планирования процесс занимает столько процессорного времени, сколько ему необходимо. При этом переключение процессов возникает только при желании самого исполняющегося процесса передать управление (для ожидания завершения операции ввода-вывода или по окончании работы).

Планирование процессов

Если планировщик принимает и вынужденные, и невынужденные решения, говорят о ***вытесняющем планировании (приоритетный алгоритм)***. Термин "вытесняющее планирование" возник потому, что исполняющийся процесс помимо своей воли может быть вытеснен из состояния исполнение другим процессом. Вытесняющее планирование обычно используется в системах разделения времени. В этом режиме планирования процесс может быть приостановлен в любой момент исполнения.

Планирование процессов

Предмет оптимизации для планировщика не может совпадать во всех системах. При этом стоит различать три среды:

1) пакетную;
2) интерактивную;
3) реального времени.

Планирование процессов

В пакетных системах нет пользователей, работающих в интерактивном режиме. Поэтому для них зачастую приемлемы неприоритетные алгоритмы или приоритетные алгоритмы с длительными периодами для каждого процесса. Такой подход сокращает количество переключений между процессами, повышая при этом производительность работы системы. .

Планирование процессов

В среде с пользователями, работающими в интерактивном режиме, приобретает важность приоритетность, сдерживающая отдельный процесс от захвата центрального процессора. Для предупреждения такого поведения необходимо использование приоритетного алгоритма. Под эту категорию подпадают и серверы, поскольку они, как правило, обслуживают нескольких (удаленных) пользователей.

Планирование процессов

В системах, ограниченных условиями реального времени, приоритетность иногда не требуется, поскольку процессы могут запускаться только на непродолжительные периоды времени, и зачастую выполняют свою работу довольно быстро, а затем блокируются. В отличие от интерактивных систем в системах реального времени запускаются лишь те программы, которые предназначены для содействия определенной прикладной задаче.

Планирование в пакетных системах

«Первый пришел - первым обслужен» (FIFO - First In Fist Out)

Простейший неприоритетный алгоритм.

Процессор выделяется процессам в порядке поступления их запросов. Используется одна очередь процессов, находящихся в состоянии готовности. По мере поступления других заданий они помещаются в конец очереди. В случае блокировки процесса, при достижении состояния готовности, , он, помещается в конец очереди.

Достоинства: простота, справедливость.

Планирование в пакетных системах

«Первый пришел - первым обслужен» (FIFO - First In Fist Out)

Простейший неприоритетный алгоритм.

Достоинства: простота, справедливость.

Недостатки: среднее время ожидания и среднее полное время выполнения для этого алгоритма существенно зависят от порядка расположения процессов в очереди.

Планирование в пакетных системах

<u>«Кратчайшее задание – первое»</u>
Неприоритетный алгоритм для пакетных систем, *в котором предполагается, что сроки выполнения заданий известны заранее. Когда в ожидании запуска во входящей очереди находится несколько равнозначных по важности заданий, планировщик выбирает сначала самое короткое задание.*

Планирование в пакетных системах

Приоритет наименьшему оставшемуся времени выполнения

Это версия алгоритма выполнения первым самого короткого задания.

При использовании этого алгоритма планировщик всегда выбирает процесс с самым коротким оставшимся временем выполнения. Время выполнения заданий нужно знать заранее. При поступлении нового задания выполняется сравнение общего времени его выполнения с оставшимися сроками выполнения текущих процессов. Если для выполнения нового задания требуется меньше времени, чем для завершения текущего процесса, этот процесс приостанавливается и запускается новое здание.

Планирование в пакетных системах

Приоритет наименьшему оставшемуся времени выполнения

Это версия алгоритма выполнения первым самого короткого задания.

При использовании этого алгоритма планировщик всегда выбирает процесс с самым коротким оставшимся временем выполнения. Время выполнения заданий нужно знать заранее. При поступлении нового задания выполняется сравнение общего времени его выполнения с оставшимися сроками выполнения текущих процессов. Если для выполнения нового задания требуется меньше времени, чем для завершения текущего процесса, этот процесс приостанавливается и запускается новое здание.